Safety and Security Analysis using LDA based on Case Reports: Case Study and Trust Evaluation Method

Katsuyuki Umezawa
Shonan Institute of Technology / National Institute of AIST
Fujisawa, Kanagawa, Japan
umezawa@info.shonan-it.ac.jp

Hiroki Koyanagi
Internet Initiative Japan Inc.
Chiyoda-ku, Tokyo, Japan
h-koyanagi@iij.ad.jp

Sven Wohlgemuth
SECOM Co., Ltd.
Shibuya-ku, Tokyo, Japan
s-wohlgemuth@secom.co.jp

Yusuke Mishina
National Institute of AIST
Koutou-ku, Tokyo, Japan
yusuke.mishina@aist.go.jp

Kazuo Takaragi
National Institute of AIST
Koutou-ku, Tokyo, Japan
kazuo.takaragi@aist.go.jp

ABSTRACT

There are many cases where the safety and security of systems are threatened by accidental or intentional human error. This study focuses on the fact that there is information available about human error in design and operation documents and case reports, and they are in natural language. Therefore, we propose a method to analyze the impact of human error on safety and security using Latent Dirichlet Allocation (LDA), which is one of the topic model methods. First, we matched the given information to create a list of similarities (co-occurrence list) between documents. Based on this co-occurrence list, a fault and attack tree was constructed. While manually considering them, the critical points were identified through sensitivity analysis. We show the effectiveness of this proposed method through two characteristic case studies of cyber-based connected car design deficiencies and physical-based manufacturing inspection fraud. Both analyzes add a way to leverage big data interoperability in manufacturing processes using the IoT.

CCS CONCEPTS

• **Security and privacy → Vulnerability management**; • **Computing methodologies → Natural language processing**.

KEYWORDS

security, topic model, natural language processing, vulnerability information

ACM Reference Format:
Katsuyuki Umezawa, Hiroki Koyanagi, Sven Wohlgemuth, Yusuke Mishina, and Kazuo Takaragi. 2022. Safety and Security Analysis using LDA based on Case Reports: Case Study and Trust Evaluation Method. In *The 17th International Conference on Availability, Reliability and Security (ARES 2022), August 23–26, 2022, Vienna, Austria.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3538969.3538993

ARES 2022, August 23–26, 2022, Vienna, Austria

ACM ISBN 978-1-4503-9670-7/22/08…$15.00
https://doi.org/10.1145/3538969.3538993

1 INTRODUCTION

Conventionally, in the field of safety engineering, methods have been developed to learn from past accident cases and connect them to future accident prevention. For example, a method of applying natural language processing to 4,469 accident reports on aviation accidents accumulated in the National Aeronautics and Space Administration (NASA) to semi-automatically extract future accident candidates similar to the text description of past accidents has been done [1]. It is said that this method can be used to efficiently perform safety analysis by Event Tree Analysis, Fault Tree Analysis, and Hazard and Operability Study (HAZOP). In particular, it is effective in extracting human errors. However, conventional safety engineering alone cannot solve safety and security problems.

The following are problems in the safety and security evaluation of the supply chain.

(1) Damage to the safety and security of products and services due to human causes affects the overall risk.
(2) The reference database used for safety and security evaluation becomes significantly complicated when the intentional misconduct of human beings is included.
(3) In relation to safety and security risks, IoT-based systems have a large number of attack surfaces.

This paper has focused on human causes and considered solutions to these problems.

2 SAFETY AND SECURITY EVALUATION SCHEME

2.1 Risk Assessment Methodology

When evaluating safety and security related to human causes in the supply chain, it is crucial to consider defects that can occur due to accidental and intentional factors. It is common to think that the risk R related to a defect increases in proportion to the probability and the amount of damage, as shown below,

R = probability of event occurrence × size of loss per event occurrence.

In the case of evaluation based on accidental factors, the assurance level of quantitative assessment is relatively high because it

is often possible to give a fault probability based on solid statistics in the field of natural science. Accidental human error with a lot of statistical data can also fall into this category. Nevertheless, an attack such as intentional data modification is a cognitive problem, because the variation in statistics is large and the guarantee level of quantitative evaluation is low. To clarify legal liability and compensation for damages in the event of an accident, the risk assessment process for accidental and intentional factors with different guarantee levels must conducted out separately. The former context is represented by a fault tree (FT), and the latter is represented by an attack tree (AT).

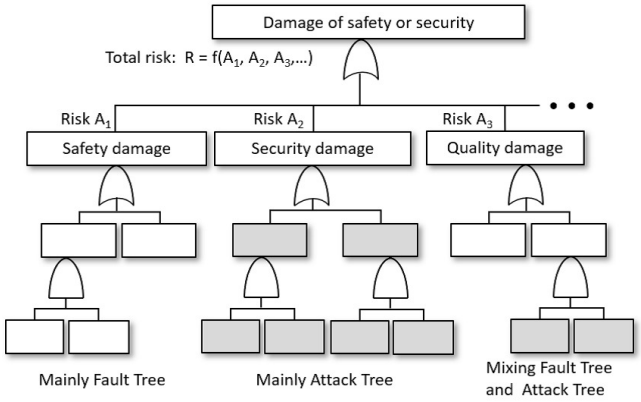


Figure 1: Damage to Safety and Security due to Human Causes

A multi-path process occurs in which different evaluation scales are examined separately to achieve safety and security. Then, the risks are comprehensively evaluated as a whole, and countermeasures are taken.

Let $A_i (i = 1, 2, 3, \dots)$ be the risks associated with the above evaluation scales, respectively. The combined risk R is a function of $A_1, A_2, \dots$, such as $R = f(A_1, A_2, \dots)$. The function f has various evaluation expressions depending on the business environment, such as a simple sum $f(A_1, A_2, \dots) = \sum_{i=1}^{n} A_i$ or a simple vector value $f(A_1, A_2, \dots) = (A_1, A_2, \dots)$.

2.2 Distribution Methods of Risk Assessment

In the supply chain, the safety and security guarantee obtained in this way is sent, received, and certified among the entities. Therefore, it is effective to use the PKI certification infrastructure, which is being developed as a social infrastructure in Japan, the United States, and Europe. For example, Figure 2 shows the usage of a certification path based on the public key infrastructure (PKI) certification [2].

Here, the Trust Anchor is a certification authority that is the base of trust on the certification path from the certificate to be verified to the trusted certification authority. Certificate B (A) from Figure 2 is a certificate issued to B and signed by A. The security of the certification mechanism itself (PKI certification) must be ensured in the process. For example, there is a high need to perform common vulnerability and exposures (CVE) checks reliably, which will be described later.

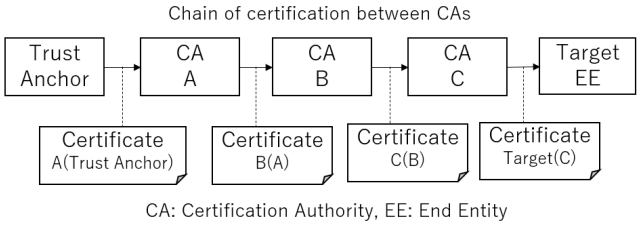


Figure 2: Example of a Certification Path of PKI

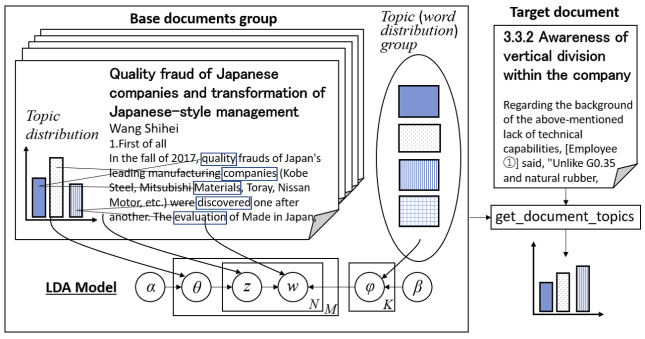
3 NATURAL LANGUAGE PROCESSING

3.1 Overview of LDA Model

In evaluations that include several human factors, we refer to the physical approach to failure and past records of human behavior.

In this proposal, we use Latent Dirichlet Allocation (LDA) [3], which compares sentences based on statistical theory. Usually, when a literature search is done, it is useful if a keyword matching search can also be done. Similarly, it would be very useful if we could specify a sentence consisting of a group of sentences and perform a matching search for sentences that are semantically similar to those sentences. This proposal uses LDA for such a "sentence matching search."

In LDA, we first focused on the sentences that existed in advance. Then we focused on the "base documents group" in the center of Figure 3. The word appearance frequency $tf_{i,j}$ is calculated for each document as follows:



M: number of documents, N: number of words in a given document, α: parameter of the Dirichlet prior on the per-document topic distributions, β: parameter of the Dirichlet prior on the per-topic word distribution, θ: topic distribution for document, φ: word distribution for topic, z: topic for the word in document, w: specific word

Figure 3: LDA Model

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}}$$

Here, $tf_{i,j}$ is the term frequency of the word t_i in the document d_j, $n_{i,j}$ is the number of occurrences of the word t_i in the document d_j, and $\sum_k n_{k,j}$ is the sum of the number of co-occurrences of all words in the document d_j. Also, the inverse document frequency idf is calculated as follows.

$$idf_i = log \frac{| D |}{| d : d \ni t_i |}$$

Here, idf_i is the inverse document frequency for the word t_i, $\mid D \mid$ is the total number of documents, and $\mid d : d \ni t_i \mid$ is the number of documents containing the word t_i.

The idf_i is a numerical value indicating the rarity of t_i. When t_i appears in many documents, the value decreases, and when it appears only in a specific document, the value increases. For example, for a word that appears in any sentence (such as "the" in English), idf has a minimum value of 0.

The product of these two numbers,

$$tfidf_{ij} = tf_{i,j} \cdot idf_i$$

indicates the weight of the word t_i in the document d_j. When more t_i words appear in the document d_j, the value of $tfidf_{ij}$ increases. Conversely, when the word t_i appears in another document, the value of $tfidf_{ij}$ decreases. In this way, the word distribution vector of the sentence is weighted with $tfidf_{ij}$, and then the LDA model is generated.

In the generation of the LDA model, the actualized data such as w: specific word, N: number of words in a given document, and M: number of documents in Figure 3 derived from the given sentence groups are input first. Next, it assumes the statistical distribution parameters α and β, which are the prerequisites for the appearance of such data. Then, a reverse estimation is performed to determine what the values of other state variables (which are considered to be latent) cause this manifested data. The result of this reverse estimation is the LDA model.

3.2 LDA Implementation

In this study, we implemented LDA using Gensim, a Python library (Figure 4) [4].

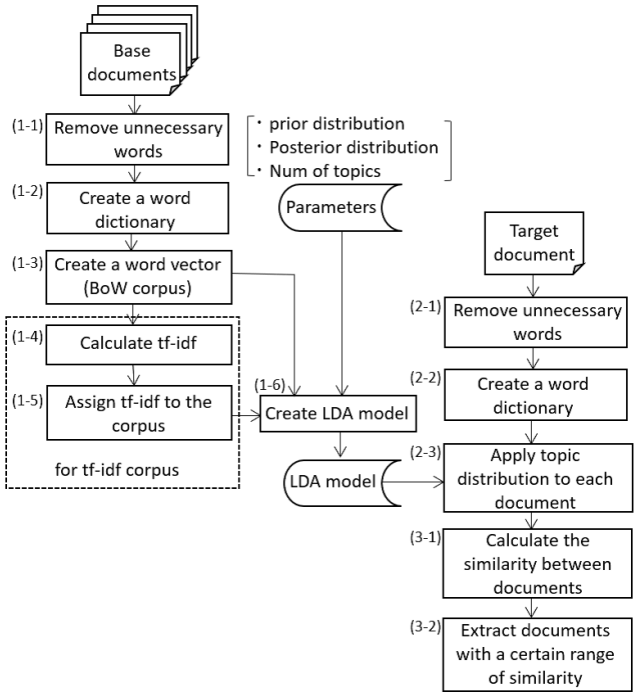


Figure 4: Overall Processing with the Proposed Tool

Gensim is licensed under the LGPL and is an open-source library for unsupervised topic modeling and natural language. This study used the source code published in the previous research [5] after adding a data reading unit and processing that excludes particles and adverbs of documents called stopwords. The details of the algorithm of the LDA generation process are as follows.

First, create an LDA model for the base document group using various parameters (See (1-1) to (1-6) in Figure 4). After creating the LDA model, assign a topic distribution to the evaluation target (See (2-1) to (2-3)). At the end of the process, the similarity is calculated (See (3-1) and (3-2)). See the appendix for details of each step and parameters.

4 CASE STUDY

4.1 Safety and Security Analysis Procedure

The procedure for safety and security analysis is as follows.

[**Step 1**] Perform matching analysis between the evaluation target and the base document group. Documents similar to the evaluation target are extracted from the base document group, and a list of the extracted document sets is created.

[**Step 2**] Create a fault or attack tree to be evaluated by referring to the list of extracted document sets. This step is done manually. Using Boolean algebra, we can use the KJ method [6] to expand the causal events of various intermediate events shown in the tree.

[**Step 3**] Give a probability to each event in the tree. However, this probability is given by an expert engineering judge based on the data accumulated in the company.

The following are assumed.

(1) The operation mechanism of the evaluation target is described in natural language.

(2) Each document in the base document group is written in natural language.

Under this assumption, to analyze using the LDA model, the amount of information in the text describing the evaluation target should be as large as possible. If possible, use detailed internal data such as design documents, test data, or human motion monitoring data owned by the company. However, these details are often not disclosed as they are considered trade secrets. This case study shows an example to exemplify the method using only published documents. The Japanese evaluation targets and base documents appearing in [Step 1] to [Step 2] were all translated into English by Google Translate.

4.2 Case Study 1: Industrial Rubber Inspection Fraud

Here, we evaluated the risk of industrial rubber inspection fraud in a manufacturing company (TG company). The evaluation target was the document [7] about the company, and the base document group was the document list [8] that describes manufacturers other than the company.

First, in [Step 1], we divided the description of the evaluation target [7] into 19 documents and the description of the base document group [8] into 77 documents. We performed LDA document matching for document pairs belonging to the former and the latter, and

obtained 19×77 similarities in the calculated. Table 1 shows typical parts of the similarity calculation results in [Step 1] (where the description in the base document is very similar to the description in the evaluation target).

Table 1: Results of LDA Matching in Case Study 1 [Step 1]

Base Document Group / Evaluation Target (TG company)	…	3.1 KS company	3.3 TR company	4.1 NI company	4.2 SU company	5 Summary: On-site fraud	…
3.2.3 Background of policy decision	…	0.698	0.760	0.407	0.389	0.231	…
3.3.4 Boss instructions and pressure from related departments	…	0.586	0.411	0.801	0.778	0.684	…
3.5.2 Unutilized risk management organization	…	0.247	0.125	0.548	0.500	0.912	…
3.5.2.3 Outside officer	…	0.871	0.646	0.557	0.567	0.272	…
3.5.4.1 Wishful thinking	…	0.853	0.324	0.449	0.443	0.614	…

Here, LDA was used in Gensim, a Python library, for the matching process in [Step 1]. The processing time was about 2 seconds. The documents used in this example were of A4 size of 98 (590 sentences and about 53,000 words) [7] and 17 (420 sentences and about 20,000 words) [8] pages respectively. If we try to perform the matching evaluation comprehensively using only visual inspection without the proposed tool, tens of thousands of cross-checks will occur, and even a skilled technician will take more than ten days.

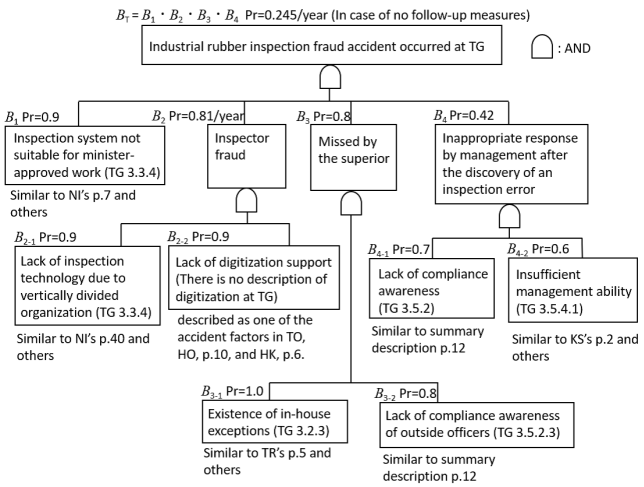


Figure 5: Fault Tree of Industrial Rubber Inspection Fraud

We manually performed [Step 2] and [Step 3] based on the similarity calculation results obtained in [Step 1]. The top event B_T in the FT (the industrial rubber inspection fraud accident that occurred at a TG company) was expressed as follows using a logical product.

$$B_T = B_1 \cdot B_2 \cdot B_3 \cdot B_4$$

Here, B_1 is "Inspection system not suitable for minister-approved work," B_2 is "Inspector fraud," B_3 is "Missed by the superior," and B_4 is "Inappropriate response by management after the discovery of an inspection error."

Furthermore, the details are expanded as shown in Figure 5.

When industrial espionage is negligible (Case 1), human error is evaluated with probability as an accidental event. In other words, in this FT, if no measures are taken after the fact, there is the probability that an "industrial rubber inspection fraud accident" will occur at the TG company once every four years. This evaluation result was input as a subtree in both "A_1: Safety damage" and "A_3: Quality damage" in the tree analysis of Figure 1.

When industrial espionage cannot be ignored (Case 2), some events were difficult to give with probability. In this case, after giving a different definition evaluation such as large, medium, and small as the probability, it was input to the subtree of "A_2: Security damage."

4.3 Case Study 2: Attack on Connected Cars

In this case, we evaluate the risk of attacking a connected car. Two sections of the document [9]. Browser Hacking (BH) section and Local Privilege Escalation (LPE) sections were used for evaluation target, and all cases (119,479 cases) of CVE database [10] were used as the base document group.

First, in [Step 1], LDA document matching was performed between the two sections to be evaluated and the description section written in the natural language of each CVE. A part of the result is shown in Table 2.

Table 2: Results of LDA Matching in Case Study 2 [Step 1]

Base Document Group / Evaluation Target (Reference [14])	…	CVE-2011-3512	CVE-2011-3928	CVE-2013-3774	CVE-2013-6282	CVE-2016-0566	…
Chapter: Browser Hacking	…	0.738	0.985	0.766	0.922	0.763	…
Chapter: Local Privilege Escalation	…	0.632	0.812	0.749	0.944	0.767	…

Table 2 is a horizontally long table with 119,479 items in the horizontal direction. Regarding the BH section in this table, the number of grayed parts (highly similar parts) was narrowed down to 1,514 out of 119,479 (overall ratio 1.3%). We narrowed the LPE section down to 5,180 cases (total discount 4.3%). CVE-2011-3928 and CVE-2013-6282 used in the actual attack were included in this narrowed down item. It also means that similar vulnerabilities that were not used but could be used in the future have been found.

We manually performed [Step 2] based on the similarity calculation results obtained in [Step 1]. From here, we arrive at the actual attacks CVE-2011-3928 and CVE-2013-6282 by semi-automatic analysis using DFD (data flow diagram) and manual checks (See Figure 6). Although a load of manual checks of 1,514 cases is not small, it is significant to automatically narrow down the initial candidate CVE description cases from 119,479 to 1,514 cases. If all the cases are manually matched, it will take several days.

We executed Python's open source library Gensim on Intel core i5 7500 for LDA calculation. The processing time was about 3 min. When the defect was extracted in this way, it was input as a subtree of "A_2: Security damage" in the tree analysis in Figure 1.

Since the matching process was performed only for a part of the evaluation target document (two sections this time), not all the

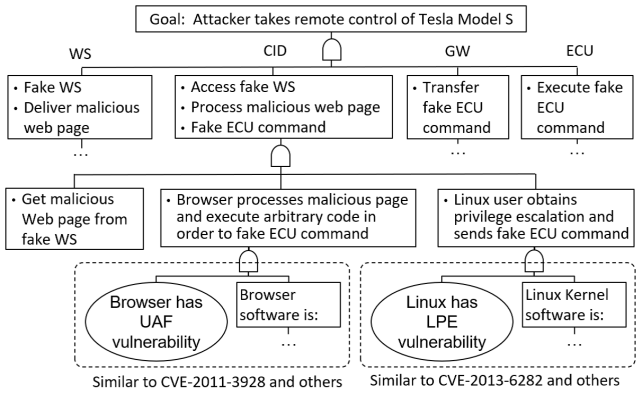


Figure 6: Attack Tree of Connected Car

trees can be described. Therefore, the provision of the probability in [Step 3] was omitted.

5 ESTABLISH TRUST

5.1 Building Framework for Trust Chain

It is assumed that the idea of how to evaluate the trust is widely shared and agreed upon by various communities. In such a case, for example, the proposed method can be applied as part of the process surrounded by the dotted line in the building framework for the trust chain [11] shown in Figure 7. The outline of the building framework for the trust chain is as follows. (1) Trust common requirements are defined as requirements that can absorb differences in industry, product, region, and nation and share trust standards across industries, products, regions, and nations. (2) Trust individual requirements for each value created by each provider according to the provider's situation and environment (regional, national, industrial sector, etc.) in the individual Value Creation Process can be defined. (3) In this way, the trusts of products and services with the same trust sharing requirements are compared and evaluated.

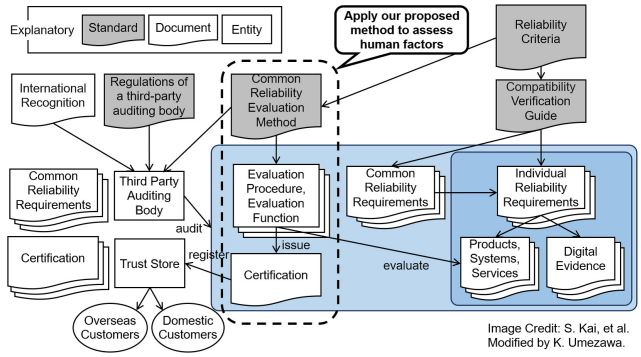


Figure 7: Building Framework for Trust Chain

5.2 US NIST Standards SP1500

To obtain trust in the supply chain, it is desirable to comply with trusted standards. Here we refer to the US National Institute of Standard and Technology (NIST) standard SP1500 [12], which contains many latest findings in the industry environment using IoT and big data. Each requirement is as follows.

Req. S_1 Requires improved security content automation tools such as SCAP, NBD-SPSL, etc., for scalability challenges, numerous false positives, and crippling information overload from the human computer interaction (HCI) perspective (p.44).

Req. S_2 Learn lessons from the aviation sector's extraordinary security level regarding Security and Privacy (pp. 26–27).

Req. S_3 Maintain a safety framework that can be made self-aware with human touchpoints and identify and monitor interactions (p. 73).

Req. S_4 Implement an automated dependency model that incorporates interoperating information security tools such as Security Information and Event Management (SIEM) (p. 79).

Req. S_5 Guarantee the authenticity and history of data while protecting personal information. (pp. 41–43)

Req. S_6 Keep in mind the management role that is responsible for registering human users (p. 116)

Req. S_7 Note the strong demand for compliance, and role of discipline, which cannot be overturned by compliance (p. 17, p. 39, p. 41, p. 44, p. 45, p. 100)

5.3 Relationship Between SP1500 and Case Studies

The relationship between these SP1500 requirements and the recurrence prevention measures for industrial rubber inspection fraud cases in Case Study 1 is shown in Table 3.

Table 3: Relationship between Fraud Measures for Industrial Rubber Inspection and SP1500

Industrial Rubber \ SP1500			S_1	S_2	S_3	S_4	S_5	S_6	S_7
B_T	B_1			appl.					appl.
	B_2	$B_{2\text{-}1}$		appl.				appl.	
		$B_{2\text{-}2}$	appl.		appl.	appl.			
	B_3	$B_{3\text{-}1}$		appl.					appl.
		$B_{3\text{-}2}$		appl.					appl.
	B_4	$B_{4\text{-}1}$		appl.					appl.
		$B_{4\text{-}2}$		appl.					appl.

By creating such a relationship, it is possible to show that the trust to be evaluated is being gauged by referring to a reliable reference model such as SP1500. This model in itself contributes to improving the accuracy of the evaluation content and ensuring trust. The measures to prevent attacks on connected cars shown in Case Study 2 can be positioned as compliance with SP1500 requirement S_1.

5.4 Relationship with SP800-126/SCAP

In addition, this proposed method can be positioned in relation to SP800-126/SCAP (Security Content Automation Protocol) [13] in

Figure 8. In this figure, SCAP is a tool that automatically checks the security of the evaluation target by referring to the CVE vulnerability database. This proposed method can be positioned as an additional tool for collation evaluation between the evaluation target, CVE, etc., in SCAP. SCAP is a tool developed by NIST in response to the requirement for automatic checking in many requirements of SP800-53 [14]. It is thought that referring to such a reliable standard improves the accuracy of the evaluation content and ensures trust.

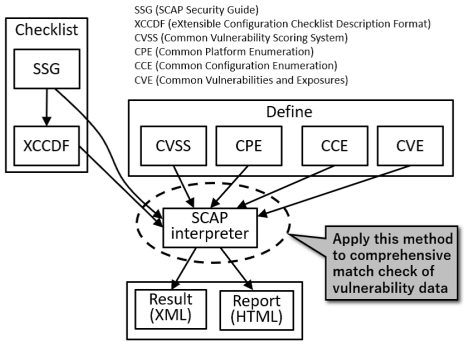


Figure 8: Relationship with SP800-126/SCAP

5.5 Combination of SP800-126/SCAP and SP1500

SCAP is primarily used to facilitate coordination between configuration settings and NIST SP800-53. However, this approach has not been designed with the appropriate speed or amount of big data security information. Integrating realtime logs with internal and external SCAP feeds can result in information overload due to scalability challenges, numerous false positives, and human–computer interaction (HCI). Information security tools can comprehensively handle risk assessment results obtained from third parties by using SCAP in conjunction with the application of LDA to SP1500 (see Figure 9). This model can address potential near realtime violations through automated reasoning. Additionally, it can indirectly affect networks and applications that need a combination of human and machine intelligence.

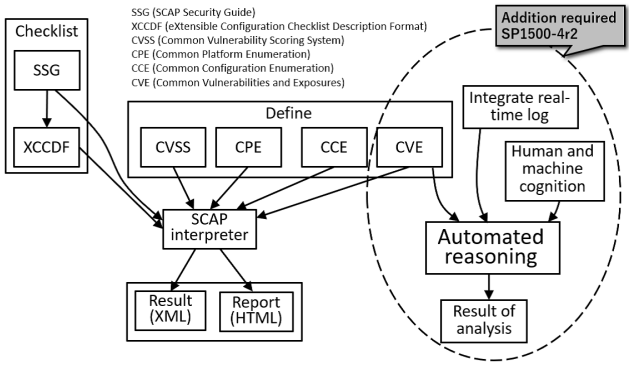


Figure 9: Combination of SP800-126/SCAP and SP1500

6 CONCLUSION

We focused on human factors in the supply chain and showed a method for evaluating safety and security using LDA natural language processing. We also showed the applicability to actual cases through case studies. Given the base documents and evaluation targets, we can comprehensively check with LDA. This effectively gives the evaluator awareness in maintaining and improving the safety and security of the evaluation target. However, reducing the amount of manual checking after a comprehensive check is a future task.

We also showed the correspondence between the proposed method and the related standardization, and showed the trust evaluation measures. Specifically, through a case study, we showed the relationship with the US standard SP1500 and the relationship with SP800-126/SCAP, which describe many best practices in industrial environments that utilize IoT and big data. The effectiveness of the proposed method was shown.

We believe our work could benefit from significant further explanation and scenarios presented to back up the premise and evaluate of effectiveness of our paper.

ACKNOWLEDGMENTS

This work was supported by the Cabinet Office (CAO), Cross-ministerial Strategic Innovation Promotion Program (SIP), “Cyber-Security for Critical Infrastructure” (funding agency: NEDO).

REFERENCES

[1] T. Nakata, “Text-Mining on Incident Reports to Find Knowledge on Industrial Safety,” IEEE Annual Reliability and Maintainability Symposium (RAMS 2017), January 2017.
[2] M. Cooper, et al., “Internet X.509 Public Key Infrastructure: Certification Path Building,” RFC 4158, September 2005.
[3] D. Blei, A. Ng, and M. Jordan, “Latent Dirichlet Allocation,” Journal of Machine Learning Research(2003), pp. 1107-1135.
[4] H. Koyanagi, Y. Mishina, K. Takaragi, S. Wohlgemuth and K. Umezawa, “Research on attack cases via topic-model analysis and selection of vulnerability candidates from large-scale vulnerability database,” International Workshop on Security (IWSEC) Poster Session, Fukui, Japan, September 2020.
[5] “Topic analysis by LDA with Gensim,” https://qiita.com/Spooky_Maskman/items/0d03ea499b88abf56819. (accessed January 28, 2021).
[6] J. Kawakita, “Way of Thinking – for creativity development,” Chuokoron-Shinsha. Inc. 1967.
[7] H. Higuchi, “Study of the Data Falsification Cases by Toyo Tire & Rubber Co. (TTR),” The Journal of Chiba University of Commerce 54 (1), pp. 57–98, September 2016.
[8] Z. Wang, “Nihon kigyou no hinshitsu fusei to nihon-teki keiei no henyo (Quality Fraud of Japanese Companies and Transformation of Japanese Management),” Keizai-to-keiei 50-1-2, Review of the Economics and Business Administration, Sapporo University, p.p. 33–49, May 2020.
[9] S. Nie, L. Liu, and Y. Du, “FREE-FALL: HACKING TESLA FROM WIRELESS TO CAN BUS,” Briefing, Black Hat USA 2017, July 2017.
[10] MITRE Corporation: CVE - Common Vulnerability and Exposure, https://cve.mitre.org/, (accessed January 28, 2021).
[11] S. Kai, Y. Isobe, K. Hirata, H. Shimizu, S. Hane, T. Atsumi, “Proposal of building framework for Trust Chain beyond the multi-sectors,” 2020 Simposium on Cryptography and Information Secuirty (SCIS 2020), January 2020.
[12] NIST Big Data Public Working Group, “NIST Big Data Interoperability Framework: Volume 4, Security and Privacy Version 3,” NIST Special Publication 1500-4r2, October 2019.
[13] D. Waltermire, S. Quinn, H. Booth, K. Scarfone and D. Prisaca, “The Technical Specification for the Security Content Automation Protocol (SCAP) SCAP Version 1.3,” NIST Special Publication 800-126 Revision 3, US DOC, February 2018.
[14] NIST JOINT TASK FORCE, “Security and Privacy Controls for Information Systems and Organizations,” Draft NIST Special Publication 800-53 Revision, US DOC, March 2020.
[15] J. Chang, S. Gerrish, C. Wang, J. L. Boyd-Graber, and D. M Blei, “Reading tea leaves: How humans interpret topicmodels,” Advances in NIPS, p.p. 288–296,

2009.
[16] D. Newman, J. H. Lau, K. Grieser and T. Baldwin, "Automatic Evaluation of Topic Coherence," p.p. 100–108, 2010.
[17] M. Roder, A. Both, and A. Hinneburg, "Exploring the Space of Topic Coherence Measures," Proceedings of the Eighth ACM International Conference on Web Search and Data Mining. Shanghai, WSDM. Association for Computing Machinery, February 2015.

A ABOUT LDA IMPLEMENTATION

A.1 Details of LDA Implementation

The details of the algorithm of the LDA generation process shown in Figure 4 are as follows.

(1-1) Read the description text of the base document group as array data. For every word, use the Morphy method of the wordnet class of nltk.corpus, which is a Python library, to restore the word to its original form. Furthermore, specify English in words method of the stopwords class of nltk.corpus and remove the stop words from the read sentences. Remove words that appear less than an arbitrary number of times (twice or less in this study) from the remaining text data.

(1-2) Create a word dictionary from the documents that have completed the above processing using the Dictionary method of the corpora class.

(1-3) Each document is converted into a vector of words from the created dictionary and document group by the doc2bow method of dictionary class (this process is called creating BoW corpus). If we want to perform weighting using tf-idf, perform the following (1-4) and (1-5).

(1-4) Create TfidfModel by TfidfModel method in Gensim's models class.

(1-5) The TF-IDF corpus is obtained by passing the BoW corpus created in (1-3) to the TfidfModel.

(1-6) Create an LDA model by specifying the dictionary and corpus (BoW corpus or tf-idf corpus) created in the LdaModel method of Gensim's ldamodel class and parameters.

After creating the LDA model, assign a topic distribution to the evaluation target. The flow is as follows.

(2-1) Perform the same processing as (1-1) to read the comparison document.

(2-2) Perform the same processing as (1-2), and set the evaluation target as a word vector.

(2-3) A vectorized document is given as an argument to the get_document_topics method of the LdaModel class, and a probability distribution is assigned.

At the end of the process in Figure 4, the similarity is calculated.

(3-1) The similarity of the topic distribution vectors $\vec{x}, \vec{y}$ of each document obtained by the flow from the base document and the flow from the evaluation target is the cosine similarity $cos(\vec{x}, \vec{y}) = \frac{\vec{x} \cdot \vec{y}}{|\vec{x}||\vec{y}|}$.

(3-2) Finally, the documents whose similarity exceeds the preset threshold value are extracted.

A.2 Perplexity and Coherence

It is necessary to give the "number of topics" in the parameters used in step (1-6) of the algorithm in the previous section. This section describes how to determine the number of topics. There are indicators for evaluating prediction accuracy and topic quality in topic modeling. Perplexity is an index of prediction accuracy, and coherence measures the topic quality.

Perplexity is obtained by calculating the geometric mean of appearance probability in the created model for all words. When finding it, the derivation method differs depending on the LDA sampling method. Coherence evaluates the quality of the topic and determines if it is easy for humans to interpret. However, there are many ambiguities, often lacking in validity. Therefore, various methods have been proposed. Examples are previous work [15][16]. There are several methods for calculating coherence in Gensim. This proposal adopted an algorithm called Cv according to the previous work [17].

Generally, the lower the value of perplexity, the better the accuracy; the higher the coherence, the better the quality of the topic. These two indicators are often in a trade-off relationship. For this reason, we adopted a number of topics with low perplexity and high coherence in this proposal.