IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received September 15, 2021, accepted October 21, 2021, date of publication November 1, 2021,
date of current version November 8, 2021.

*Digital Object Identifier 10.1109/ACCESS.2021.3124672*

# Automated Waterloo Rubric for Concept Map Grading

**SHRESHT BHATIA[1], SAJAL BHATIA[2], AND IRFAN AHMED[3], (Senior Member, IEEE)**
[1]Tandon School of Engineering, New York University, Brooklyn, NY 11201, USA
[2]School of Computer Science and Engineering, Scared Heart University, Fairfield, CT 06825, USA
[3]Department of Computer Science, Virginia Commonwealth University, Richmond, VA 23284, USA

Corresponding author: Sajal Bhatia (bhatias@sacredheart.edu)

This work was supported by NSF Cyber Training under Award 2017371 and Award 2017337.

**ABSTRACT** Concept mapping is a well-known pedagogical tool to help students organize, represent, and develop an understanding of a topic. The grading of concept maps is typically manual, time-consuming, and tedious, especially for a large class. Existing research mostly focuses on topological scoring based-on structural features of concept maps. However, the scoring does not achieve comparable accuracy to well-defined rubrics for manual analysis on the quality of content in a concept map. This paper presents `Kastor`, a new method to automate the Waterloo Rubric of scoring concept maps by quantifying the rubric's quality assessment parameters. The evaluation is performed on a publicly-available dataset of 39 concept maps of two cybersecurity courses, i.e., digital forensics, and supervisory control and data acquisition (SCADA) system security. The evaluation results show that `Kastor` achieves the accuracy of around 84% and 95% (at accurate and close-to-accurate levels) for SCADA and forensics concept maps, respectively. Furthermore, `Kastor`'s comparison with a topological scoring method shows improvement by around 32% and 79% on SCADA and forensics concept maps, respectively.

**INDEX TERMS** Concept map, automatic grading, cybersecurity education.

## I. INTRODUCTION

Concept mapping is a process of representing a student's knowledge on a topic in a graph-like structure referred to as concept map [1]. It is a cognitively intensive task making the students recall their concepts on a subject and organize and relate them in a graphical representation. A concept map consists of circles (or boxes) and links; a circle represents a concept, while a link describes the relationship between the concepts connecting the two circles. Figure 1 shows a simple example of a concept map presenting a SCADA system's basic operations. A concept map begins with an abstract/broad concept (mainly the topic being addressed). It then adds circles and connecting links at different levels to identify more specific concepts and their relationships with each other as the map proceeds deeper into the hierarchy.

The grading of concept maps tends to be manual, tedious, and time-consuming. The automation of the grading process is a significant research problem to help teachers use the concept maps effectively in class. Existing research primarily focuses on topological scoring that utilizes structural features of a concept map for grading such as incoming child connections from a parent concept, total number of circles and links, and a number of propositions [2], [3]. Recently, Deshpande and Ahmed [3] show that a topological scoring [2] is not equally effective as compared to the Waterloo Rubric, which is a well-defined grading rubric for manual analysis [4].

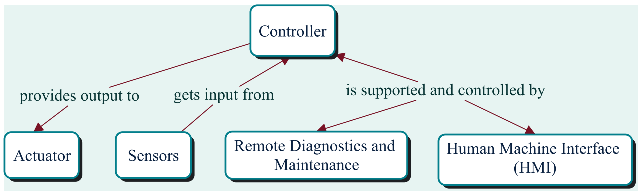

Controller
provides output to
gets input from
is supported and controlled by
Actuator
Sensors
Remote Diagnostics and Maintenance
Human Machine Interface (HMI)

**FIGURE 1.** Concept map example of basic SCADA operations.

In this paper, we propose a new method, `Kastor`, that automates the Waterloo Rubric [4] effectively to provide comparable grading of concept maps from the manual analysis. The Waterloo Rubric is developed by the University of Waterloo and defines quality metrics of a concept map such

The associate editor coordinating the review of this manuscript and approving it for publication was Ilsun You.

as the "breadth of net", "embeddedness and interconnectedness", "use of descriptive links", and "efficient links". `Kastor` quantifies the rubric by extracting and utilizing the keywords from a content-source of a concept map, e.g., textbook chapters and an instructor's PowerPoint slides. For instance, for the *use of descriptive links*, `Kastor` employs the wordnet similarity [5] to find the overall similarity of every link-node pair with the keywords from the overall source text

For the evaluation, we utilize a publicly-available dataset of 39 concept maps for two cybersecurity courses, i.e., digital forensics, and SCADA system security [6]. We compare `Kastor`'s automated analysis on the dataset with the ground truth of the manual analysis using the Waterloo Rubric. Overall, `Kastor` achieves around 84%, and 95% accuracy (at the accurate or close to accurate levels) for SCADA and forensics concept maps, respectively. We further compare `Kastor` with a topological scoring method [2] used by Deshpande and Ahmed [3] and show improvement by around 32% and 79% (at accurate and close-to-accurate levels) for SCADA and forensics concept maps, respectively.

The contributions of the paper are as follows:

- We develop a set of algorithms to automate the quality assessment parameters for concept maps in the Waterloo Rubric such as "breadth of net", and "embeddedness and interconnectedness".
- We create a grading tool, `Kastor` written in Python. We released it on GitHub at [7] for the community to use.
- We show the effectiveness of `Kastor` on a publicly-available dataset of 39 concept maps for two cybersecurity courses.

### A. ROADMAP

The rest of the paper is organized as follows: Section II provides the background and related work. Section III outlines the problem statement and an overview of the proposed approach. Section IV presents `Kastor` and its implementation, followed by its evaluation in V. Section VI concludes the paper.

## II. RELATED WORK

Concept mapping as a pedagogical tool has received little attention not only in cybersecurity education, but in general education as well [8]–[11]. Early users of concept maps, Novak and Gowin [12], use this technique to demonstrate comprehension of complex concepts on student interview data among young students. Beyerbach's [13] proposes iterative analysis of concepts by teachers and students to better understand the experience of teachers on a course along with the evolution of the viewpoint.

Dexter [14] uses concept maps to outline required concepts for cybersecurity management to delve into subtopics e.g., malicious behavior (deployment of code and usage of vulnerability scanners) on an organization's network, and managing key security policies. Similarly, Tanner and Dampier [15] utilizes concept maps in six phases of a digital investigative process i.e., identification, collection, preservation, examination, analysis, and presentation, as well as important procedures and concepts within each phase such as chain of custody. They show that a concept map thrives in presenting contexts of a particular evidence such as properties, dates, and how to best analyze the proof. The authors further detail how case-specific concept maps may be shared by the law enforcement community as well as how a concept map could be shown in court in order to detail a complex investigative process.

Hay *et al.* [16] propose the pedagogical use of concept mapping in a general higher educational and summarize the prior use of concept maps in both the teaching and learning processes. They focus on measuring the prior knowledge of a student to allow a teacher to schedule his lessons accordingly to avoid duplication of already known information.

Another line of research focuses on automated grading and scoring of concept maps. Anohina-Naumeca and Grundspenkis [17] assess various scoring schemes for student concept maps by comparing them with an expert-created concept map.

Cline *et al.* [18] developed a web-based concept map construction and rule-based evaluation system called the Concept Mapping Tool. The authors separated the creation of concept maps by students and instructor. Each concept map was compared to itself by the rule-based evaluation system to find a maximum score and to the top-level concept of the instructor's map to find the minimum score. The evaluation system didn't use natural language processing and therefore was not expected to perform at the level of a human instructor. Similarly Luckie *et al.* [19] also developed a web-based, concept mapping Java applet with automatic scoring. This used WordNet, an electronic lexical database and thesaurus, and compares the results with human scoring approaches. This approach showed only a 10% gain by using WordNet's synonyms for automatic grading and instructor-provided and WordNet-supplemented grading matrices together successfully grade 26% of the user made concept maps. Harrison *et al.* [20] confirmed that usage of WordNet increases the grading of concept maps.

Pinandito *et al.* [21] talked about developing a concept map authoring support tool, adopting a semi-automatic concept mapping approach to help teachers create concept maps. It finds that the support tool yields better concept mapping efficiency while maintaining concept maps of similar quality. Rye and Rubba [22] suggested usage of expert referents and emphasized concept relationships in assessment of concept maps.

Recently, Deshpande and Ahmed [3] assess the effectiveness of a state-of-the-art topological scoring method for the grading of concept maps automatically. They created 41 concept maps for two cybersecurity courses, digital forensics, and SCADA system security and then, manually applied the Waterloo Rubric to the maps to obtain the ground-truth about their quality. Deshpande and Ahmed further apply a topological scoring method to the maps and compared its results with the ground-truth. They concluded that compared to the Waterloo Rubric, the topological scoring is not equally accurate.

The proposed method, Kastor addresses the existing shortcomings of using concept maps as an effective pedagogical tool by automating the Waterloo Rubric, a well-defined grading rubric for manual analysis [4]. The proposed novel method reduces the time, provides consistency to the grading process as well as increases the accuracy over other proposed methods such as the topological scoring method [3].

## III. AUTOMATED GRADING OF CONCEPT MAPS

### A. PROBLEM STATEMENT

Given a concept map developed by a student using a textbook or an instructor's notes and slides, our goal is to score the map accurately and automatically, comparable to a manual analysis using a well-defined rubric. The challenge is that the analysis based only on the map's topology is not sufficient and must include qualitative parameters to achieve comparable accuracy of manual analysis.

### B. PROPOSED APPROACH - KASTOR

Figure 2 presents an overall methodology of Kastor to automate Waterloo Rubric for the qualitative analysis of concept maps. The rubric defines "embeddedness", "efficient links", "breadth of net", and "use of descriptive links" parameters for the evaluation of a concept map. Kastor quantifies these parameters by using the corresponding text source of a concept map. The text source include presentation slides, textbooks and other literary sources used to teach a topic. Kastor extracts keywords from the text source and then, uses wordnet similarity [5] and other comparison methods to assess the words in a concept map for grading.

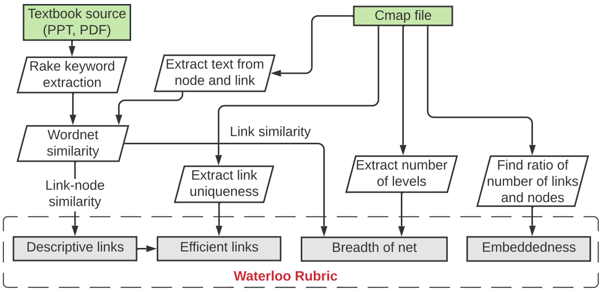


**FIGURE 2.** **Kastor methodology.**

Figure 2 illustrates how the results of the four Waterloo Rubric parameters are determined by Kastor. The root nodes (represented in green boxes) are the source files provided to Kastor along with a concept map (cmap) file for grading. Kastor extracts keywords from the source file using RAKE, an automatic keyword extraction technique [23]. It then employs the Wordnet similarity to find a similarity score between the keywords from the text source and cmap file. The wordnet similarity score of the link-node pairs completely determine the result of *Embeddedness* (algorithm 2). The result of *Efficient links* (algorithm 4) is the average of Embeddedness and the uniqueness of the links. The *breadth of net* (algorithm 1) is obtained by averaging the wordnet similarity score and the number of levels, while the *descriptive link* (algorithm 3) is the ratio of the number of links and nodes in the cmap.

## IV. DETAILED DESIGN OF KASTOR

Kastor consists of three modules. The first module identifies the text source, converts it into desired format and then extracts keywords from it. The second module finds the similarity score between the keywords of text source and concept map file using Wordnet similarity. The third module computes the scores of Waterloo Rubric parameters automatically to assign a grade to the map being analyzed.

### A. CONCEPT MAP PROCESSING

The keywords extracted from a text source using RAKE are used as the comparison metric. Words are extracted from the concept map, since concept maps are concise representations, all terms are viewed as keywords and compared to keywords from the text source. Concept Map files (.cmap) are exported in an Extensible Markup Language (.xml) format using the cmap application used to make concept maps [2]. The XML includes tags that define and distinguish nodes from the links in the concept map. These nodes are then used to extract the content of the link and nodes for similarity assessment. Atapattu *et al.* [24] talks about the process of extraction of concepts which provides a great base for our research.

### B. KEYWORD EXTRACTION

To extract keywords from the text source and its relevant concept maps, we utilize a natural language processing algorithm referred to as rapid automatic keyword extraction (RAKE) [23]. RAKE is a domain-independent keyword extraction algorithm that sifts through the text to extract keywords from documents, web pages, and any other form of literature. RAKE determines a keyword by analysing the *frequency of a word* and its *co-occurrence with other words* in a given text. RAKE achieves higher precision and similar recall when compared to other existing keyword extraction techniques extracting all the keywords in one pass making it more efficient and versatile [23].

Specifically, RAKE splits the document into a collection of words, separated by a word delimiter specified by the user. At phrase delimiters and stop-word positions, it splits this list into sequences of contiguous words. Words within a sequence are designated the same place in the text and together are deemed a candidate keyword. A score which is "defined as the sum of its member word scores" is then assigned to these keywords.

### C. WORDNET SIMILARITY OF KEYWORDS

We use wordnet similarity as a standard measure for various evaluations of Waterloo Rubric automation to calculate the similarity coefficient. Wordnet similarity is a method for measuring the semantic similarity of texts, using corpus-based and knowledge-based measures of similarity [5]. This method gives a score between 0 and 1 based on how similar a given sentence is. Kastor utilizes the similarity score to check the similarity of words in a concept map with the keywords extracted from a text source.

### D. AUTOMATING WATERLOO RUBRIC

The Waterloo Rubrics is developed by the University of Waterloo for manual evaluation of concept maps [4]. The rubric defines a four-level standard to identify the quality of concept maps i.e. Excellent, Good, Poor, Failing, based on six parameters: 1) Breadth of net, 2) Embeddedness and interconnectedness, 3) Use of descriptive links, 4) Efficient links, 5) Layout, and 6) Development over time. Kastor automates the first four parameters and does not consider the last two categories because they are not directly related to the quality of a concept map i.e., whether a concept map can be printed on one page, and the overall time to develop a concept map. Kastor quantifies the four parameters using their descriptions in the Waterloo Rubric.

#### 1) BREADTH OF NET

##### a: WATERLOO RUBRIC DESCRIPTION

Waterloo Rubric scores the *breadth of net* on the basis of the presence of significant concepts in the concept map at multiple levels. For excellent, the map includes the most important concepts and defines them on multiple levels. However, to fail, a map misses many important concepts.

##### b: KASTOR AUTOMATION

Based on the description in the Waterloo Rubric, Kastor solves the following two challenges:

1) *Evaluating the significance of target concepts* - Kastor uses wordnet similarity to get the similarity coefficient of each node by comparing it with the keywords extracted from the source text. To further divide the concept map into *Excellent*, *Good*, *Poor*, and *Bad*, Kastor uses the ratio of nodes with a similarity coefficient greater than 50 percent and the total nodes in the concept map. This ratio is converted into a percentage and then a grade is assigned using a predefined grading table.
2) *Multiple levels of concept map* - We convert a cmap file into an XML file. Kastor processes the XML file to find the root node in a concept map. Since the XML file has no details about the root node, Kastor finds it by listing all the nodes and removing those nodes which have no connections leading to them until it is left with only root nodes. After finding the root node, Kastor finds multiple levels by using a depth-first search algorithm and keeping count of the max number of levels in the concept map starting from the root node. Each node and link is counted as one level, so Kastor treats node-link pair as one level. Since XML cannot differentiate between a node and a link, Kastor divides the final result by 2 as between every two nodes there's at least one link.

##### c: IMPLEMENTATION

As previously outlined, based on the description in Waterloo Rubric, the breadth of net presents a two-part problem i.e., the inclusion of significant concepts, and multiple levels in a concept map. For multiple levels, we define the assigned grades along with both Waterloo and Kastor descriptions in Table 1. Similarly, for the inclusion of significant concepts, Table 2 is used. Both features have equal weights and hence Kastor averages their numerical values to assign an overall grade for breadth of net using the same convention, i.e., 3 (excellent), 2 (good), 1 (poor), and 0 (failing). Algorithm 1 shows the pseudo-code.

**Algorithm 1** Breadth of Net

```
Result: Quantified output of Breadth of Net
Find number of root nodes using backtracking
if no. of root nodes = 1 then
    use depth first search to find all the nodes  if node is not
     visited then
        visited.add(node)
        depth = depth+1
        continue finding maximum depth using Depth first
         search
    else
        move to next node
    end if
else
    for all root nodes in the list
        depth first Search to find the root node with maximum depth
    end for
end if
assign a score based on the depth
find Wordnet similarity score of the concept map and assign
 it a score
find average of both scores and assign it a grade based on
 the following
if average > 75 then
    result = "Excellent"
else if average > 50 then
    result = "Good"
else if average > 25 then
    result = "Poor"
else
    result = "Failing"
end if
```

**TABLE 1.** Grade assignment for multiple levels in breadth of net.

| Grades | Waterloo Description | Kastor Description | Score |
|---|---|---|---|
| Excellent | Describes domain on multiple levels | Concept map has more than 3 levels | 3 |
| Good | Describes domain on limited number of levels | Concept map has 2 levels | 2 |
| Poor | Describes domain on only one level | Concept map has only 1 level | 1 |
| Failing | Map includes minimum concepts | Concept map has no level i.e. only root node | 0 |

**TABLE 2.** Grade assignment for inclusion of concepts in breadth of net.

| Grades | Waterloo Description | Kastor Description | Score |
|---|---|---|---|
| Excellent | Map includes the important concepts | More than 75 percent of all concepts are present | 3 |
| Good | Map includes most important concepts | More than 50 percent of concepts are present | 2 |
| Poor | Important concepts missing | More than 25 percent of concepts are present | 1 |
| Failing | Map includes minimum concepts | Less than 25 percent of concepts are present | 0 |

**TABLE 3.** Grade assignment for embeddedness and interconnectedness.

| Grades | Waterloo Description | Kastor Description | Score |
|---|---|---|---|
| Excellent | All concepts interlinked with several other concepts | More than 75 % | 3 |
| Good | Most concepts interlinked with other concepts | 51- 75 % | 2 |
| Poor | Several concepts linked to other concepts | 26-50 % | 1 |
| Failing | Few concepts linked to other concepts | 0-25 % | 0 |

### 2) EMBEDDEDNESS AND INTERCONNECTEDNESS

#### *a: WATERLOO RUBRIC DESCRIPTION*

Waterloo Rubric scores the embeddedness and interconnectedness on the basis of how many nodes are interlinked with other nodes. For excellent, all concepts are interlinked, and for fail, few concepts are interlinked.

#### *b: KASTOR AUTOMATION*

For this feature, `Kastor` finds the ratio of the total number of links and the total number of nodes. It is based on the intuition that a concept map with high embeddedness will have a high number of links as compared to nodes. Note that if a highly interconnected concept map will have more links than the number of nodes, it makes it possible to have a percentage greater than 100.

#### *c: IMPLEMENTATION*

We define the assigned grades in Table 3 for this parameter along with the `Kastor` and Waterloo descriptions. Algorithm 2 shows the pseudo-code.

**Algorithm 2** Embeddedness

```
Result: Quantified output of Embeddedness
extract all the concepts and the links joining the concepts
  find the number of concepts and links
  embed = number of links / number of concepts * 100
  assign a score based on the following
  if embed > 75 then
    result = “Excellent”
else if embed > 50 then
    result = “Good”
else if embed > 25 then
    result = “Poor”
else
    result = “Failing”
end if
```

**TABLE 4.** Grade assignment for descriptive links.

| Grades | Waterloo Description | Kastor Description | Score |
|---|---|---|---|
| Excellent | Links succinctly and accurately describe all relationships | More than 75 % similarity | 3 |
| Good | Links are descriptive and valid for most relationships | 51- 75 % similarity | 2 |
| Poor | Some links unclear or vague; some invalid or unclear | 26-50 % similarity | 1 |
| Failing | Links are vague; show inconsistent relationships | 0-25 % similarity | 0 |

### 3) USE OF DESCRIPTIVE LINKS

#### *a: WATERLOO RUBRIC DESCRIPTION*

Waterloo Rubric scores the descriptive links based on the quality of the description of the links used in a concept map. A concept map is excellent in which links succinctly and accurately describe all relationships while a map is failing if links are vague and don’t define relationships accurately.

#### *b: KASTOR AUTOMATION*

`Kastor` uses wordnet similarity to find the overall similarity of every link-node pair i.e. combination of the node and the connecting link, with the keywords from the overall text. Every link-node pair with a coefficient of similarity greater than 0.5 is considered to be similar. `Kastor` takes the ratio of the number of similar pairs and the total number of link-node pairs.

#### *c: IMPLEMENTATION*

The Waterloo description, `Kastor` definition, assigned ranges, and numerical scores for descriptive links element of the concept map are defined in Table 4. Algorithm 3 shows the pseudo-code.

### 4) EFFICIENT LINKS

#### *a: WATERLOO RUBRIC DESCRIPTION*

This is graded on basis of the uniqueness of links and the quality of them. For excellent, each link type is distinct and

**Algorithm 3** Use of Descriptive Links

```
Result: Quantified output of Use of descriptive links
extract all the concepts and the links joining the concepts
  create link-node pairs
  find the overall similarity of the link-node pairs
  if $similarity > 0.5$ then
      similarcount = similarcount + 1
      count = count + 1
else
    count = count + 1
end if
similarityresult = similarcount / count * 100
  assign a score based on the following
  if $similarityresult > 75$ then
    result = "Excellent"
else if $similarityresult > 50$ then
    result = "Good"
else if $similarityresult > 25$ then
    result = "Poor"
else
    result = "Failing"
end if
```

**Algorithm 4** Efficient Links

```
Result: Quantified output of Efficient links
extract all the concepts and the links joining the concepts
  create link-node pairs
  find the overall similarity of the link-node pairs
  if $similarity > 0.5$ then
      similarcount = similarcount + 1
      count = count + 1
else
    count = count + 1
end if
similarityresult = similarcount / count * 100
  assign a score based on the following
  if $similarityresult > 75$ then
    result = "Excellent"
else if $similarityresult > 50$ then
    result = "Good"
else if $similarityresult > 25$ then
    result = "Poor"
else
    result = "Failing"
end if
```

clearly describes the relationship, while for fail, most links are vaguely described, and not distinct from each other.

#### b: KASTOR AUTOMATION

Kastor solves the following two challenges based-on the description of the Waterloo Rubric:

1) *Evaluating uniqueness of the links* - Kastor finds the ratio of the number of unique links and the total number of links.
2) *Quality of description of the links* - Kastor leverages the results of Embeddedness and Interconnectedness. Since Embeddedness and Interconnectedness describes the quality of the interlinkage of nodes, the result of this is directly taken and used for Efficient link.

#### c: IMPLEMENTATION

Table 5 gives the Kastor definition and grading criteria for efficient links. The second part of this concept map element (link description) takes input from the previous element (descriptive links) and takes an average of both scores to compute the final score assigned to this element along with the grade, i.e., 3 (excellent), 2 (good), 1 (poor), and 0 (failing). Algorithm 4 shows the pseudo-code.

**TABLE 5. Grade assignment for efficient links.**

| Grades | Waterloo Description | Kastor Description | Score |
|---|---|---|---|
| Excellent | Each link type is distinct from all others | 100 % links are unique | 3 |
| Good | Most links are distinct from others | >75 % of links unique | 2 |
| Poor | Several links are synonymous | <75% of links unique and similarity score of links less than 0.5 | 1 |
| Failing | Most links are synonymous | <75% of links unique and similarity score of links greater than 0.5 | 0 |

## V. EVALUATION

### A. CONCEPT MAP DATASET

We evaluate Kastor on a publicly available 39 concept maps of two cybersecurity courses viz., 20 maps for SCADA system security, and 19 maps for digital forensics [6].

1) *SCADA System Security* - SCADA systems monitor and control critical U.S. infrastructure, such as the power grid, pipelines, water management, etc. It is of primary importance to protect their integrity and availability [3]. The dataset has 20 concept maps on SCADA security covering topics on SCADA network protocols, ladder logic programming of programming logic controllers, cyberattacks on SCADA systems, and case studies on smart cities and power grid stations.
2) *Digital Forensics* - defined as the application of scientific tools and methods to identify, collect, and analyze digital artifacts in support of legal proceedings [25]. The dataset has 19 concept maps on digital forensics covering topics on evidence acquisition, file system analysis, sleuthkit and volatility frameworks, forensic analysis of web browsers and Windows registry.

### B. KASTOR ACCURACY

#### 1) ACCURACY LEVELS

We use manual Waterloo Rubric scores of the dataset as a ground truth to measure the accuracy of Kastor. For this purpose, we define four levels of accuracy i.e., *accurate*, *close to accurate*, *close to inaccurate*, and *inaccurate*.

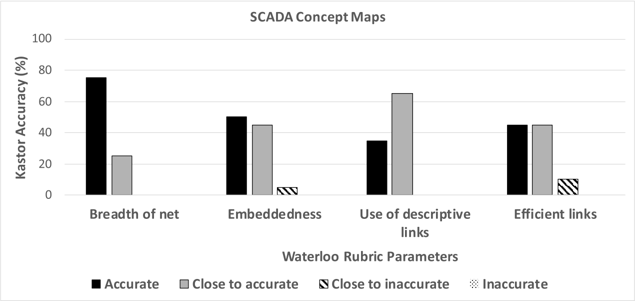


**FIGURE 3.** **SCADA system security - Kastor accuracy using the ground-truth obtained from the manual Waterloo Rubric analysis of concept maps.**

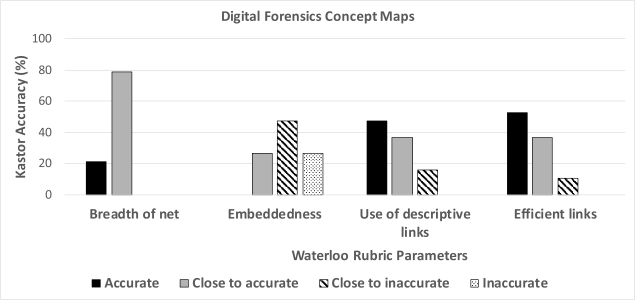


**FIGURE 4.** **Digital forensics - Kastor accuracy using the ground-truth obtained from the manual Waterloo Rubric analysis of concept maps.**

- *Accurate (level-4).* If the `Kastor`'s score is the same as that of manual rubric score, it is accurate.
- *Close to accurate (level-3).* If the `Kastor`'s score deviates by one point, it is close to accurate.
- *Close to inaccurate (level-2) and Inaccurate (level-1).* If the deviations are by two and three points, they are identified as close to inaccurate, and inaccurate, respectively.

### 2) KASTOR ACCURACY ON WATERLOO RUBRIC PARAMETERS

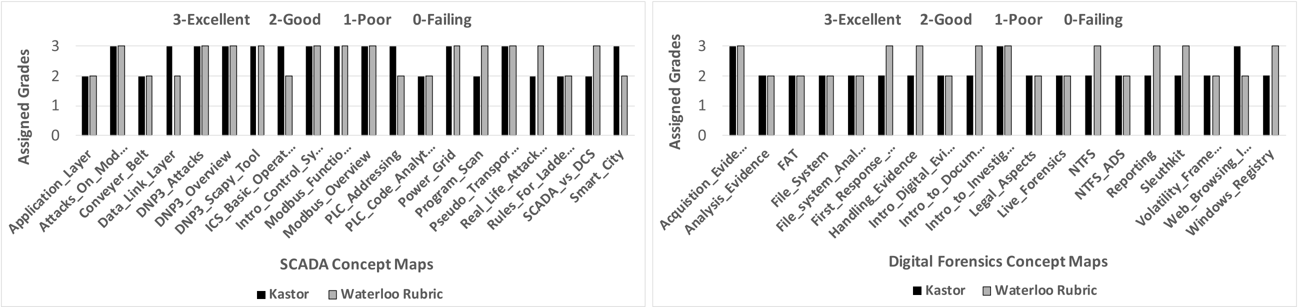
`Kastor` analyzes a concept map using four Waterloo Rubric parameters. Figures 3 and 4 summarize `Kastor`'s accuracy on the individual rubric parameters when compared to manual rubric analysis as ground-truth. The results are promising and show that `Kastor`'s accuracy at combined accurate and inaccurate levels is 100% on breadth of net, 95% on embeddedness, 100% on the use of descriptive links, and 90% on efficient links for SCADA concept maps. For forensics maps, the accuracy is 100%, 26.31%, 84.21%, and 89.47% respectively.

### 3) KASTOR ACCURACY ON CONCEPT MAP-LEVEL

Figures 5(a) and 5(b) present `Kastor`'s accuracy on the concept maps of SCADA, and Forensics respectively. `Kastor`'s accuracy is obtained by comparing `Kastor`'s scores with manual Waterloo Rubric scores. The evaluation results show that most scores reported by `Kastor` are accurate or close to accurate.

Table 6 summarizes the results. Specifically, `Kastor` achieves the accuracy of around 84% and 95% (at accurate and close-to-accurate levels) for SCADA and forensics concept maps respectively. We notice that the inaccuracies are due to the following reasons:

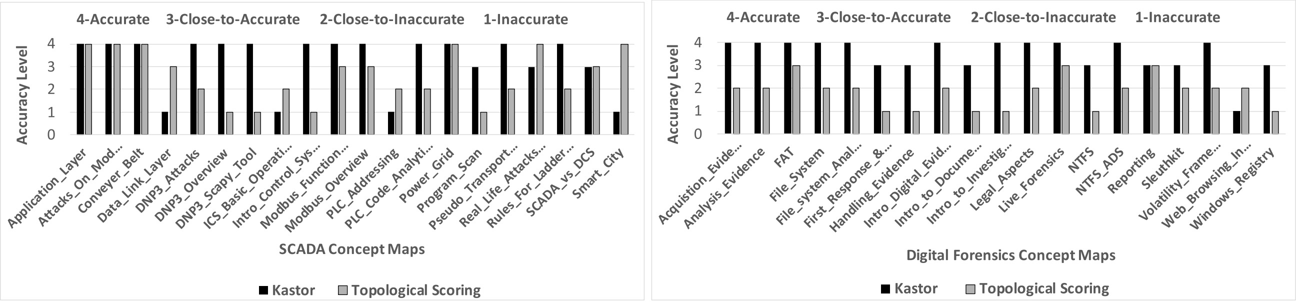
- *Broad source text* - If a source text is a chapter of a textbook and a concept map is of a particular topic within the chapter, the keyword matching doesn't work accurately hence, `Kastor` reports an inaccurate score. The results are better and closer to accurate for the Digital Forensics course as the source text used are the PowerPoint slides with specific slides segregated for each concept map. However, for the SCADA System course, chapters of the textbook are used as the source text. Since the book chapter is broader in scope as compared to the PowerPoint slides, it has more keywords which may not present in the corresponding concept map, hence resulting in a comparatively lower accuracy score.
- *Concept maps with directory signs (c:,d:)* - The directories are not recognised properly during keyword extraction by RAKE - when keywords are extracted by RAKE; words get split by joining words, so all directories are treated as individual characters instead of directories, which is error prone when comparing and gives an inaccurate score.



(a) SCADA System Security

(b) Digital Forensics

**FIGURE 5.** **Head-to-head grading comparison of Kastor with the ground-truth i.e., manual Waterloo Rubric analysis.**



(a) SCADA System Security

(b) Digital Forensics

**FIGURE 6.** **Accuracy comparison of Kastor with a prior state-of-the-art work of topological scoring.**

**TABLE 6.** **Comparing accuracy of Kastor (in percentage) with a prior state-of-the-art work of topological scoring for SCADA and digital forensics concept maps.**

| | SCADA System | | Digital Forensics | |
|---|---|---|---|---|
| **Accuracy Level** | **Topolo.** | **Kastor** | **Topolo.** | **Kastor** |
| Accurate (%) | 31.57 | 68.42 | 0 | 57.89 |
| Close to accurate (%) | 21.05 | 15.78 | 15.78 | 36.84 |
| Close to inaccurate (%) | 31.57 | 0 | 52.63 | 0 |
| Inaccurate (%) | 21.05 | 21.05 | 31.57 | 5.26 |

- *Concept maps with long descriptive nodes and links* - since text from nodes and links is directly extracted instead of extracting keywords, long descriptive nodes are treated as one long keyword. When comparing these with the keywords extracted from source text, they give an inaccurate overall score.

### C. KASTOR COMPARISON WITH PRIOR WORK

Topological scoring utilizes structural features of the concept map for grading and has been explored in the past. We compare Kastor with the topological scoring method, recently evaluated by Deshpande and Ahmed [3] to evaluate Kastor's efficacy on a state-of-the-art concept map grading method. We use the same set of concept maps used for topological scoring and since we already know the ground-truth (correct scores) of the concept maps in our datasets, we employ the topological scoring and obtain their accuracy levels for the concept maps to accurate, close-to-accurate, close-to-inaccurate, and inaccurate. Similarly, we obtain the Kastor's accuracy levels on the concept maps and perform head-to-head comparison as shown in Figures 6(a) and 6(b). Since the same set of concept maps are used in both, statistical analysis is provided to show the overall improvement in the accuracy of the evaluated scores.

The evaluation results in Table 6 show that Kastor provides more accurate scores as compared to the scores obtained by Topological scoring. Specifically, Kastor's comparison with a topological scoring method shows improvement by around 32% and 79% (at combined accurate and close-to-accurate levels) on SCADA and forensics concept maps respectively.

## VI. CONCLUSION

Concept mapping is a well-known, cognitively intensive pedagogical tool which helps students to organize, graphically represent, and develop a deep understanding of a topic or a concept. The grading of concept maps is typically manual and hence can be very tedious and time-consuming, especially for a large class. Existing research mainly focuses on topological scoring of concept maps which are not very effective as compared to the Waterloo Rubric, a well-defined grading rubric for manual analysis.

This paper proposed a new method, Kastor that automates the quality assessment parameters for concept maps in the Waterloo Rubric. Kastor automated this method by using keyword extraction and comparison techniques to obtain the accuracy of grading concept maps, similar to the ground truth. The effectiveness of the proposed method is evaluated using a publicly available dataset of 39 concept maps for two cybersecurity courses on digital forensics and SCADA systems. The evaluation results showed that most concept maps were graded with accurate or close to an accurate level. Kastor also outperformed from an existing state-of-the-art topological scoring method. The results are promising but most of our parameters for results depend on keyword extraction algorithm used in Kastor. As part of the future work, the authors will work on improving the existing results by using a concept map for comparing with other concept maps instead of a text source currently being used. This would remove the dependency on keyword extraction as all the words in that concept map will be treated as keywords.

## REFERENCES

[1] M. A. Ruiz-Primo and R. J. Shavelson, "Problems and issues in the use of concept maps in science assessment," *J. Res. Sci. Teach.*, vol. 33, no. 6, pp. 569–600, Aug. 1996.

[2] A. J. Ca nas, L. Bunch, J. D. Novak, and P. Reiska, "Cmapanalysis: An extensible concept map analysis tool," *J. Educ., Teac. Trainers*, vol. 4, no, 1, pp. 36–46, 2013.

[3] P. Deshpande and I. Ahmed, "Topological scoring of concept maps for cybersecurity education," in *Proc. 50th ACM Tech. Symp. Comput. Sci. Educ.*, New York, NY, USA, Feb. 2019, pp. 731–737, doi: 10.1145/3287324.3287495.

[4] University of Waterloo. (2016). *Rubric for Assessing Concept Maps Centre for Teaching Excellence*. [Online]. Available: https://uwaterloo.ca/centre-for-teaching-excellence/sites/ca.centre-for-teaching-excellence/files/uploads/files/rubric_for_assessing_concept_maps.pdf

[5] R. Mihalcea, C. Corley, and C. Strapparava, "Corpus-based and knowledge-based measures of text semantic similarity," in *Proc. AAAI*, 2006, pp. 775–780.

[6] P. Deshpande and I. Ahmed. (2018). *Concept Map Datasets for Cybersecurity Courses*. Accessed: Jul. 23, 2018. [Online]. Available: https://gitlab.com/iahmed4/concept-map-datasets-for-cybersecurity-courses

[7] (2020). *Script to Automate Waterloo*. [Online]. Available: https://github.com/shresht77/autograder-for-concept-maps

[8] I. Ahmed and V. Roussev, "Peer instruction teaching methodology for cybersecurity education," *IEEE Secur. Privacy*, vol. 16, no. 4, pp. 88–91, Jul. 2018.

[9] P. Deshpande, C. B. Lee, and I. Ahmed, "Evaluation of peer instruction for cybersecurity education," in *Proc. 50th ACM Tech. Symp. Comput. Sci. Educ.*, Feb. 2019, pp. 720–725.

[10] W. E. Johnson, A. Luzader, I. Ahmed, V. Roussev, G. G. Richard, and C. B. Lee, "Development of peer instruction questions for cybersecurity education," in *Proc. Workshop Adv. Secur. Educ.*, 2016, pp. 1–9.

[11] W. Johnson, I. Ahmed, V. Roussev, and C. B. Lee, "Peer instruction for digital forensics," in *Proc. Workshop Adv. Secur. Educ.*, 2017, pp. 1–9

[12] J. D. Novak and D. B. Gowin, *Learning How to Learning*. Cambridge, U.K.: Cambridge Univ. Press, 1984.

[13] B. A. Beyerbach and J. M. Smith, "Using a computerized concept mapping program to assess preservice teachers' thinking about effective teaching," *J. Res. Sci. Teaching*, vol. 27, no. 10, pp. 961–971, Dec. 1990.

[14] J. H. Dexter, "The cyber security management system: A conceptual mapping," SANS Inst., Bethesda, MD, USA, Tech. Rep. John_Dexter_GSEC.doc, 2002. [Online]. Available: https://sansorg.egnyte.com/dl/C5LbpMRzt7

[15] A. Tanner and D. Dampier, "Concept mapping for digital forensic investigations," in *Advances in Digital Forensics V*, G. Peterson and S. Shenoi, Eds. Berlin, Germany: Springer, 2009, pp. 291–300.

[16] D. Hay, I. Kinchin, and S. Lygo-Baker, "Making learning visible: The role of concept mapping in higher education," *Stud. Higher Educ.*, vol. 33, no. 3, pp. 295–311, Jun. 2008.

[17] A. Anohina-Naumeca and J. Grundspenkis, "Scoring concept maps: An overview," *Proc. Int. Conf. Comput. Syst.*, Jan. 2009, p. 78.

[18] B. E. Cline, C. C. Brewster, and R. D. Fell, "A rule-based system for automatically evaluating Student concept maps," *Expert Syst. Appl.*, vol. 37, no. 3, pp. 2282–2291, Mar. 2010. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S095741740900712X

[19] J. W. Douglas Luckie, S. Harrison, and D. Ebert-May, "Studying C-TOOLS: Automated grading for online concept maps," in *Proc. Conf. Conceptual Assessment Biol.*, 2008, pp. 1–13.

[20] S. H. Harrison, J. L. Wallace, D. Ebert-May, and D. B. Luckie, "C-TOOLS automated grading for online concept maps works well with a little help from' WordNet," in *Proc. 1st Int. Conf. Concept Mapping, Concept Maps, Theory, Methodol., Technol.*, vol. 2, 2004, pp. 211–214.

[21] A. Pinandito, D. D. Prasetya, Y. Hayashi, and T. Hirashima, "Design and development of semi-automatic concept map authoring support tool," *Res. Pract. Technol. Enhanced Learn.*, vol. 16, no. 1, pp. 1–19, Dec. 2021.

[22] J. A. Rye and P. A. Rubba, "Scoring concept maps: An expert map-based scheme weighted for relationships," *School Sci. Math.*, vol. 102, no. 1, pp. 33–44, Jan. 2002. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1949-8594.2002.tb18194.x

[23] S. Rose, D. Engel, N. Cramer, and W. Cowley, *Automatic Keyword Extraction From Individual Documents*. Berlin, Germany: Springer, 2010, pp. 1–20.

[24] T. Atapattu, K. Falkner, and N. Falkner, "Automated extraction of semantic concepts from semi-structured data: Supporting computer-based education through the analysis of lecture notes," in *Database and Expert Systems Applications*, S. W. Liddle, K.-D. Schewe, A. M. Tjoa, and X. Zhou, Eds. Berlin, Germany: Springer, 2012, pp. 161–175.

[25] V. Roussev, "Digital forensic science: Issues, methods, and challenges," *Synth. Lectures Inf. Secur., Privacy, Trust*, vol. 8, no. 5, pp. 1–155, Dec. 2016.

**SHRESHT BHATIA** received the bachelor's degree from Manipal University, India, in 2020. He is currently pursuing the Master of Science degree in cybersecurity with New York University (NYU). His research interest includes combining machine learning concepts with cybersecurity.

**SAJAL BHATIA** received the bachelor's degree in communication and computer engineering from the LNM Institute of Information Technology, India, and the Ph.D. degree from the Queensland University of Technology, Australia. He is currently an Assistant Professor in cybersecurity with the School of Computer Science and Engineering, Sacred Heart University (SHU). He is also the Director of Cybersecurity Program at SHU. His research interests include the area of cybersecurity, with a focus on Denial-of-Service (DoS) attacks, synthetic traffic generation, critical infrastructure security, intrusion detection, industrial control system security, cyber-physical system security, and cybersecurity education.

**IRFAN AHMED** (Senior Member, IEEE) is currently an Associate Professor in computer science with Virginia Commonwealth University (VCU). He is also the Director of the Security and Forensics Engineering (SAFE) Laboratory, and a Faculty Fellow of the VCU Cybersecurity Center. His research interests include the area of cybersecurity, currently focusing on digital forensics, malware, cyber-physical systems, and cybersecurity education.