Deciphering Cyber Threats: A Unifying Framework with GPT-3.5, BERTopic and Feature Importance

Chun Man Tsang
School of Computing and Communications
Lancaster University
Lancaster, United Kingdom
chunman_tsang@hotmail.com

Tom Bell
Electronics and Computer Science
University of Southampton
Southampton, United Kingdom
t.bell@soton.ac.uk

Antonios Gouglidis and **Mo El-Haj**
School of Computing and Communications
Lancaster University
Lancaster, United Kingdom
a.gouglidis | m.el-haj @lancaster.ac.uk

Abstract

This paper presents a methodology for the categorisation and attribute quantification of cyber threats. The data was sourced from Common Weakness Enumeration (CWE) entries, encompassing 503 hardware and software vulnerabilities. For each entry, GPT-3.5 generated detailed descriptions for 12 key threat attributes. Employing BERTopic for topic modelling, our research focuses on clustering cyber threats and evaluates the efficacy of various dimensionality reduction and clustering algorithms, notably finding that UMAP combined with HDBSCAN, optimised through parameterisation, outperforms other configurations. The study further explores feature importance analysis by converting topic modelling results into a classification paradigm, achieving classification accuracies between 60% and 80% with algorithms such as Random Forest, XGBoost, and Linear SVM. This feature importance analysis quantifies the significance of each threat attribute, with SHAP identified as the most effective method for this calculation.

1 Introduction

In response to the evolving threat landscape, a range of techniques have been employed to enhance the pace and quality of vulnerability discovery and threat analysis. A core activity in this endeavor is the use of cyber threat modelling techniques. Cyber threat modelling typically approaches the problem from the perspective of software vulnerabilities (Khan et al., 2017), attacker profiles (MITRE), or system assets (Caralli et al., 2007). Asset-based modelling, in particular, offers several advantages, including the capability to conduct automated reasoning over a threat knowledge base.

There are two specific research gaps which this research seeks to address. Firstly, there is a lack of concise sources of threat information with sufficient coverage for asset-based threat modelling. For a cyber threat modelling process to be valid, it needs a broad and up-to-date threat information database. However, for structured asset-based models, such as those using the Web Ontology Language (OWL), the database must also be concise. An ideal threat database should be generated using a repeatable and automated process to ensure it stays up-to-date as the threat landscape changes.

Existing open-source threat databases, like CVE (MITRE, 2023b), CWE (MITRE, 2024), and CAPEC (MITRE, 2023a), are typically too large to be converted into a structured representation for meaningful analysis. This makes it difficult to ensure the validity of the threat model. Researchers tend to select a subset of threat entries of these databases, thereby reducing their coverage. Even if a complete threat knowledge base is modeled, it quickly becomes outdated as new entries are added. Either way, there is a need to develop a technique for repeatably generating a consolidated and up-to-date threat knowledge base without compromising coverage.

Secondly, there is no robust quantitative methodology for characterising cyber threats from a given threat knowledge base. Existing techniques, including ontology engineering methodologies (Fernández-López et al., 1997; Uschold and Gruninger, 1996), do not offer quantitative meth-

Proceedings of the 1st International Conference on NLP & AI for Cyber Security, pages 175–185
July 29–30, 2024.

ods for identifying key threat attributes which are pertinent to threat modelling. Hence, threat models are typically, at least to some degree, based on the subjective experience and intuition of the designer (Shostack, 2014), weakening the academic justification for selecting specific threat attributes. Therefore, a new robust technique is needed to automatically identify threat attributes from a knowledge base for characterising cyber threats.

This research addresses these gaps by demonstrating a viable method to generate a concise threat database using a highly repeatable and largely automated process. It also identifies the key attributes which constitute a cyber threat based on this database. The technique developed involves two main steps. First, it uses topic modelling to cluster primary cyber threat information into groups of normative threat classes. Second, it performs feature importance analysis to determine the relative importance of each threat attribute. This allows us to identify the most important concepts for creating a generic threat model for asset-based cyber threat modelling.

2 Background

2.1 Topic modelling

The advent of newer topic models such as BERTopic (Grootendorst, 2022) and Top2Vec (Angelov, 2020) attracted attention in academia, particularly in comparison to traditional topic modelling techniques like Latent Dirichlet Allocation (LDA) (Blei et al., 2003) and Non-Negative Matrix Factorisation (NMF) (Seung and Lee, 1999). For instance, Egger and Yu (2022) undertook a comprehensive performance assessment across LDA, NMF, Top2Vec, and BERTopic using Twitter posts as the primary dataset. Their findings revealed that BERTopic outshined its counterparts across multiple aspects of topic modelling. Contrarily, Top2Vec demonstrated limitations, most notably the overlap of generated topics and the encapsulation of multiple concepts within individual topics, which compromised its proficiency in distinct topic identification.

Additional studies corroborated the superiority of BERTopic over traditional models. In particular, de Groot et al. (2022), Zankadi et al. (2023), and Ogunleye et al. (2023) conducted evaluations that favoured BERTopic against LDA. While the dataset employed by Groot et. al, was multi-domain in nature, the latter two studies utilised Twitter posts from specific user groups. Despite the variability in datasets, a consensus emerged across these works: BERTopic consistently outperformed LDA in generating more coherent and distinct topics.

One of the principal challenges of this study was the dataset's unique nature, which set it apart from those commonly used in existing topic modelling research. Unlike the wider thematic scope of datasets examined in prior studies, our dataset contained texts that exclusively described cyber threats. Consequently, the latent themes inherent in these texts were expected to be significantly narrower. This limited thematic range presented a formidable challenge for any topic model tasked with producing distinct yet coherent topics.

The second challenge stemmed from the structural complexities of our dataset. In stark contrast to the datasets employed in previous studies, which consisted of 'documents'—each being a standalone text object of variable length, our dataset comprised multiple distinct texts for each data object. Each of these texts corresponded to a specific pre-defined threat attribute, effectively making each data object a multidimensional textual entity. This contrasted sharply with traditional textual datasets and resembled more closely a numerical dataset where each data point possesses values across a range of distinct variables or features. Given the structural complexities of the dataset, our approach to data handling and structural preservation could prove to be a pivotal factor influencing experimental outcomes.

Recent research has exhibited a notable interest in applying topic modelling techniques to the cyber security domain, albeit with varied objectives and scopes. Kolini and Janczewski (2017) employed LDA to analyse governmental documents, aiming to shed light on national cyber security strategies and policies. Another research project led by Adams et al. (Aug 2018) utilised LDA on Common Attack Pattern Enumeration and Classification (CAPEC) data, an alternative to Common Weakness Enumeration (CWE), for the classification of cyber threats. However, the study principally used topic modelling as a mechanism for generating intermediary outputs for subsequent modelling, rather than focusing on clustering cyber threats or extracting latent topics from the onset. Kumar et al. (2022) harnessed LDA to examine academic databases and cyber security blogs, aiming to evaluate the shifting popularity of overarching cyber

themes in the pre- and post-COVID-19 era. Additionally, a study by Suryotrisongko et al. (2022) utilised advanced methods such as BERTopic and Top2Vec for keyword extraction from a leaked dataset pertaining to hacker forums, primarily to augment cyber threat intelligence gathering.

While these studies demonstrated the versatility and applicability of topic modelling in the cyber security domain, they do not directly align with the primary aim of our research, which is to cluster established types of cyber threats based on their textual descriptions. Moreover, a common shortcoming among these studies was the lack of a structured evaluation of the performance of these topic models when applied to textual data associated with cyber security.

2.2 Feature importance analysis in clustering

Clustering-Model-agnostic approaches proposed by Ellis et al. (2021) and Scholbeck et al. (2022) deployed permutation techniques, which involved the shuffling of feature values to gauge their respective impact on clustering outcomes. While promising, these approaches posed significant challenges including demand of considerable computational resources, requirement of non-trivial selection of a suitable metric by practitioners, and inadequate evaluation in recent research.

A distinct methodology was proposed by Ismaili et al. (2014) and Badih et al. (2019), which involved training a classifier to predict the cluster allocation based on feature values. Feature importance for clustering was subsequently deduced from the importance metrics utilised in the classifier. Examples included metrics like mean decrease impurity in Random Forest (Breiman, 2001) and XGBoost (Chen and Guestrin, 2016), as well as weight coefficients in Support Vector Machines (SVM) (Rakotomamonjy, 2003). This classifier-based approach offered the dual advantage of implementation feasibility and methodological robustness by leveraging well-established feature importance methods from classification tasks.

3 Data

3.1 Source list - common weakness enumeration

Common Weakness Enumeration (CWE)[1] is a community-developed list of software and hardware weakness types. It has been created to serve as a standardised method of describing and classifying security-related weaknesses in code and design. CWE list acts as a baseline collection of cyber threats.

We selected a total of 503 CWE entries for this study - all 399 available Software Development entries and all 104 available Hardware Design entries were included. Research Concepts related entries were excluded due to its redundancy with the other two groups and the low relevance with the future development of cyber threat models.

3.2 Enhanced descriptions using GPT-3.5

GPT, or Generative Pre-trained Transformer, is a large language model (LLM) developed by OpenAI, and GPT-3.5 is its 3.5th generation version[2]. Seeing the potential in GPT-3.5, we decided to leverage its capabilities to improve the dataset. We picked 12 key threat attributes: vulnerability, method, technical impact, security properties affected, severity, likelihood, relevant assets, the attack vector(s), the attacker type(s), the attacker motive(s), relevant cyber controls/countermeasures, and detection methods. For every CWE entry, GPT-3.5 was used to generate text descriptions for these attributes[3]. Notwithstanding our study was aided by GPT, the discussion on its properties and performance was out of the scope of this study. Table 1 summarises the average word counts of the primary dataset.

4 Method - topic modelling

For this task, we employed "BERTopic", an advanced approach built on the foundation of BERT (Bidirectional Encoder Representations from Transformers) (Devlin et al., 2019). BERT is renowned for its ability to understand the context in which words are used, making it especially valuable for datasets such as ours which focused on a specialised field like cyber security.

Our decision to opt for BERTopic can be attributed to two reasons. First is **Contextual Understanding**: Traditional models like LDA view texts

[1]The source CWE list can be downloaded at https://cwe.mitre.org/data/downloads.html.

[2]https://platform.openai.com/docs/guides/gpt

[3]The prompt used in the Chat Completions API: "Here is the description of CWE {ID}: {CWE Description}; Use what you know about this CWE and the description provided to describe the following attributes of this threat for me: the vulnerability, method, technical impact, security properties affected, severity, likelihood, relevant assets, the attack vector(s), the attacker type(s), the attacker motive(s), relevant cyber controls/countermeasures, and detection methods."

CWE Entries			Type		Total
			Hardware Design	Software Development	
Item Count			104	399	503
Average Word Count	Attr.1	Vulnerability	23.8	26.3	25.8
	Attr.2	Method	24.4	24.6	24.6
	Attr.3	Technical impact	32.3	34.1	33.8
	Attr.4	Security properties	31.5	31.6	31.6
	Attr.5	Severity	32.9	33.1	33.1
	Attr.6	Likelihood	36.8	35.1	35.5
	Attr.7	Relevant assets	25.7	25.9	25.9
	Attr.8	Attack vector	28.7	28.7	28.7
	Attr.9	Attacker type	28.4	27.8	27.9
	Attr.10	Attacker motive	27.7	27.0	27.1
	Attr.11	Counter-measures	35.3	34.6	34.7
	Attr.12	Detection methods	37.4	35.6	35.9

Table 1: Average word counts of primary dataset.

as simple bags of words, often missing the varied meanings a word can have in different contexts. BERT, on the other hand, can discern these distinctions. For instance, it recognises that the word "bank" in "I sat on the bank of the river" and "I went to the bank to withdraw money" conveys different meanings. Second is **Flexibility in Handling Texts**: The BERT-based model excels in dealing with shorter texts, whereas many traditional models fail. Its ability to understand context ensures that even concise sentences are interpreted correctly, making it invaluable for datasets with varied text lengths. Especially, one of the following proposed approaches required iterations of processing on one short sentence.

4.1 BERTopic implementation

Our BERTopic implementation was organised in four primary steps: Embedding, Dimension Reduction, Clustering, and Topic Representation. In the **Embedding** phase, we used numerical vectors to transform each text into a unique fingerprint. Specifically, we employed the default BERT Sentence Embedder (Reimers and Gurevych, 2019) with the pre-trained model "*all-MiniLM-L6-v2*".

The second step, **Dimension Reduction**, is important due to the high-dimensionality of the data. We explored two methods for this: UMAP (Uniform Manifold Approximation and Projection, "UMP") (McInnes et al., 2018) and Principal Component Analysis (PCA) (Jolliffe, 2002). UMAP, the default method in BERTopic, excels at preserving both local and global structures in the data, making it suitable for textual data. On the other hand, PCA aims to capture the maximum variance from the original data in fewer dimensions but may overlook local structures.

For **Clustering**, we investigated two primary algorithms: HDBSCAN (Hierarchical Density-Based Spatial Clustering of Applications with Noise, "HDB") (McInnes et al., 2017) and K-Means ("KMS") (Arthur and Vassilvitskii, Jan 7, 2007). HDBSCAN, the default method in BERTopic, offers several features like density-based clustering, identification of clusters with differing densities, and the ability to spot outliers. It is also advantageous because it does not require specifying the number of clusters beforehand. K-Means, a well-established method, features centroid-based clustering and mandates prior specification of the number of clusters (K).

The final step, **Topic Representation**, involves identifying the main themes or topics for each cluster by locating the keywords or terms, known as "topic words". BERTopic utilises c-TF-IDF, a variation of the well-known TF-IDF algorithm for this purpose.

Before initiating the BERTopic process, we also considered two different data pre-handling strategies. The first, dubbed **Unified Document Approach ("UNI")**, amalgamates the 12 attributes for each entry into one comprehensive document. This aims to simulate the typical data structure used for topic modelling. The second strategy, **Attribute-Specific Approach ("ATT")**, treats each of the 12 attributes separately and combines them only after individual processing. This preserves the distinct nature of each attribute and provides a point of contrast with the Unified Document Approach. Figure 1 depicts the high-level process of topic modelling pipeline.

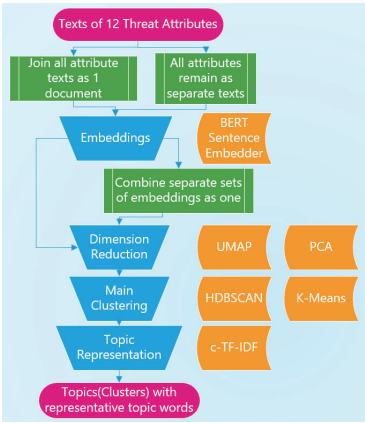

Figure 1: High-level process of topic modelling.

4.2 Hyperparameter tuning

Our approach was quite comprehensive, involving the proposal of two textual data pre-handling approaches, two dimension reduction methods, and two primary clustering algorithms, together forming eight different model combinations, which we refer to as meta-models. It is important to note that our experimentation was not limited to these eight configurations. While the steps of **Embedding** and **Topic Representation** remained constant, both **Dimension Reduction** and **Clustering** methods were accompanied by a myriad of user-specifiable hyperparameters, each forming what we call a sub-model of a meta-model.

Navigating this space posed a multi-dimensional challenge. In traditional applications like clustering, the computational resources required tend to escalate exponentially with the introduction of each new hyperparameter. In our case, because we were intertwining dimension reduction and clustering, the interplay between these methods could not be ignored. Acknowledging this complexity, our strategy focused on adjusting one or two key parameters from each method while keeping the rest at their default settings.

5 Method - feature importance analysis

In the evolving landscape of feature importance analysis, many recently proposed methods for clustering are model-specific. These tailored techniques impose constraints when applied across different clustering and topic modelling methodologies. Given this challenge, our approach leveraged the robust feature importance techniques from the classification paradigm, which exhibit (clustering) model-agnostic properties:

Our approach encompassed four key facets. Firstly, we transformed the clustering outcome as a **Conversion to Classification Task**. This entailed using classification models, or classifiers, to predict the clustering labels based on BERT embeddings. The attribute-specific BERT embeddings and clustering labels obtained post-topic modelling served as our input data and target variables for classifier training, respectively.

Secondly, during **Classifier Training**, we utilised three established classifiers: Random Forest, XGBoost, and Linear SVM. Each classifier has its unique mechanism for evaluating feature importance.

In the third aspect, **External Method Integration**, we broadened our analytical scope by adding external methodologies, specifically SHAP or permutation importance, to each classifier. These methods provided an independent basis for contrasting with the classifiers' built-in feature importance techniques.

Lastly, the **Aggregation and Normalisation** step was crucial. Given that our importance analysis hinged on BERT embeddings rather than directly on the 12 threat attributes, an aggregation step was essential. This step summarised the importance values attributed to each of the 12 threat attributes. To ensure a consistent interpretation across different methods, we normalised these importance values into relative percentages. Figure 2 highlights the high-level process of feature importance analysis. In contrast to the previous Topic Modelling task, our Feature Importance analysis did not employ any Dimension Reduction techniques and hence the results retained their interpretability.

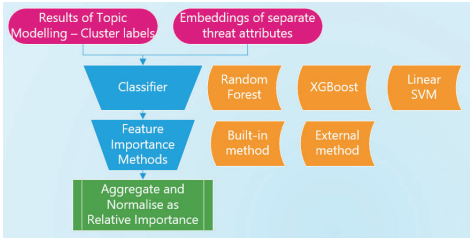

Figure 2: High-level process of feature importance analysis.

5.1 Classifier training

Classifier training also required hyperparameter tuning. Similar to what we did in clustering, this process helped find the best parameterised settings for the classifier to work most effectively. Initial tests showed that reducing the dimension of embeddings before training the classifier adversely affected its accuracy. Therefore, we decided to use the original embeddings without any changes. This decision allowed us to adjust a broader array of hyperparameters for each classifier. However, given our resources, it was not feasible to test every possible combination of varying hyperparameters. To manage this, we used a two-step approach using tools like "RandomizedSearchCV" and "GridSearchCV" from the Scikit-learn (Pedregosa et al., 2011) package:

- **Initial Exploration**: We picked 500 random settings from a list of common hyperparameters for the classifier. The aim was to see

which setting among these offered the best accuracy based on cross-validation results.

- **Refined Search**: After identifying the best settings from the initial exploration, we then did a more detailed search. Here, we looked at settings that were slightly higher or lower (or combinations of these changes) than the best ones we identified. Again, the goal was to find the best setting based on cross-validation results.

5.2 External feature importance methods

This subsection focuses on the techniques we employed for external feature importance analysis. Our primary choice for this purpose was SHAP, an approach based on cooperative game theory. We also faced some challenges related to computational resources, particularly when applying these methods to different types of models.

5.2.1 SHAP analysis

SHAP, or SHapley Additive exPlanations (Lundberg and Lee, 2017), is a game theory-inspired tool designed to explain machine learning model predictions. In the machine learning context, SHAP assigns importance scores to features for each specific prediction, helping to reveal how each feature influences the outcome. One of its primary advantages is its strong theoretical foundation, which derives from cooperative game theory (Štrumbelj and Kononenko, 2014). This theoretical robustness ensures that SHAP offers a sound approach to feature importance. Additionally, SHAP stands out for its ability to account for complex interactions between different features, a facet often overlooked by other methods.

Despite these merits, SHAP is not without its challenges, the most prominent of which is its computational intensity. Fortunately, optimised implementations for tree-based models like Random Forest and XGBoost are available in dedicated libraries. However, when we initially tried applying SHAP to our Linear SVM models, we found that the computational resources required exceeded what was available to us. Consequently, we sought alternative methods for feature importance analysis in the context of Linear SVM.

5.2.2 Permutation importance

Permutation importance provides an independent way of gauging the importance of individual features (Fisher et al., 2019). This is accomplished by evaluating how much a model's performance drops when the values of a particular feature are shuffled around randomly. Essentially, by mixing up the feature values, we disrupt its connection to the target variable. This helps us discern how reliant the model is on that feature to make accurate predictions.

To understand a feature's importance, we compare the model's baseline performance (without any permutation) to its performance after the feature values are shuffled. A significant drop in performance indicates a vital feature, while a marginal decrease suggests that the feature is not pivotal for the model's predictive capability.

6 Results and evaluation

6.1 Metrics for clustering and topic modelling

In our investigation, we utilised a dual set of evaluation metrics: general clustering metrics and topic modelling-specific metrics. The general clustering metrics employed were the Silhouette Method (Rousseeuw, 1987) and Calinski-Harabasz (CH) Index (Caliński and JA, 1974), both of which are widely acknowledged for gauging clustering efficacy. For topic modelling, we assessed models based on topic diversity (Dieng et al., 2020) and coherence scores (Röder et al., 2015).

6.2 Strategic topic model choice

The quest for models that excelled across all metrics proved impractical due to the inherent trade-offs observed among them—especially the typically inverse relationship between topic diversity and coherence scores. Although our initial tendency was to prioritise topic modelling metrics, particularly topic diversity, we found that it was imperative to have a balanced evaluation using all metrics. This approach led us to shortlist 10 sub-models (parameterised versions of meta-models), with one or two representing each meta-model (Table 2).

Before advancing to qualitative evaluation, we were inclined to emphasise the importance of topic diversity. Given our specific focus on cyber security, a higher topic diversity was more critical as it ensured that each cluster was distinct from one another, implying clearer contextual categories were formed for the cyber threat texts. High topic coherence, on the other hand, implied the topic words in each cluster being consistent to derive a single latent theme. As most of the topic words across the

SN	Meta-Model	Total clusters	Silhouette	CH	Diversity	Coherence
1	ATT+UMP+HDB	55	0.0457	4.2512	0.8145	0.3927
2	ATT+UMP+HDB	52	0.0461	4.3232	0.7827	0.4017
3	UNI+UMP+HDB	55	0.0635	5.4099	0.8455	0.3969
4	UNI+UMP+HDB	57	0.0725	5.4661	0.8211	0.3915
5	ATT+UMP+KMS	19	0.0328	8.3167	0.5833	0.6529
6	ATT+UMP+KMS	19	0.0392	8.406	0.5444	0.6154
7	UNI+UMP+KMS	10	0.0451	15.0817	0.4667	0.7112
8	UNI+UMP+KMS	10	0.044	15.0501	0.4444	0.7434
9	ATT+PCA+KMS	28	0.0391	6.729	0.6556	0.5372
10	UNI+PCA+KMS	7	0.0495	19.356	0.4667	0.7117

Table 2: Summary of finalist sub-models. Silhouette and CH scores for evaluating clustering performance. Diversity and coherence scores for evaluating topic modelling performance. The highlighted sub-model 4 is eventually selected as the final parameterised model.

clusters were expected to revolve around the theme of cyber security, topic coherence might be less pivotal compared to topic diversity for our specific objectives.

Subsequently, Subject Matter Experts (SMEs) in cyber security opted for sub-model 4 of (UNI+UMP+HDB), which is of relatively higher topic diversity score, as the most effective model for generating distinct and meaningful clusters after their qualitative evaluation. This concurrence between expert opinion and our quantitative metrics further substantiated our evaluation approach.

6.3 Classifier efficacy and refinement

We leveraged our domain expertise to consolidate the number of clusters to 19 from original 57 with a coherent set of contextual descriptions for threat categories post cluster merging.

This reduction in number of clusters not only anticipated an increase in classifier accuracy but also resolved the issue of under-represented clusters. Post-merging, every cluster comprised a minimum of five data points. Consequently, this allowed for a stratified 70:30 training-test data split and a 2-fold Cross-Validation (CV) on the training data, ensuring that each cluster (or class) was adequately represented in all splits.

Subsequently, three classifiers, namely Random Forest, XGBoost and Linear SVM were trained with the training data with CV approach. Two distinct classification accuracy metrics were considered: CV score and Test Accuracy. While the former accuracy score was gauged through cross-validation on the training data, the latter was assessed using the independent testing data. A summarised performance of the three classifiers, is presented in Table 3. Empirical results assuredly indicated that the Linear SVM model exhibited superior performance. It was closely followed by XGBoost, and finally Random Forest.

Classifier	CV Score	Test Accuracy
Random Forest	0.65	0.62
XGBoost	0.65	0.72
Linear SVM	0.71	0.80

Table 3: Classification accuracy of 3 classifiers.

6.4 Feature importance analysis

After training the classifiers, feature importance was analysed for 12 distinct threat attributes. For each of the three classifiers—Random Forest, XGBoost, and Linear SVM—two methods were utilised for this analysis: one built-in method inherent to each classifier and one external method, yielding six methods in total. Figure 3 depicts a box-plot summarising the relative importance of the 12 attributes across all methods.

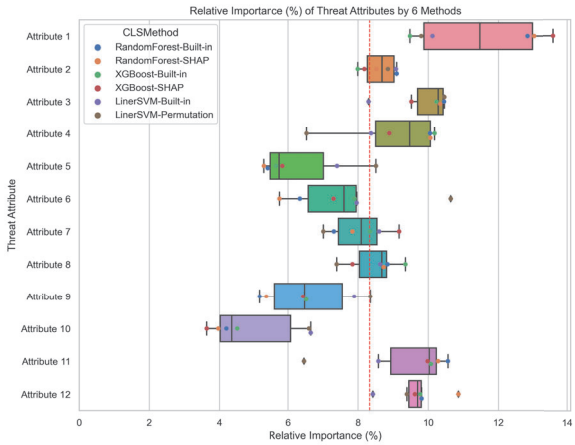

Figure 3: Relative importance of 12 threat attributes across 6 methods.

A reference point of 8.33% was considered, predicated on an even distribution of feature importance across all attributes. Among the attributes, Attribute 1 conspicuously led the pack, followed by Attributes 3, 11, and 12, all of which surpassed the reference point. Attributes 4 and 2 also held relative importance, as evidenced by the majority of box exceeding the reference line. In contrast, Attribute 10 was evidently least important, followed by Attributes 5 and 9.

Nevertheless, the summarisation of feature importance scores across six different methods raises concerns about the fairness of the comparison. For instance, the Linear SVM showed a notably narrow span in the distribution of its relative importance

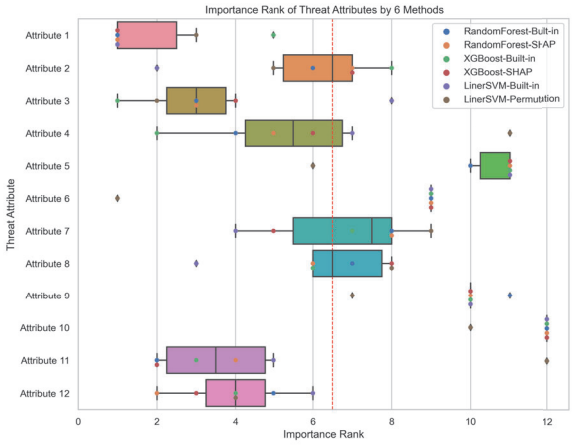

Figure 4: Importance rank of 12 threat attributes across 6 methods.

scores (approximately 6% to 10%) compared to other methods (approximately 4% to 13%), thereby understating the differences between more and less important attributes. To address this issue, a comparative analysis based on ranks of feature importance was also performed, with results summarised in another box-plot (Figure 4).

In this ranked comparison, lower ranks signified higher importance, and the median rank of 6.5 served as the reference. The general pattern largely resembled that of the initial relative importance plot, yet certain outliers became more discernible. For example, while Attribute 11 predominantly exceeded the median rank, it also exhibited one instance of ranking last (12th position). Conversely, Attribute 6, which generally held the 9th rank, had a singular instance of claiming the top rank. Notably, most of the outliers were from either method pertaining to Linear SVM.

These visual representations, however, only provide a high-level overview of the relative feature importance. Our ultimate goal was to quantitatively assess the importance of each threat attribute. A simple averaging of importance values across methods was deemed inadequate due to differing scales of relative importance among the classifiers. This was particularly evident with Linear SVM, which exhibited a narrow range that could introduce bias into the aggregated results.

As such, an alternative approach could involve the adoption of a single set of feature importance scores from just one method. However, challenge arose in the absence of an objective metric to conclusively determining the superior method among alternatives. We proposed to consider not only the accuracy of the parent classifiers but also the qualitative properties and patterns yielded in the final results.

Our primary reservations stemmed from the conspicuously narrow range of relative importance values (approximately 6% to 10%) reported by the Linear SVM. This narrow range might hamper the effective differentiation of feature importance. This limitation might be attributable to the inherent mathematical and algorithmic differences in how SVM performs classification. Specifically, SVM relies on analytical techniques to determine the optimal hyperplanes within the feature space to segregate data points. Because this is performed in an analytical fashion, SVM is inclined to utilise as many data dimensions as feasible, even though some dimensions (or features) might have a relatively higher influence, thus resulting in minor variations in feature importance.

On the contrary, tree-based algorithms like Random Forest and XGBoost adopt a "winner-takes-all" strategy in data splitting. In each split, only one single feature is selected based on its efficacy in dividing the data. This property of tree-based algorithms renders their feature importance measurements noticeably more effective than those derived from SVM.

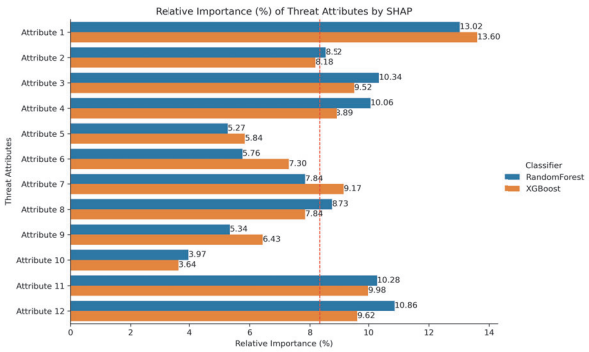

Figure 5: Relative importance of 12 threat attributes by Random Forest-SHAP and XGBoost-SHAP.

Given these considerations, we recommended the SHAP method for its rigorous mathematical underpinnings, grounded in cooperative game theory, and its model-agnostic nature. Furthermore, SHAP factors in the interactions among features when computing importance scores. The bar chart in Figure 5 compares feature importance scores generated by the SHAP method using Random Forest and XGBoost classifiers. Given the similarities in their patterns and the common tree-based algorith-

mic foundation, an average of the two sets could be considered. However, should a single set be chosen, the XGBoost-derived feature importance would be preferable owing to its superior classification accuracy.

While a full analysis of the significance of these results for cyber threat modelling is outside the scope of this particular paper, we can identify from Figure 5 that the most important attributes for characterising cyber threats include the vulnerability, technical impact and security properties[4] associated with a given cyber threat.

7 Conclusion

This paper presents a robust framework for the unsupervised classification of cyber threats and the quantitative analysis of their attributes, harnessing cutting-edge data science methods on textual data generated by GPT-3.5. BERTopic successfully addressed the principal challenge of clustering cyber threat texts with specialised and narrow themes. Our findings revealed that an optimally parameterised combination of UMAP and HDBSCAN—BERTopic's default settings—outperformed other configurations in both quantitative metrics and expert qualitative evaluations. Of the two text pre-handling strategies we employed, the one that maintained the original attribute structure proved less effective for improved topic modelling. We argue, however, that its high dimensionality may have influenced these results negatively.

In the next phase of our study, we ventured into feature importance analysis to characterise cyber threat attributes quantitatively. We sidestepped the limitations of immature feature importance methods for clustering by adopting a classification-based approach. Utilising classifiers such as Random Forest, XGBoost, and Linear SVM, we achieved classification accuracies ranging from 60% to 80%. Among the feature importance techniques evaluated, SHAP stood out for its strong theoretical foundation and reliable performance.

The cyber threat attributes identified using our feature importance technique can serve as the basis for constructing a cyber threat model to automate the analysis of cyber threats using asset-based threat modelling techniques. The methodology could also be applied to more bespoke knowledge domains for identifying threat attributes and developing threat databases and models in niche security domains. The concise threat database and corresponding threat model would be beneficial for security experts, researchers and policy makers in tasks such as cyber audits and risk assessments.

Furthermore, the methodologies and insights from this study hold potential for application in other sectors that rely on text-rich data for analytical interpretation, such as healthcare, law, and social sciences, aiding in extracting meaningful information and facilitating better decision-making. Our methodological framework is not only robust but also modular and adaptable, offering promising avenues for future research in the fast-evolving landscape of machine learning and large language models.

Limitations

Despite the challenges posed by the narrow-themed nature of the cyber security text dataset, our BERTopic-based methodology successfully formed coherent clusters. Subject-matter experts (SMEs) could summarise threat categories post-cluster merging, though this required referencing original CWE descriptions, the CWE hierarchical structure, and hierarchical clustering distances. This effort to transform topic words into a human-interpretable narrative is an universal challenge in topic modelling endeavours.

Like many machine learning methodologies, our topic modelling framework incorporated elements of randomness. Specifically, algorithms such as UMAP, PCA, and K-Means introduce randomness. Therefore, a potential refinement could involve parameterising the random seed in hyperparameter tuning to ensure greater robustness yet maintain reproducibility.

Likewise, our feature importance analysis pipeline involved stochastic elements, and they are not merely confined to the algorithms of classifiers—Random Forest and XGBoost; it extends to the randomness inherent in the training and test data split as well as in cross-validation procedures.

Ethics Statement

This research is underpinned by a commitment to ethical practices in all aspects of data collection, analysis, and interpretation.

[4]Please refer to Table 1 for the mapping between each threat attribute and the attribute number.

References

Stephen Adams, Bryan Carter, Cody Fleming, and Peter A. Beling. Aug 2018. Selecting system specific cybersecurity attack patterns using topic modeling. pages 490–497. IEEE.

Dimo Angelov. 2020. Top2vec: Distributed representations of topics.

David Arthur and Sergei Vassilvitskii. Jan 7, 2007. k-means++: the advantages of careful seeding. SODA '07, pages 1027–1035. Society for Industrial and Applied Mathematics.

Ghattas Badih, Michel Pierre, and Boyer Laurent. 2019. Assessing variable importance in clustering: a new method based on unsupervised binary decision trees. *Computational statistics*, 34(1):301–321.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3(null):993–1022.

Leo Breiman. 2001. Random forests. *Machine learning*, 45(1):5–32.

Tadeusz Caliński and Harabasz JA. 1974. A dendrite method for cluster analysis. *Communications in Statistics - Theory and Methods*, 3:1–27.

Richard Caralli, James Stevens, Lisa Young, and William Wilson. 2007. Introducing octave allegro: Improving the information security risk assessment process. *Scientific and technical aerospace reports*, 45(25).

Tianqi Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. *Proceedings of the 22nd ACM SIGKDD International Conference on knowledge discovery and data mining*, pages 785–794.

Muriël de Groot, Mohammad Aliannejadi, and Marcel R. Haas. 2022. Experiments on generalizability of bertopic on multi-domain short text.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Adji B. Dieng, Francisco J. R. Ruiz, and David M. Blei. 2020. Topic Modeling in Embedding Spaces. *Transactions of the Association for Computational Linguistics*, 8:439–453.

Roman Egger and Joanne Yu. 2022. A topic modeling comparison between lda, nmf, top2vec, and bertopic to demystify twitter posts. *Frontiers in sociology*, 7:886498.

Charles A. Ellis, Mohammad S. E. Sendi, Eloy P. T. Geenjaar, Sergey M. Plis, Robyn L. Miller, and Vince D. Calhoun. 2021. Algorithm-agnostic explainability for unsupervised clustering.

Mariano Fernández-López, Asunción Gómez-Pérez, and Natalia Juristo. 1997. Methontology: From ontological art towards ontological engineering. Facultad de Informática (UPM).

Aaron Fisher, Cynthia Rudin, and Francesca Dominici. 2019. All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously.

Maarten Grootendorst. 2022. Bertopic: Neural topic modeling with a class-based tf-idf procedure.

Oumaima A. Ismaili, Vincent Lemaire, and Antoine Cornuéjols. 2014. A supervised methodology to measure the variables contribution to a clustering. *Neural Information Processing*, pages 159–166.

Ian T. Jolliffe. 2002. *Principal component analysis*, 2nd ed. edition. Springer series in statistics. Springer, New York.

Rafiullah Khan, Kieran McLaughlin, David Laverty, and Sakir Sezer. 2017. Stride-based threat modeling for cyber-physical systems. pages 1–6. IEEE.

Seyedfarzan Kolini and LECH Janczewski. 2017. Clustering and topic modelling: A new approach for analysis of national cyber security strategies. In *Twenty First Pacific Asia Conference on Information Systems*.

Rajesh Kumar, Siddharth Sharma, Chirag Vachhani, and Nitish Yadav. 2022. What changed in the cybersecurity after covid-19? *Computers & Security*, 120:102821. ID: 271887.

Scott M Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 4765–4774. Curran Associates, Inc.

Leland McInnes, John Healy, and Steve Astels. 2017. hdbscan: Hierarchical density based clustering. *Journal of open source software*, 2(11):205.

Leland McInnes, John Healy, and James Melville. 2018. Umap: Uniform manifold approximation and projection for dimension reduction.

MITRE. ATT&CK. `https://attack.mitre.org/`. Accessed: 2024-04-14.

MITRE. 2023a. CAPEC. `https://capec.mitre.org/`. Accessed: 2024-04-14.

MITRE. 2023b. CVE. `https://cve.mitre.org/`. Accessed: 2024-04-14.

MITRE. 2024. CWE. `https://cwe.mitre.org/`. Accessed: 2024-04-14.

Bayode Ogunleye, Tonderai Maswera, Laurence Hirsch, Jotham Gaudoin, and Teresa Brunsdon. 2023. Comparison of topic modelling approaches in the banking context. *Applied sciences*, 13(2):797.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(85):2825–2830.

Alain Rakotomamonjy. 2003. Variable selection using svm based criteria. *Journal of machine learning research*, 3(null):1357–1370.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: sentence embeddings using siamese bert-networks.

Michael Röder, Andreas Both, and Alexander Hinneburg. 2015. Exploring the space of topic coherence measures. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*, WSDM '15, page 399–408, New York, NY, USA. Association for Computing Machinery.

Peter J. Rousseeuw. 1987. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65.

Christian A. Scholbeck, Henri Funk, and Giuseppe Casalicchio. 2022. Algorithm-agnostic interpretations for clustering.

H. Sebastian Seung and Daniel D. Lee. 1999. Learning the parts of objects by non-negative matrix factorization. *Nature (London)*, 401(6755):788–791.

Adam Shostack. 2014. *Threat modeling: designing for security*. Wiley.

Hatma Suryotrisongko, Hari Ginardi, Henning Titi Ciptaningtyas, Saeed Dehqan, and Yasuo Musashi. 2022. Topic modeling for cyber threat intelligence (cti). In *2022 Seventh International Conference on Informatics and Computing (ICIC)*, pages 1–7.

Mike Uschold and Michael Gruninger. 1996. Ontologies: principles, methods and applications. *Knowledge engineering review*, 11(2):93–136.

Erik Štrumbelj and Igor Kononenko. 2014. Explaining prediction models and individual predictions with feature contributions. *Knowl. Inf. Syst.*, 41(3):647–665.

Hajar Zankadi, Abdellah Idrissi, Najima Daoudi, and Imane Hilal. 2023. Identifying learners' topical interests from social media content to enrich their course preferences in moocs using topic modeling and nlp techniques. *Education and information technologies*, 28(5):5567–5584.