SOA Research INSTITUTE

CyLit: An NLP-Powered Repository and Search Tool for Cyber Risk Literature

APRIL | 2023

CyLit: An NLP-Powered Repository and Search Tool for Cyber Risk Literature

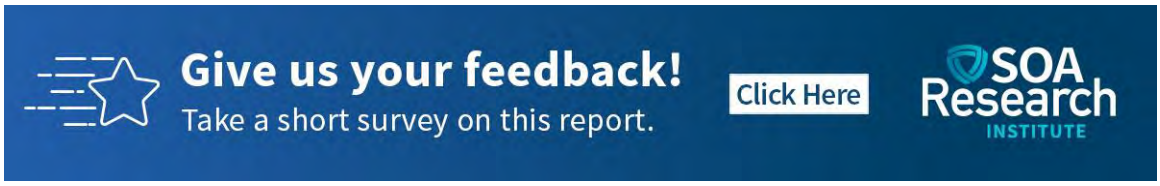
AUTHORS Zhiyu Quan, PhD

Linfeng Zhang

Wing Fung Chong, ASA, PhD

Runhuan Feng, FSA, CERA, PhD

SPONSOR General Insurance Research Committee



CONTENTS

CyLit: An NLP-Powered Repository and Search Tool for Cyber Risk Literature

Executive Summary

With the large and growing body of cyber risk literature, we observe three major challenges faced by the actuarial research community: there is no context-aware tool for finding cyber literature, no central repository of cyber risk resources, and a lack of integration of literature on the trends. To address the challenges, we built CyLit, a repository of cyber risk articles, with a Natural Language Processing (NLP)-powered search tool that can easily be used by researchers to find relevant materials.

Note: CyLit can be accessed via the link which will be published on the Society of Actuaries web page which contains this report.

Section 1: Introduction

1.1 MOTIVATIONS

The volume of cyber risk literature is growing on a daily basis, owing to the rising awareness of cyber risk and cybersecurity. Cyber risk is an interdisciplinary issue and can be approached from various perspectives, such as monetary losses and legal consequences caused by cyber incidents, techniques for improving cybersecurity, etc. (Eling, 2020) demonstrated the broad range of topics that the cyber risk literature covers and proposed ten categories of academic articles in this field based on the disciplines involved, including management, economics, and telecommunications.

(Eling, 2020) is a survey paper that offers a decent overview of cyber-related literature and summarizes some key studies, including their research questions and methodologies. However, constrained by the capacity to collect and review articles manually, the number of papers examined in this survey paper is 217, which is a rather small portion of all cyber-related papers. In comparison, as of October 10, 2022, querying the Scopus[1] database resulted in around 30,000 articles related to cyber risk, and the number is still growing.

To complement the limitations posed by the static nature of current survey papers in cyber risk literature and to facilitate cyber risk research, we build CyLit, an NLP-powered repository and search tool for cyber risk literature. Natural Language Processing (NLP) techniques empower this system through the following aspects:

- Improved accuracy and efficiency: NLP can help automate processes, such as clustering relevant articles, in addition, accurately interpreting and responding to users' search inquiries, saving time for users, and mitigating human search errors.
- Better insights: NLP can help extract valuable insights from large amounts of unstructured text data related to cyber.
- Enhanced summarization: NLP can generate summaries for cyber-related articles and identify key themes and topics for users.

This report documents the current state of this tool, including its key functions, as well as the methods and technologies under the hood.

1.2 PROJECT INSPIRATION

The premise of CyLit was inspired by the Explainable Artificial Intelligence (XAI) for the NLP project (see also (Danilevsky et al., 2020)). The researchers collected a substantial number of articles and compiled a database with different ordering systems. Although this process was performed manually with human reviewers, we tend to create a similar system with NLP techniques, thus allowing us to process larger quantities of data.

1.3 CONTRIBUTIONS

The contribution of this project to the cyber risk research community is three-fold. Firstly, traditional research tools, such as Google Scholar, excel at finding research papers with keyword-based approaches.

[1] https://www.scopus.com

However, their lack of contextual awareness can lead to results of little relevance, especially with respect to interdisciplinary subjects, such as cyber risk. CyLit addresses this issue by providing a context-aware tool for finding cyber risk resources.

Second, while large literature databases, such as Scopus, exist, the generality of subjects of these collections does not allow easy access to information related to cyber risk. In addition, articles about cyber risk may be scattered around multiple databases, such as Scopus and arXiv[2]. CyLit provides a centralized and self-updating repository for cyber risk literature.

Finally, there is a lack of integration of the trends in cyber risk research. The relevancy of research topics on cyber risk evolves over time with people's changing perceptions of cyber risk and the development of the cyber insurance industry. The timeliness of the literature and trends in cyber risk research is hardly captured by existing research tools. CyLit aims to capture the history and trends of cyber risk research, and this helps researchers determine the temporal relevance of research questions.

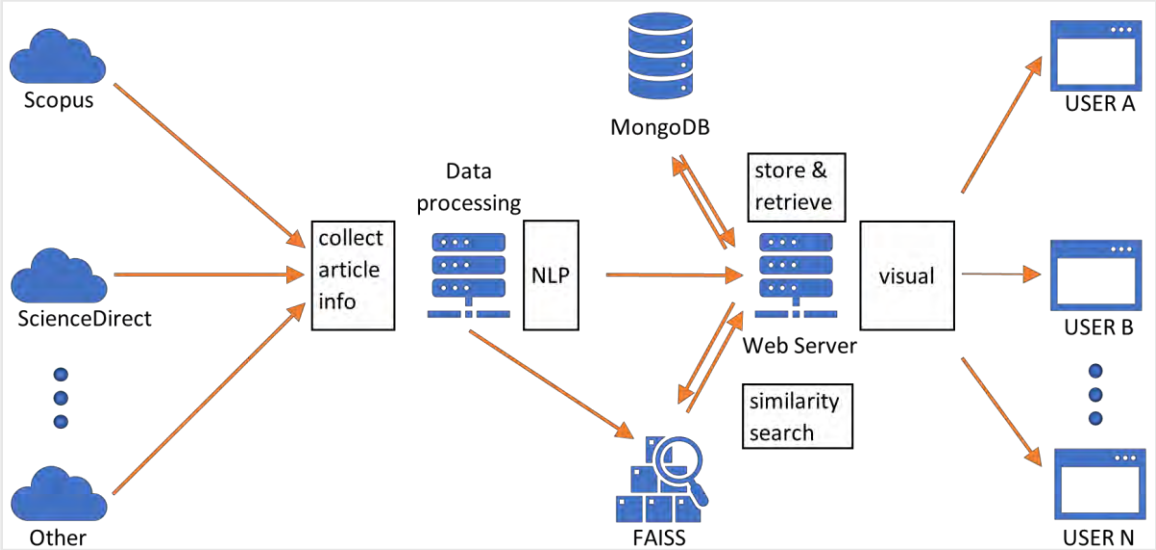
[2] https://arxiv.org/

Section 2: System Architecture

2.1 ARCHITECTURE OVERVIEW

The architecture of CyLit can be summarized and visualized in Figure 1.

Figure 1
CYLIT SYSTEM ARCHITECTURE DIAGRAM



Generally, the system is composed of a data collection unit, an NLP unit, a document-oriented database, an indexing system, and a web server. Each of the components is described in the following sections.

2.2 DATA COLLECTION UNIT

With the goal of building a repository of cyber risk literature, there are generally two approaches that can be taken to get the articles and feed them into the repository. One is scraping web content that includes articles about cyber risk, and the other is using literature database Application Programming Interfaces (APIs). The former approach requires the identification of a set of websites that stably publish articles, such as news media sites and blogs, and that allow their contents to be collected by automated tools. There are plenty of resources that meet the first criterion (see, for example, Cyware[3]), but few have the generosity to allow scraping. Therefore, this approach is saved for future research when the diversity of sources becomes the main priority. The latter approach, using literature database APIs, is taken. This approach guarantees that a large collection of academic papers can be obtained, which helps us quickly expand the volume of the repository at the early stage of its development.

[3] https://cyware.com/cyber-security-news-articles

Currently, the data collection unit mainly collects cyber risk literature from ScienceDirect[4] and Scopus using their APIs. Several other sources have been investigated, including arXiv, Google Scholar[5], CORE[6], and are planned to be included in the resources pool for future data collection. The case-insensitive search query sent over to Scopus and ScienceDirect is as follows.

```
"Cybersecurity" OR "Cyberrisk"
OR "Cyber security" OR "Cyber risk"
OR "Cyber literature" OR "Cyber insurance".
```

Information about papers that match this query in the title, abstract, or keyword lists is retrieved and archived. The information of each article collected contains:

- Identifiers, such as its identifier in Scopus and its Digital Object Identifier (DOI), which is its unique identifier in the public domain;
- Document metadata, including article information, author information, publication information, etc. Some selected metadata are presented in Table 1, and a complete list of attributes is outlined in Appendix A.

Table 1
SELECTED METADATA IN THE ARTICLE

Metadata ID	Description
title	The title of the article
subtypeDescription	Type of article
authorNames	Names of the authors
description	The abstract of the article
authKeywords	Author-provided keywords
coverDate	Publishing date

Due to the complications surrounding text licensing and access rules, the full text is not retrieved or stored. The DOI and the link to the publisher's site are retained for each article, and they are made available to users who need to locate the full text.

This unit also performs some simple data processing to ensure that the collected data and the existing data in our repository are consistent in format. For example, some attributes of a newly collected article shall be renamed or changed to a different data type to match the existing data. In addition, this unit also checks for duplication of articles. If it collects an article that has the same information as an existing article in our repository, then the newly collected article is discarded.

This duplication check is also useful for scheduled monthly data collection. To ensure the relevancy and timeliness of the articles in the repository, cron jobs are scheduled in the data collection unit[7]. CRON is a Linux utility that functions as a job scheduler, allowing us to schedule the execution of article-fetching and article-processing scripts monthly. To avoid going through everything in external literature sources like Scopus over and over to get just a few hundred new records each time, the current implementation of the

[4] https://www.sciencedirect.com
[5] https://scholar.google.com
[6] https://core.ac.uk
[7] See, for example, https://www.redhat.com/sysadmin/linux-cron-command

data collection unit sorts the results returned by those external sources by their publication dates in descending order, and once the duplication check suggests that the collected information starts to collide with the existing information in our database, the data collection process is terminated, so that no more duplicated information will be collected.

2.3 NLP UNIT

After the information about the collected article is preprocessed in the data collection unit, it is then handled by the NLP unit. The main piece of information that will be used for NLP is the abstract of each article, which will go through the process of keyword extraction and clustering. This process shall be elaborated on later in Section 3.2.

2.4 DATABASE SYSTEM AND INDEXING SYSTEM

The storage and retrieval of data is a critical component of CyLit. To facilitate this, a database system and an indexing system powered by MongoDB and Facebook AI Similarity Search (FAISS), respectively, are implemented and operate jointly to achieve the goal.

The responsibility of the database system is two-fold. After the article information is collected from literature sources and has the NLP-generated information, including extracted keywords and assigned clusters, attached, it is added to the MongoDB database for storage. The database also responds to definite queries, such as retrieving article information by publication type.

In contrast, the meanings of queries in a semantic fashion, such as “the cost of a data breach", are less clear to the database system, and thus are interpreted by the FAISS indexing system; see Section 3.5 for more discussions. After collecting the abstract of an article, its sentence embedding vector generated by the NLP unit will be added to the indexing system. In the event of a semantic search, FAISS performs a similarity comparison between the search query and the sentence embeddings in the system and returns the most relevant results. This shall be elaborated in Section 3.5.

2.5 WEB SERVER

A web application is built to make the repository accessible from any device anywhere. We take a standard web development approach to create the backend and the frontend of the application. The backend is realized in Python using the Django framework on the basis of all the NLP models being implemented in Python. For the frontend, we swapped out the Jinja template, which is built in the Django framework, and replaced it with React, which is a frontend library that has better support and scalability.

The functionality of the web application includes article lookup via filtering and semantic search, as well as some visualizations that provide an overview of the cyber risk literature from different perspectives.

Section 3: Methods

3.1 THE LOGIC BEHIND SYSTEM DESIGN

To get an impression of a paper, we take the following two steps: first, we look at the title and keywords; second, we read the abstract.

A compelling title for an academic research paper should have several characteristics, including indicating the topic, purpose, and scope of the research, using the current nomenclature in the field of study, and accurately capturing what has been done. However, we often observe titles with abbreviations, even in the form of questions or words that create a positive impression and intrigue readers. This can cause problems when performing NLP techniques to capture useful information from the title. For example, the title of one of the most popular papers in NLP is “Attention Is All You Need”. This is a very attractive title to the reader, but from a text-mining perspective, only the word “Attention", a nomenclature that differs from actual English, matters when summarizing such a paper based on the title.

Keywords are words that capture the essence of the paper. These words make the paper searchable, and the authors use the most relevant keywords to help others find the paper. In general, keywords should be phrases between 3-5 words. Fortunately, this will help us with our task, since each paper can be summarized in 3-5 keywords. Often, similar papers also have keywords that overlap.

The abstract, usually in a paragraph of no more than 350 words, summarizes the main aspects of the entire paper in the prescribed sequence that includes: the overall purpose of the study and the research problem; the basic design of the study; major findings or trends found in the results of the analysis; and, a brief summary of conclusions. The writing style of abstract varies by discipline, especially for interdisciplinary research like cyber risk, which presents challenges even for humans to summarize and compare abstracts to find similar papers. The abstract may still be useful because it allows the author to elaborate on the main aspects of the paper, which can be more informative than keywords.

To summarize and cluster papers, the aforementioned key components of the paper are taken into account. Based on keyword information, we create a clustered index for a repository of cyber risk articles. When keywords are not available, we incorporate title and abstract information to create “keywords” for the article or paper. In addition, the search engine finds the summarized information that best matches the query information from the key components.

3.2 NATURAL LANGUAGE PROCESSING TECHNIQUES

NLP is a valuable tool that can be applied to turn raw unstructured text information into structured data for analysis. More specifically, one of the most critical steps in NLP is that we should convert the text into a numerical representation that the computer can recognize. The term word embedding refers to representing words for text analysis in the form of a real-valued vector that encodes the meaning of the words such that the words closer in the vector space are expected to be similar in meaning.

Word embeddings can be obtained using a set of NLP techniques where words or phrases from the vocabulary are mapped to vectors of real numbers. Our techniques mainly focus on exploiting the mathematical properties of word embeddings and how they interact in an n-dimensional vector space. In this study, we investigate several different methods to generate word embeddings: Term Frequency Inverse Document Frequency (TF-IDF), Word2vec (Mikolov et al., 2013a; Mikolov et al., 2013b), Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2018), and derivations of BERT, Sentence- BERT (Reimers and Gurevych, 2019) as well as KeyBERT (Grootendorst, 2020).

The TF-IDF is a numerical statistic that measures the relevancy of a word in relation to a collection of documents. The term frequency is the number of times a word appears in a document. Meanwhile, the inverse document frequency is the frequency of a word in the collection of documents. The TF-IDF value increases proportionally to the number of times a word appears in the document and is offset by the number of documents in the corpus that contain the word, which helps adjust for the disruption caused by some words appearing more frequently in general.

TF-IDF has become one of the most popular term-weighting schemes and can be used to generate basic summary statistics to find important keywords among the repository of cyber risk articles. However, TF-IDF is insufficient for our task because as the number of documents grows, the size of the embeddings grows exponentially along with it, causing a loss of information and an increase in noise within the data. Because of this, a bag-of-words (BoW) approach may not be the best fit for a large corpus.

To circumvent this problem, we investigate more sophisticated word embeddings. Word2vec is a collection of related models that uses shallow neural networks to generate word embeddings and learn word associations from a large corpus of text. In Word2vec, linguistic contexts are reconstructed using either continuous bag-of-words (CBOW) or continuous skip-gram architectures. In the CBOW architecture, the model predicts the current (middle) word from the window of the surrounding context words. The context consists of a few words be- fore and after the current (middle) word. In the continuous skip-gram architecture, the model uses the current word to predict the surrounding window of context words. In other words, it predicts words within a certain range before and after the current word in the same sentence. Through the large corpus of linguistic context reconstruction (model training) process, Word2vec represents each distinct word with vectors, typically of several hundred dimensions, which capture words' semantic and syntactic qualities. Ideally, word vectors are positioned in the vector space such that words that share common contexts in the corpus, i.e., semantically and syntactically similar, are located close to one another. More dissimilar words are located farther from one another in the space. Hence, the degree of semantic and syntactic similarity between words represented by vectors can be measured by a simple mathematical function, e.g., cosine similarity.

However, Word2vec may be suboptimal since it relies on local information. In other words, since the semantic and syntactic representation of a word relies only on its neighbors, it cannot comprehend words under the big picture of the document. This phenomenon is not suitable for academic papers or scientific articles. In addition, it does not solve the problem of polysemous words since words and vectors have a one-to-one relationship. Moreover, using pre-trained models that are not specific to the domain can also cause inaccurate results. In our case, a pre-trained Word2vec model using the Google News dataset does not generalize well to the cyber risk domain. Word2vec cannot handle out-of-vocabulary words well, as it generates word embeddings based on what it has studied. Out-of-vocabulary words receive a random vector representation.

BERT is designed to pre-train deep (learning) bidirectional representations from the unlabeled text by jointly conditioning on both left and right contexts in all layers. This bidirectional approach allows a model to learn deeper context than traditional directional approaches. BERT is pre-trained on two tasks: language modeling and next-sentence pre-diction. In language modeling, BERT was trained to predict that 15% of randomly masked words were from the context. In next-sentence prediction, BERT is trained to predict whether a chosen next sentence is probable or not given the first sentence since language modeling does not directly capture the relationship between two sentences. As a result of the training process, BERT learns contextual embeddings for words, and the pre-trained BERT model can be fine-tuned with just one additional output layer to create state-of-the-art models for a wide range of downstream tasks.

Word2vec generates a single-word embedding representation for each word in the corpus. For example, the vector for "Attention" will have the exact Word2vec vector representation for both of its circumstances in the sentences "Attention Is All You Need" and "Please Pay Attention". On the other hand, BERT provides a contextualized embedding that is different according to the sentence. Since the nature of a sequential input, BERT takes into account the context for each occurrence of the given word and allows the word embeddings to store contextual information. BERT is also able to address the out-of-vocabulary issue. BERT learns at the sub-word level, which gives members of the BERT family a smaller vocabulary than the initial training data. Because of this, BERT can generate out-of-vocabulary words, giving it an expansive vocabulary. Therefore, BERT is more suitable for our purpose compared to Word2vec.

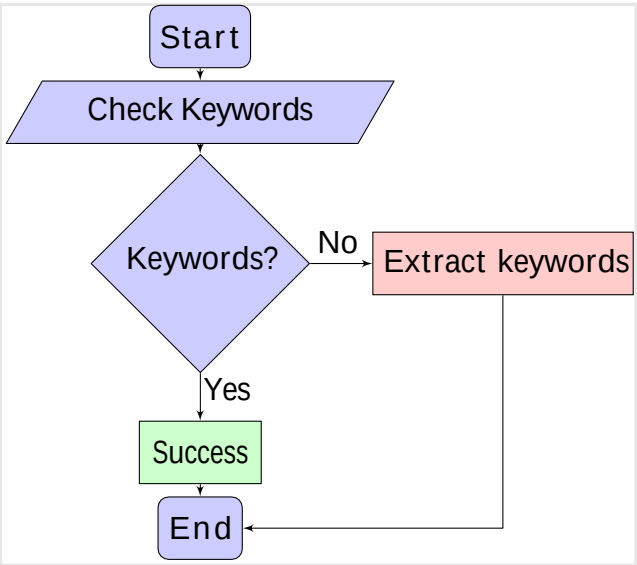
With an increasing collection of documents, time and calculations become exponentially more involved. On our website, we need to perform massive semantic text searches and similarity clustering. However, we notice that traditional BERT has computational overhead, which leads us to find an alternative model. Sentence-BERT is a modification of the pre-trained BERT to derive semantically meaningful sentence embeddings that can be compared using cosine similarity. It significantly reduces computation time for finding the most similar in an extensive collection of sentences. Because of the size of the current document corpus and the anticipated growth in the future, Sentence-BERT was the most suitable approach in the BERT family.

KeyBERT is a keyword extraction technique that leverages BERT embeddings to create keywords and key phrases that are most similar (cosine similarity) to a document. Specifically, document embeddings are extracted using BERT to obtain document-level representations. Then, word embeddings are extracted for N-gram words/phrases. Finally, we use cosine similarity to find the most similar word/phrase to the document. The most similar words/phrases can then be identified as the words that best describe the entire document.

3.3 KEYWORD EXTRACTION AND CLUSTERING

To observe trends in the document collection as well as generate more relevant results in search and query, we utilize `authKeywords` provided to us in the metadata. In order to standardize the provided keywords, we performed a series of pre-processing procedures. Specifically, we remove punctuation and special characters, convert all keywords to lowercase, and eliminate duplicates by removing blank spaces between some key phrases (e.g., converting "cyber security" to "cybersecurity") and converting some abbreviations to their corresponding full names (e.g., converting "iiot" to "industrial internet of things"). Pre-processing procedures are essential for effective text analysis and NLP, as they help ensure that the data is clean, consistent, and ready for analysis. Following the cleaning process, we obtain a total of 38,043 keywords and key phrases, which we consider as a comprehensive and representative keyword library for the cyber risk literature.

Figure 2
KEYWORD PIPELINE FOR ARTICLES



The two cases discussed in this section are when authors provide keywords and when they do not. They are treated differently as illustrated in Figure 2. When the keywords are provided, we are able to generate word embeddings for `authKeywords` using the pre-trained models. However, we need to extract our own keywords using KeyBERT from the document for cases where the keywords are absent. Users must define a range of `n_grams` from which KeyBERT will select. Because the user does not know the optimal range of keyphrase lengths, some experimenting will be required. In addition, the sliding window nature of the algorithm does not allow for capturing grammatical structures, leading to some grammatically incorrect answers. To address these issues, we use the methodology used in KeyphraseVectorizers. This extracts phrases with the appropriate part-of-speech (POS) patterns from a corpus before converting it into a document-keyphrase matrix. We use this in conjunction with KeyBERT to generate syntactically correct keyphrases for our use case. First, the vectorizer generates keyphrases for consideration which are then ranked by KeyBERT. The `top_n` keywords are then returned. To ensure a diverse selection of keywords, we also calculated Maximal Marginal Relevance (MMR) between candidate keywords and the abstract to select keywords that strike an optimal balance between diversity and relevance to the abstract. The hyperparameters, such as diversity, are chosen by evaluating the extracted keywords on papers where `authKeywords` are provided. We determined the final set of extracted keywords by identifying the intersection between our extracted keywords and the preprocessed `authKeywords` library. Through this method, we are able to generate keywords that capture the semantic meaning as well as represent the article in an appropriate fashion.

To generate word embeddings for keywords, we utilize the `all-MiniLM-L6-v2` model, a pretrained sentence-transformers model based on the BERT architecture. The model maps our keywords to a 384-dimensional vector space, providing numerical representations for the clustering task. We employ multiple clustering techniques for keyword clustering, including *K*-Means, DBSCAN (Density-Based Spatial Clustering of Applications with Noise), and HDBSCAN (Hierarchical Density-Based Spatial Clustering of Applications with Noise). For DBSCAN and HDBSCAN, we utilize both cosine similarity and Spearman correlation as dissimilarity metrics. After analyzing the clustering results using the techniques mentioned earlier in this section, we chose to apply *K*-means clustering to all the keyword and keyphrase embeddings, resulting in 30 keyword clusters.

To visually represent the keyword clusters, we create word clouds for each cluster. The size of each word corresponds to its distance from the centroid of its respective cluster. Specifically, the size of a word is larger if it is closer to the centroid and vice versa. For a fair comparison across clusters, we standardize the distances within each cluster using the maximum distance observed in the cluster. Figure 3 presents the word cloud plot for keyword cluster 1. We can see that the central keyword for cluster 1 is "risk detection". Addition- ally, we observe that keywords closely located to the centroid include "risk identification", "vulnerability detection", "threat detection", "threat analysis", and "threat assessment", suggesting that the cluster primarily focuses on the detection of risks and threats. Therefore, we label this cluster as "detection". For additional information on the keyword clusters, including cluster names, the number of keywords in each cluster, and the number of papers related to keywords in each cluster, see Appendix B.

Figure 3
WORD CLOUD FOR KEYWORD CLUSTER 1

3.4 ASSOCIATION ANALYSIS

Some topics defined by the clusters appear simultaneously in papers with high frequency, suggesting some association among these topics. For example, the search query "Security issues in cyber-physical systems" in CyLit returns a number of articles, such as (Agrawal and Kumar, 2022), (Dsouza et al., 2019), and (Bou-Harb, 2016), many of which have keywords in both C8 (System Security) and C15 (Cyberphysical Devices).

To uncover the potential cross-topic research activity, we investigate the association patterns among keyword clusters. We conduct association analysis using the Apriori algorithm to investigate the connection among keyword clusters in papers. The Apriori algorithm is one of the most widely used algorithms for association rule mining, which uses the metrics `support`, `confidence`, and `lift` to measure the strength of the relationships between items. Specifically, support measures the frequency of an itemset in the dataset, confidence measures the conditional probability of an itemset given another itemset, and lift measures the degree to which the occurrence of one itemset is dependent on the occurrence of another itemset. Table 2 shows the result from the Apriori analysis, where the association rules have been filtered based on multiple criteria, that is, `support` ≥ 0.05, `confidence` ≥ 0.5, and `lift` ≥ 1.5.

Table 2
ASSOCIATION RULES FROM APRIORI ANALYSIS

antecedents	consequents	antecedent support	consequent support	support	confidence	lift
C3	C16	0.230	0.352	0.132	0.573	1.630
(C3, C8)	C16	0.096	0.352	0.065	0.674	1.915
(C3, C11)	C16	0.068	0.352	0.051	0.745	2.119
(C3, C19)	C16	0.072	0.352	0.051	0.705	2.004
(C3, C29)	C16	0.135	0.352	0.073	0.540	1.535
(C11, C8)	C16	0.084	0.352	0.055	0.657	1.867
(C11, C16)	C8	0.109	0.327	0.055	0.502	1.539
(C15, C8)	C16	0.097	0.352	0.058	0.597	1.698
(C15, C16)	C8	0.101	0.327	0.058	0.576	1.765
(C19, C8)	C16	0.084	0.352	0.050	0.598	1.699
(C23, C8)	C16	0.085	0.352	0.051	0.594	1.688

Figure 4
PCA PROJECTION OF CLUSTER CENTROIDS WITH ASSOCIATIONS AMONG CLUSTERS

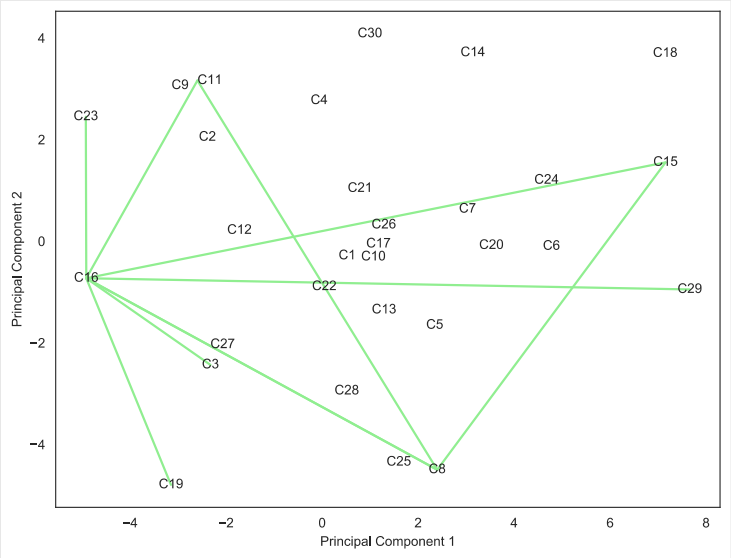


Figure 4 presents the semantic relationships and associations obtained from the Apriori association analysis among the keyword clusters. We employ Principal Component Analysis (PCA) to project the centroids of the keyword clusters onto a two-dimensional plane, with the node size indicating the number of keywords in each cluster and edges connecting nodes indicating the associations between clusters presented in Table 2. The proximity of the nodes (centroids of keyword clusters) suggests their semantic similarity; for example, clusters C3 (Cyber System Management) and C27 (Assessment) are close to each other, as are C14 (Malware) and C30 (Cyber Attack). Interestingly, association rules from the Apriori association analysis reveal connections among clusters that are not always semantically close. For instance, the antecedents C15 (Cyberphysical Devices) and C16 (Miscellaneous Terms) are associated with the consequent C8 (System Security) with a confidence of 0.576, indicating that the probability of keywords from C8 co-occurring with those from C15 and C16 is 0.576 in a paper. The lift of 1.765 also indicates a high positive correlation between the occurrence of keywords from C15 and C16 and the occurrence of keywords from C8.

3.5 SEMANTIC SEARCH

Semantic search is a query that carries meaning, unlike lexicographical search, where the search method looks for exact matches of the query. By doing so, semantic search attempts to improve search by taking into account the searcher's intention and context of the query. In our case, we are able to use semantic search to bring up relevant articles and pieces of literature. Since we can use BERT-based models to represent sections of text into vectors, we can perform similarity metrics to find pieces of text, which in this case are the abstracts of articles, that are semantically similar to the query phrase or sentence.

However, there is one problem, which is handling an increasing amount of text in the database. As the number of articles in the database continually increases, we would have to perform more and more pairwise similarity comparisons across the whole database. Furthermore, the number of comparison tasks also increases with the number of concurrent users performing semantic searches. Two problems might arise when the system tries to perform similarity searches on such a large scale. One is that some conventional approaches may require the entire set of vectors to be loaded into the system memory, which may be insufficient when the dataset size is large. The other one is that keeping the runtime down and returning the search results in a timely manner can be a great challenge. To get around these problems, we used FAISS to index the text vectors.

FAISS is a library that supports efficient similarity searches across large collections of vectors. It keeps an index file on the hard disk and uses it to build a data structure in the computer memory, which is much smaller in size. This addresses the memory insufficiency issue. When similarity searches are performed, this data structure is then used for calculating the Euclidean distance between the query vector and each of the vector representations of article abstracts. However, FAISS allows the calculation to be imprecise with a small likelihood in order to massively reduce both the time and the memory needed for computations. Finally, articles in which the embedding vectors have the smallest distances to the query vector are regarded as most relevant and are returned as results.

Section 4: Usage

CyLit has two main features, including semantic search and visualizations. These features are accessible through the navigation bar, as shown in Figure 5, at the top of the page.

Figure 5
NAVIGATION BAR OF CYLIT

4.1 SEMANTIC SEARCH

The search box on the right side of the navigation bar takes user input, and there is no restriction on the search query format i.e., it can be words, phrases, or sentences. In addition, the user can specify the type of documents to retrieve, e.g., journal articles, or books, by using the *document type* dropdown menu next to the search box.

Assuming that the user is interested in reading journal articles related to the cost of data breaches, the search query could be *how much does a data breach cost?*, and the choice of document type selected from the dropdown menu is *Article*. Then, pressing the search button next to it will send the query to the server. Then the server will send back the corresponding results, including the articles found and the visual representations of some statistics related to those articles.

By default, the visualizations are displayed first. The user can press the Articles button on the navigation bar to see the returned articles. The articles are listed according to their relevance to the search query. The most relevant one is shown at the top of the list. The user can click on each article in the results list to see more details, such as the citation information and the associated clusters.

4.2 VISUALIZATIONS

A key objective of this tool is to offer an overview of the cyber risk literature, and this is achieved through the visualizations of the aggregate data with respect to the literature.

When the tool is loaded without a specific search query, the aggregate information of all articles in the repository is displayed in the *Number of Articles per Year by Cluster* bar plot. In addition, users can filter the information by year or cluster to see what topics are written about in a given year and how the number of articles in a cluster evolves over time, as shown in Figure 6.

If a search query is provided, the graphs related to the search results can be viewed by clicking the *Data Visualization* tab on the navigation bar. The plots exhibit the number of articles addressing the search query in different years and in different clusters, as shown in Figure 7.

Figure 6

NUMBER OF ARTICLES BY CLUSTERS IN A YEAR (LEFT) AND NUMBER OF ARTICLES IN A CLUSTER OVER TIME (RIGHT)

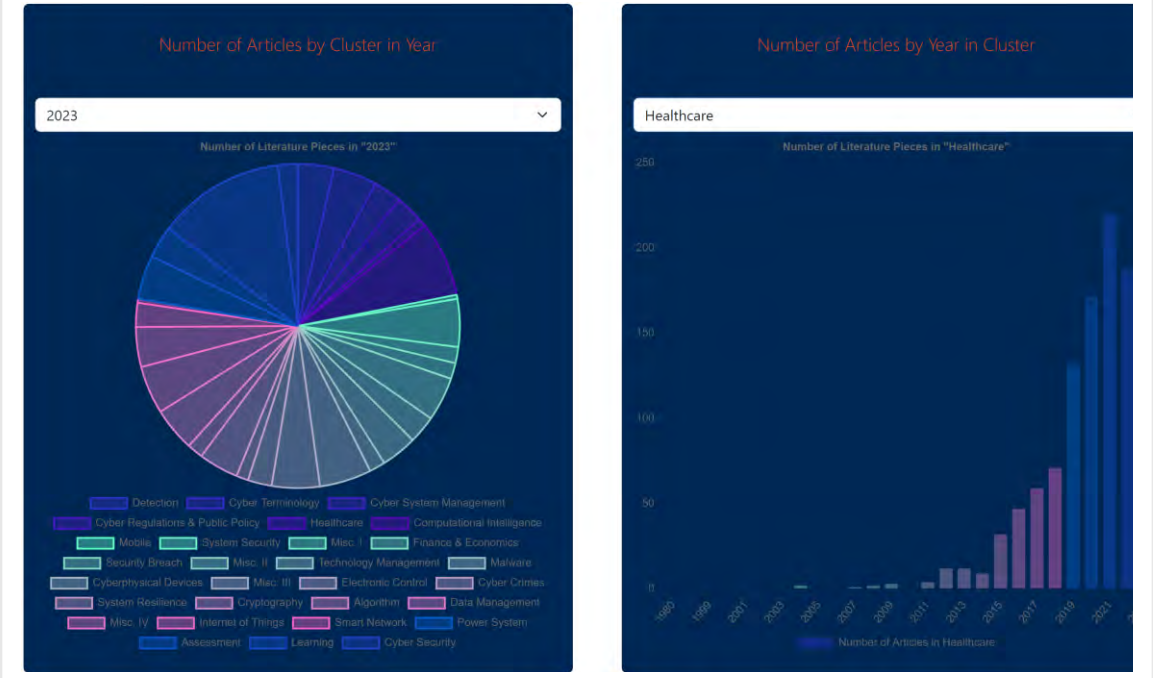
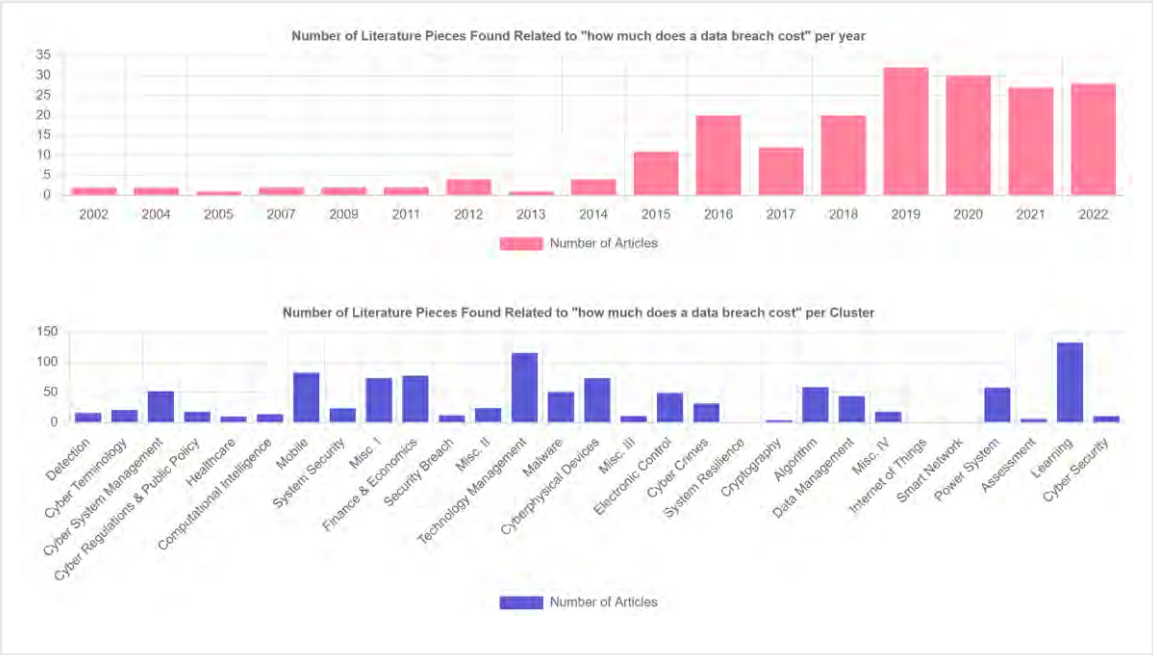


Figure 7

NUMBER OF ARTICLES RELATED TO THE SEARCH QUERY IN DIFFERENT YEARS AND CLUSTERS



Section 5: Conclusion and Future Directions

The interest in cyber risk research is on the rise, but with a growing volume of literature in this field and the interdisciplinary nature of this topic, it becomes difficult for researchers to find the information that is most helpful to their research questions. In this project, we built CyLit, an NLP-powered repository and search tool for cyber risk literature. The repository is self-updating, thus staying relevant to the latest topics in the field of cyber risk. NLP techniques are adopted to extract keywords from article abstracts, organize articles into clusters, and enable semantic searches. Furthermore, the repository is equipped with a web application, which makes querying the repository effortless. All these features allow cyber risk researchers to locate the needed information efficiently.

Given the current state of CyLit, the following developments are proposed as our future directions.

To keep the corpus relevant to the changing nature of research, we design a document collection pipeline that scrapes documents from various sources of literature, as mentioned previously. Although we will not reiterate the document collection process, we will discuss some designs and challenges with this process. As of now, only Scopus and ScienceDirect are used as data sources, and we plan to add more sources, including APIs from literature databases, such as arXiv, and articles scraped from media sites and blogs.

One challenge that we might face in this expansion of data sources is the discrepancies in the metadata collected. Each database supports different formats of metadata, and when retrieved, it provides conflicts. In our use case, we propose to only retain the most relevant metadata such as title, authors, abstract, etc.

Another component that needs refinement is the user interface. The current version of the web application has not been publicly tested. Similar to many other products in their early stages, it needs to be refined based on users' feedback in iterative testing. We plan to gather more opinions from users of different demographics, with a focus on what additional features should be added and how the usability of the tool can be improved.

Section 6: Acknowledgments

The researchers' deepest gratitude goes to those without whose efforts this project could not have come to fruition: the Project Oversight Group and others for their diligent work overseeing development and analysis and reviewing and editing this report for accuracy and relevance.

Project Oversight Group members:

Himchan Jeong, FSA, PhD

Rasa McKean, ASA, MAAA, ACAS

Taha Mokfi

Jianxi Su, FSA, PhD

At the Society of Actuaries Research Institute:

Rob Montgomery, ASA, MAAA, Research Project Manager

Appendix A: Article Information Collected

Table 3

ATTRIBUTES OF EACH ARTICLE

Column Name	Column Definition
affiliation_city	Affiliation city
affiliation_country	Affiliation country
affilname	Affiliation name
afid	Affiliation ID
aggregationType	Type of publication (Book, Journal, etc.)
article_number	Article number
authkeywords	Author provided keywords
author_afids	Author affiliations
author_count	Number of authors
author_ids	Author IDs
author_names	Author names
citedby_count	Number of times that this article is cited
coverDate	Publication date
coverDisplayDate	Publication year
creator	Corresponding author
description	Abstract
doi	Digital Object Identifier
eIssn	Electronic International Standard Serial Number
eid	Scopus EID
fund_acr	Sponsor acronym
fund_no	Grant number
fund_sponsor	Sponsor name
identifier	Scopus ID
issn	International Standard Serial Number
issueIdentifier	Issue
openaccess	Open access flag (0 or 1)
openaccessFlag	Open access flag (False or True)
pageRange	Page range
pii	Publisher Item Identifier
publicationName	Publication name (Journal name, book name, etc.)
source_id	Scopus source ID
subtype	Subtype code
subtypeDescription	Subtype description (Review, article, etc.)
title	Title of article
url	Link to article
volume	Volume

Appendix B: Keyword Clusters at a Glance

Table 4

SUMMARY OF KEYWORD CLUSTERS

Cluster No.	Cluster Name	# of Keywords	# of Papers
C1	Detection	1150	4866
C2	Cyber Terminology	2000	5086
C3	Cyber System Management	1392	6588
C4	Cyber Regulations & Public Policy	1855	3966
C5	Healthcare	495	971
C6	Computational Intelligence	1090	5709
C7	Mobile	377	1075
C8	System Security	1588	9372
C9	Misc. I	944	2821
C10	Finance & Economics	1169	3114
C11	Security Breach	1597	6104
C12	Misc. II	1002	3038
C13	Technology Management	1315	5203
C14	Malware	800	3289
C15	Cyberphysical Devices	925	6813
C16	Misc. III	2015	10097
C17	Electronic Control	1365	5017
C18	Cyber Crimes	753	2711
C19	System Resilience	1787	6114
C20	Cryptography	526	1454
C21	Algorithm	1322	2536
C22	Data Management	2131	6147
C23	Misc. IV	2835	6895
C24	Internet of Things	845	4295
C25	Smart Network	802	2920
C26	Power System	809	2115
C27	Assessment	2347	6676
C28	Learning	761	3254
C29	Cyber Security	1242	16868
C30	Cyber Attack	804	3280

References

Agrawal, Neha and Rohit Kumar. 2022. “Security Perspective Analysis of Industrial Cyber Physical Systems (I-CPS): A Decade-wide Survey”. In: ISA Transactions 130, pp. 10–24.

Bou-Harb, Elias. 2016. “A Brief Survey of Security Approaches for Cyber-Physical Systems”. In: 2016 8th IFIP International Conference on New Technologies, Mobility and Security (NTMS). 2016 8th IFIP International Conference on New Technologies, Mobility and Security (NTMS). Larnaca, Cyprus: IEEE, pp. 1–5.

Danilevsky, Marina, Kun Qian, Ranit Aharonov, Yannis Katsis, Ban Kawas, and Prithviraj Sen. 2020. “A Survey of the State of Explainable AI for Natural Language Processing”. In: arXiv.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. “Bert: Pre-training of deep bidirectional transformers for language understanding”. In: arXiv preprint arXiv:1810.04805.

Dsouza, Joanita, Laura Elezabeth, Ved Prakash Mishra, and Rachna Jain. (Feb. 2019). “Security in Cyber-Physical Systems”. In: 2019 Amity International Conference on Artificial Intelligence (AICAI). 2019 Amity International Conference on Artificial Intelligence (AICAI). Dubai, United Arab Emirates: IEEE, pp. 840–844.

Eling, Martin. 2020. “Cyber risk research in business and actuarial science”. In: European Actuarial Journal 10.2, pp. 303–333.

Grootendorst, Maarten. 2020. KeyBERT: Minimal keyword extraction with BERT. Version v0.3.0.

Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. “Efficient estimation of word representations in vector space”. In: arXiv preprint arXiv:1301.3781.

Mikolov, Tomas, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. “Distributed representations of words and phrases and their compositionality”. In: Advances in neural information processing systems 26.

Reimers, Nils and Iryna Gurevych. 2019. “Sentence-bert: Sentence embeddings using siamese bert-networks”. In: arXiv preprint arXiv:1908.10084.

About The Society of Actuaries Research Institute

Serving as the research arm of the Society of Actuaries (SOA), the SOA Research Institute provides objective, data-driven research bringing together tried and true practices and future-focused approaches to address societal challenges and your business needs. The Institute provides trusted knowledge, extensive experience and new technologies to help effectively identify, predict and manage risks.

Representing the thousands of actuaries who help conduct critical research, the SOA Research Institute provides clarity and solutions on risks and societal challenges. The Institute connects actuaries, academics, employers, the insurance industry, regulators, research partners, foundations and research institutions, sponsors and non-governmental organizations, building an effective network which provides support, knowledge and expertise regarding the management of risk to benefit the industry and the public.

Managed by experienced actuaries and research experts from a broad range of industries, the SOA Research Institute creates, funds, develops and distributes research to elevate actuaries as leaders in measuring and managing risk. These efforts include studies, essay collections, webcasts, research papers, survey reports, and original research on topics impacting society.

Harnessing its peer-reviewed research, leading-edge technologies, new data tools and innovative practices, the Institute seeks to understand the underlying causes of risk and the possible outcomes. The Institute develops objective research spanning a variety of topics with its strategic research programs: aging and retirement; actuarial innovation and technology; mortality and longevity; diversity, equity and inclusion; health care cost trends; and catastrophe and climate risk. The Institute has a large volume of topical research available, including an expanding collection of international and market-specific research, experience studies, models and timely research.

Society of Actuaries Research Institute
475 N. Martingale Road, Suite 600
Schaumburg, Illinois 60173
www.SOA.org