

THE UNIVERSITY OF CHICAGO

COMPARING AND CONTRASTING SECURITY STANDARDS USING LARGE
LANGUAGE MODELS: A CASE STUDY OF NIST SPECIAL PUBLICATION 800-53

A THESIS SUBMITTED TO
THE FACULTY OF THE DIVISION OF THE PHYSICAL SCIENCES
IN CANDIDACY FOR THE DEGREE OF
BACHELOR OF SCIENCE

DEPARTMENT OF COMPUTER SCIENCE

BY
POLLY REN

CHICAGO, ILLINOIS

MAY 9, 2025

Copyright © 2025 by Polly Ren
All Rights Reserved

TABLE OF CONTENTS

LIST OF TABLES	v
ACKNOWLEDGEMENTS	vi
ABSTRACT	vii
1 INTRODUCTION	1
2 NIST AND NIST SP 800-53	4
3 BACKGROUND AND RELATED WORK	8
3.1 Semantic analysis with sentence embeddings	8
3.2 Large language models for textual analysis	8
3.2.1 Context windows	9
3.2.2 Hallucinations and abstentions	10
4 METHODOLOGY	12
4.1 Document parsing and control extraction	12
4.2 Control-level mapping using embeddings	14
4.2.1 Embedding model selection	15
4.2.2 Finding similar controls	15
4.2.3 Generating mappings	16
4.3 LLM-based comparison	17
4.3.1 Model choice and parameters	17
4.3.2 Prompt structure	17
4.3.3 Comparing iteratively	19
4.4 LLM-based summarisation	19
4.5 Evaluation	21
4.5.1 Revision 5 and Revision 4 comparison	21
4.5.2 Other revision comparisons	22
5 RESULTS AND DISCUSSION	23
5.1 Analysis of embeddings-based mappings	23
5.1.1 Backward mappings tend to be more diffuse	24
5.1.2 Mapping is largely asymmetric	26
5.2 Evaluation of LLM comparisons	28
5.2.1 The LLM correctly identifies most changes, even subtle changes	29
5.2.2 The LLM is more sensitive to smaller changes and more susceptible to omission for substantial changes	32
5.2.3 The LLM can sometimes interpret the text too literally	34
5.3 The evolution of NIST SP 800-53	36
5.3.1 An expansion in conceptual scope	36

5.3.2	An increase in modularity	37
5.3.3	Summary of changes	39
5.4	Limitations	39
6	CONCLUSION	41
6.1	Future work	41
	REFERENCES	44
A	PROMPT TEMPLATES	51
A.1	LLM comparison prompt template	51
A.2	LLM summarisation prompt template	52
B	LLM-GENERATED SUMMARIES	53
B.1	Revision 4, compared to Revision 5	53
B.2	Revision 3, compared to Revision 5	54
B.3	Revision 2, compared to Revision 5	56
B.4	Revision 0, compared to Revision 5	57

LIST OF TABLES

2.1	Revision history of NIST SP 800-53.	5
2.2	Document-level statistics for each revision.	6
2.3	Count of controls by control family across revisions.	7
5.1	Summary of control mapping metrics across revision comparisons.	23
5.2	Deductive coding categories used to evaluate LLM-generated comparisons. . . .	29
5.3	Coding counts for LLM comparison of Revision 5 against previous revisions. . .	29
5.4	LLM labels for control mappings.	36

ACKNOWLEDGEMENTS

My undergraduate years have been profoundly shaped by the ideas and support of many people, all of whom this project would not have been possible without. First and foremost, I would like to thank my advisor, Professor Grant Ho, for the opportunity to work on this project, and for his guidance and direction through the research and writing process. He always made time to discuss the project with me and was there whenever I needed his help, offering thoughtful feedback, suggesting new avenues to explore and challenging me to investigate each question more thoroughly. His kind and understanding mentorship not only strengthened this project but has also shaped the way I think about research.

I've had the great pleasure and privilege to learn from and work alongside many incredible teachers. I'd like to thank Professor Matthew Wachs, for the best introduction to computer science one could possibly ask for. I probably would not be studying computer science without his infectious enthusiasm that first drew me to the subject. His encouragement and support also made my foray into computer science teaching a fun and fulfilling one, and working with him continues to inspire my curiosity and interest in the field. I am also grateful for Professors Adam Shaw and Borja Sotomayor, who, too, have endlessly given me the patience, trust and space to grow my teaching. My approach to learning, teaching and thinking about computer science will always bear their influence, along with the guidance and inspiration of so many other professors in the department I've had the remarkable fortune to learn from and interact with.

I would also like to thank my peers and colleagues, for the collaboration, cheerleading, commiseration, laughter and friendship; it is hard to imagine debugging in Crerar after hours with any other community of people. Last but certainly not least, I will always be thankful for my family, friends and roommates, for their unwavering support and for making this entire journey an enjoyable, enriching and memorable one.

ABSTRACT

Tracking the evolution of security standards is essential for ensuring regulatory compliance and maintaining robust cybersecurity practices. This paper presents a scalable, data-driven methodology for comparing revisions of the NIST Special Publication 800-53 using sentence embeddings and large language models (LLMs). Using a retrieval-optimised embedding model, we identify semantically similar controls across revisions and apply an LLM to classify and explain the relationships between mapped controls. Our evaluation shows that the LLM accurately identifies both fine-grained and conceptual differences, though it exhibits occasional omission of details and misinterpretations. We find that the structure and scope of SP 800-53 controls have grown significantly more modular and comprehensive over time, with new revisions introducing controls that lack direct counterparts in earlier versions. Our approach enables interpretable analyses of semantic changes in cybersecurity standards and provides a foundation for automating the comparison of other regulatory frameworks.

Keywords: Large language models, cybersecurity standards, sentence embeddings, natural language processing, document comparison

CHAPTER 1

INTRODUCTION

The growing dependence of modern society on digital systems has dramatically increased the importance of securing these systems. As government agencies, businesses and individuals increasingly rely on these technologies, the potential consequences of cybersecurity breaches have grown substantially, ranging from financial loss to national security threats [1]. In this environment, security standards play an important role in establishing best practices for protecting sensitive information, safeguarding critical systems and mitigating risks in today’s digital landscape.

Security standards serve several essential functions. They establish a baseline of minimum controls that organisations can implement to systematically reduce vulnerabilities. A control is a measure that modifies or maintains risk [8]. Security standards offer a common language for articulating security requirements across diverse sectors and stakeholders. They also provide a framework for assessing compliance, supporting audits, certifications and regulatory reporting.

Among the most influential cybersecurity standards is the National Institute of Standards and Technology (NIST) Special Publication 800-53 (SP 800-53). Originally developed for United States federal agencies in 2002 [28], NIST SP 800-53 has grown in scope and influence over the past two decades. The publication provides a comprehensive catalogue of security and privacy controls across a broad array of control families, addressing a variety of threats to “protect the confidentiality, integrity, and availability of the system and its information” [54]. Over successive revisions, the publication has expanded its focus from largely technical controls to encompass broader organisational governance and privacy protection, with controls ranging from technical to operational to management domains. Today, NIST SP 800-53 serves not only as a foundation for federal cybersecurity but also as a widely referenced security framework for critical infrastructure, and private and non-governmental

organisations [6, 40].

While the overall structure of SP 800-53 has remained relatively stable across revisions, the content of the controls has evolved over the years, in tandem with new threats and technological advances. As a result, individual controls are often added, removed, rephrased, split into multiple controls or merged. For organisations that seek to maintain compliance or align their internal security framework with updated standards, understanding how controls change between revisions becomes a challenging but necessary task. Tracking changes between revisions is, however, not straightforward. Although new revisions generally describe the overall changes at a high level, there is often no fine-grained, control-level mapping available. As such, it is difficult to identify how a control may be split up across multiple controls or combined with other controls into different sections. Moreover, changes are not limited to the grouping of controls; there may be other semantic shifts in phrasing, scope or technical requirements without obvious structural indicators.

Traditionally, organisations and auditors have relied on manual comparison of documents to identify differences in standards and track changes across revisions—a process that is both time-consuming, inefficient, costly and prone to human error [25, 18, 12, 20, 31, 11]. Some approaches have employed automation for compliance verification and log analysis [61, 27], but the emergence of natural language processing (NLP) techniques, particularly sentence embeddings and large language models (LLMs), offers new possibilities for systematically comparing standards and their controls.

In this project, we leverage these advances to develop a scalable methodology for comparing controls across revisions of NIST SP 800-53. This paper empirically investigates how LLMs can be used to build an automated system for mapping and identifying changes in security advice, and how effectively they can classify the nature of these changes with accuracy and interpretability. As a case study, we apply this methodology to NIST SP 800-53, examining how the content of its security controls has evolved across successive revisions.

Our analysis focuses on identifying and characterising the changes in security advice over time, examining the extent to which controls are added, removed, modified, restructured or remain stable across revisions.

CHAPTER 2

NIST AND NIST SP 800-53

The National Institute of Standards and Technology (NIST), founded as the National Bureau of Standards (NBS) in 1901, is one of the United States’ first physical science laboratories, established in response to the growing need for a standardised measurement infrastructure to maintain industrial competitiveness [9, 10, 2]. Over time, with the growing digitisation of government and economic infrastructures and the increasing need for cybersecurity, NIST’s role expanded to include the development of standards, guidelines and best practices to ensure the trust and security of information systems. Today, NIST is a non-regulatory agency under the United States Department of Commerce [3].

As a non-regulatory agency, NIST lacks enforcement authority. Nevertheless, its publications have become de facto standards across both the public and private sectors. This widespread adoption is due, in part, to their formal integration into federal law and policy, particularly the Federal Information Security Management Act (FISMA) of 2002 [28] and the Office of Management and Budget (OMB) Circular A-130 [40]. Under these legislations, federal agencies are required to develop and implement risk-based information security programs that ensure the confidentiality, integrity and availability of information [2].

One of NIST’s most influential contributions to federal cybersecurity policy is SP 800-53. First released in 2005 in response to FISMA’s directive to create standards for securing federal systems, SP 800-53 provides a comprehensive catalogue of security and privacy controls designed to protect federal information systems against a wide range of threats. Although the publication was prepared for use by federal agencies, the influence of SP 800-53 extends far beyond the federal government and its immediate agencies, as many organisations interact with federal agencies and adopt its controls voluntarily. Its controls offer a baseline for security and privacy suited toward organisational operations, assets and personnel, recognising that organisational and industry differences make it difficult to have a one-size-fits-all

standard. More recently, NIST also publishes companion documents to SP 800-53—SP 800-53A and SP 800-53B—which provide detailed procedures for assessing the effectiveness of controls and offer predefined security baselines for different system impact levels. Together, these documents form a cohesive suite that supports the specification of controls, as well as their implementation, assessment and continuous monitoring. This research will focus exclusively on NIST SP 800-53.

Table 2.1: Revision history of NIST SP 800-53.

Rev.	Title	Release
0*	<i>Recommended Security Controls for Federal Information Systems</i>	Feb 2005
1	<i>Recommended Security Controls for Federal Information Systems</i>	Dec 2006
2	<i>Recommended Security Controls for Federal Information Systems</i>	Dec 2007
3	<i>Recommended Security Controls for Federal Information Systems and Organizations</i>	Aug 2010
4	<i>Security and Privacy Controls for Federal Information Systems and Organizations</i>	Apr 2013
5	<i>Security and Privacy Controls for Information Systems and Organizations</i>	Sep 2020

* Revision 0 is not labelled as such by the publication, but we will refer to the original release of NIST Special Publication 800-53 as “Revision 0” throughout this paper.

Since its initial release in February 2005 as *Recommended Security Controls for Federal Information Systems*, NIST SP 800-53 has undergone several major revisions, reflecting overall shifts in the technological landscape, emerging threat vectors and ever-evolving expectations around privacy and risk management. While the overarching structure has remained relatively consistent, the content and organisation of the publication have changed significantly over time. For one, the evolution of the title of NIST SP 800-53 reflects the broadening scope of the publication over time. Revisions 0 through 2 were all titled *Recommended Security Controls for Federal Information Systems*, emphasising a focus on technical security protections applied within federal agencies. With Revision 3, the title was expanded to *Recommended Security Controls for Federal Information Systems and Organizations*, signalling a shift toward broader organisational governance and enterprise risk management. In Revision 4, the title was updated again to *Security and Privacy Controls for Federal Information Systems and Organizations*, formally recognising the increasing integration of privacy concerns into the existing security frameworks. In the most recent Revision 5, the

qualifier “Federal” is removed entirely, reflecting NIST’s intent to provide a more universally applicable framework that can be adopted beyond federal information systems. These title changes illustrate the standard’s progression from a narrower technical document to a more comprehensive and adaptable control framework for a diverse array of operational contexts.

Along with the changes in the publication’s title, each revision has also brought significant updates to the individual controls and control families. The document-level statistical changes can be seen in Table 2.2. The original release of the document presented seventeen control families, and that number has steadily increased over time ¹ alongside the number of controls. This reflects the new technologies, such as cloud and mobile technologies, that the standard seeks to account for, and the new threat landscapes and regulatory priorities that emerge. Moreover, control descriptions have become increasingly outcome-oriented and technology-neutral over time. Rather than prescribing specific mechanisms or tools, later revisions focus on the desired security and privacy outcomes by parameterising controls and granting organisations greater flexibility to implement appropriate safeguards suited to their specific environments. This modular approach allows SP 800-53 to remain relevant across various governmental and organisational contexts.

Table 2.2: Document-level statistics for each revision.

Metric	Rev. 0	Rev. 1	Rev. 2	Rev. 3	Rev. 4	Rev. 5
Page count	85	174	189	238	462	492
Word count	12,362	18,611	18,605	27,670	49,471	62,657
Average control length (words)	590.21	840.73	840.89	1069.62	1441.93	1582.88
Number of control families	17	17	17	18	26	20
Average control count per family	9.59	10.06	10.06	11.0	10.19	14.90

1. This increasing number of control families is not observed in the changes from Revision 4 to Revision 5. This is because Revision 4 introduced several new control families for privacy-related controls, including Authority and Purpose; Accountability, Audit, and Risk Management; Data Quality and Integrity; and Data Minimization and Retention. These controls within these privacy-related control families are incorporated into other control families in Revision 5, hence, the decrease in the number of control families between these revisions.

Table 2.3: Count of controls by control family across revisions.

Control Family	Rev. 0	Rev. 1	Rev. 2	Rev. 3	Rev. 4	Rev. 5
Access Control (AC)	20	20	20	19	23	23
System and Information Privacy (AP)	—	—	—	—	2	—
Authority and Purpose (AR)	—	—	—	—	8	—
Awareness and Training (AT)	4	5	5	5	4	5
Audit and Accountability (AU)	11	11	11	14	16	15
Assessment, Authorization, and Monitoring (CA)	7	7	7	6	8	8
Configuration Management (CM)	7	8	8	9	11	14
Contingency Planning (CP)	10	10	10	9	12	12
Inventory and Device Management (DI)	—	—	—	—	2	—
Data Management (DM)	—	—	—	—	3	—
Identification and Authentication (IA)	7	7	7	8	11	12
Individual Participation (IP)	—	—	—	—	4	—
Incident Response (IR)	7	7	7	8	10	9
Maintenance (MA)	6	6	6	6	6	7
Media Protection (MP)	7	6	6	6	8	8
Physical and Environmental Protection (PE)	17	19	19	19	19	22
Planning (PL)	5	6	6	5	6	8
Program Management (PM)	—	—	—	11	16	32
Personnel Security (PS)	8	8	8	8	8	9
Privacy Authorization (PT)	—	—	—	—	—	8
Risk Assessment (RA)	5	5	5	4	5	9
System and Services Acquisition (SA)	11	11	11	14	20	16
System and Communications Protection (SC)	19	23	23	34	41	47
Security and Privacy Requirements (SE)	—	—	—	—	2	—
System and Information Integrity (SI)	12	12	12	13	16	22
Supply Chain Risk Management (SR)	—	—	—	—	—	12
Transparency (TR)	—	—	—	—	2	—
Use Limitation (UL)	—	—	—	—	2	—
Total count	163	171	171	198	265	298

CHAPTER 3

BACKGROUND AND RELATED WORK

This chapter describes the background for our methodology and discusses prior work related to performing semantic analysis with sentence embeddings, and the utility and limitations of applying large language models for textual analysis.

3.1 Semantic analysis with sentence embeddings

Semantic analysis aims to understand and compare the meaning of textual data. In recent years, sentence embedding models such as Sentence-BERT (SBERT) have emerged as a powerful method for encoding textual content as a fixed-length vector [44, 23]. Sentence embeddings involve using pre-trained models to map each sentence to a vector space such that sentences that are semantically similar are close to one another in the vector space. These vector representations enable comparison of sentence meaning using computational measures like cosine similarity or vector dot products, which facilitates downstream tasks such as document clustering, semantic search and question answering [23, 44, 14]. Embeddings have been widely studied for use in real-world information retrieval applications, including enterprise semantic search engines [51], duplicate question detection on online forums [55] and open-domain question answering systems [42]. In regulatory contexts, they have been used for similarity analysis of legal documents [22, 19], and to measure policy preferences [34].

3.2 Large language models for textual analysis

Large language models (LLMs) have shown immense promise in natural language processing tasks, particularly in the summarisation of complex texts and documentation [63, 69, 62]. These models, trained on diverse language data, are capable of understanding, analysing and generating human-like text with a high degree of fluency and contextual awareness. As

a result, they have become foundational tools in domains that involve processing large volumes of semi-structured language data, including legal analysis and clinical documentation review [15, 76, 13, 41, 26, 17]. Unlike keyword- or rule-based systems, LLMs can contextually identify relationships between passages that use different phrasing but express similar ideas. This makes them especially useful for document comparison tasks, where shifts in sentence structure may obscure underlying conceptual continuity with traditional approaches.

Recent advances have further improved the utility of LLMs for these purposes. Larger context windows and instruction-tuned models enable better multi-passage reasoning [75, 59], allowing models to retain consistency across longer inputs. These capabilities open up new opportunities for developing automated pipelines that compare and track changes across long-form documents. Prior work has shown the effectiveness of LLMs in identifying and categorising disclosures found in privacy policies [52], as well as automating technical privacy reviews [17]. In this project, we leverage these capabilities to apply LLMs to the domain of cybersecurity standards, with a focus on identifying semantic change, overlap and divergence across revisions of NIST SP 800-53.

3.2.1 Context windows

LLMs operate within a context window, which defines the maximum number of tokens—units that roughly correspond to words or subwords—that the model can process at one time when generating outputs. The size of the context window directly constrains the amount of information the model can analyse and reason about at a given time. Earlier models were constrained by relatively small context windows (GPT-3.5-turbo, for instance, had a context window of 16,385 tokens), meaning they could only process a document in smaller excerpts before losing awareness of preceding content. However, the recent GPT-4o and o-series models from OpenAI have context windows of 128,000 and 200,000 tokens [5], respectively,

which are large enough to compare the entirety of many security standards¹, without the need for significant chunking. In parallel, substantial research efforts have explored techniques to extend LLM context windows even further, achieving context window sizes of over a million tokens [50, 29, 24, 68, 39]. These methods often involve architectural innovations such as recurrent memory, retrieval-augmented generation or efficient sparse attention mechanisms. However, while context window expansion addresses technical constraints, prior research has found that model performance tends to degrade as input length increases [47, 72, 35, 45, 77]. Long documents can overwhelm model attention mechanisms and dilute focus on relevant portions of the input, leading to inaccuracies in reasoning and generation. As such, even though models like GPT-4o can technically accommodate the comparison of large security documents, it remains beneficial to minimise input size wherever possible to enhance performance. Accordingly, in this work, we focus on optimising input preparation by isolating relevant control and discussion content, allowing the LLM to reason more effectively and precisely over the inputs.

3.2.2 *Hallucinations and abstentions*

Importantly, LLMs are not infallible. A well-documented challenge in their deployment is the phenomenon of hallucination, where the model outputs unsubstantiated claims or outright false or logically inconsistent responses that may sound fluent or plausible [30, 36]. Hallucinations are understood to have a number of different causes, including misinformation and biases, knowledge boundaries, overconfidence and reasoning failures [37]. Because LLMs generate responses based on statistical patterns, some degree of hallucination is widely regarded as inevitable [70, 16], even if the model has correct knowledge [57]. Abstention, or

1. For reference, NIST SP 800-53 Revision 5 is about 90,000 tokens in length, and Revision 0 is approximately 18,000 tokens in length. This count includes only the control information and the discussion that immediately follows each control; it does not account for control enhancements or additional context that precedes the controls appendix. Including these enhancements and additional background and context increases the Revision 5 token count to about 325,000, which is well above the context window of models such as GPT-4o.

the ability of a model to refrain from generating a response when uncertain, is an important technique to mitigate these issues [67, 60]. While models tend to have trouble abstaining from binary answers [65], they are also highly sensitive to prompt wording [66, 48]. Techniques that explicitly instruct the model to evaluate its own confidence or uncertainty before responding have shown promise in encouraging more reliable abstentions [73, 71]. In using LLMs to compare security advice, we will also attempt to better understand the challenges and limitations of their use for such tasks, as well as techniques that improve their ability to abstain appropriately when faced with uncertainty.

CHAPTER 4

METHODOLOGY

We adopt a multi-step methodology to analyse semantic changes across revisions of NIST SP 800-53, combining structured document parsing, embedding-based control mapping and LLM-based comparison and summarisation. This chapter describes the methodology used to build this automated system, how we used this system to identify changes in SP 800-53 and how the correctness and interpretability of this system was evaluated. In particular, we compare Revision 5 against Revision 4, Revision 3, Revision 2 and Revision 0¹. A visual overview of our methodology is shown in Figure 4.1.

4.1 Document parsing and control extraction

Historical and current NIST revisions are available on the official NIST government website, in the Computer Security Resource Center. We sourced all six revisions of NIST Special Publication 800-53 from this repository. The documents were provided in Portable Document Format, so we began by parsing each revision into a consistent JSON format suitable for downstream semantic comparison.

Controls in NIST SP 800-53 are structured hierarchically: each top-level control (e.g., AC-2) may be followed by one or more enhancements (e.g., AC-2(1), AC-2(2)), which provide optional or situational refinements to the main control. While these control enhancements have improved substantially and expanded in scope and technical precision from the first revision to the current revision, they are inconsistent across all revisions. Though the content of the control may be similar between revisions, the absence of control enhancements in earlier revisions can introduce structural asymmetry and confound attempts to track continuity in the fundamental intent and coverage of the standards over time. As such, for purposes of

1. Revisions 1 and 2 are nearly identical in content and structure, aside from a few minor changes in punctuation and grammar, so we exclude Revision 1 from further comparison to reduce redundancy.

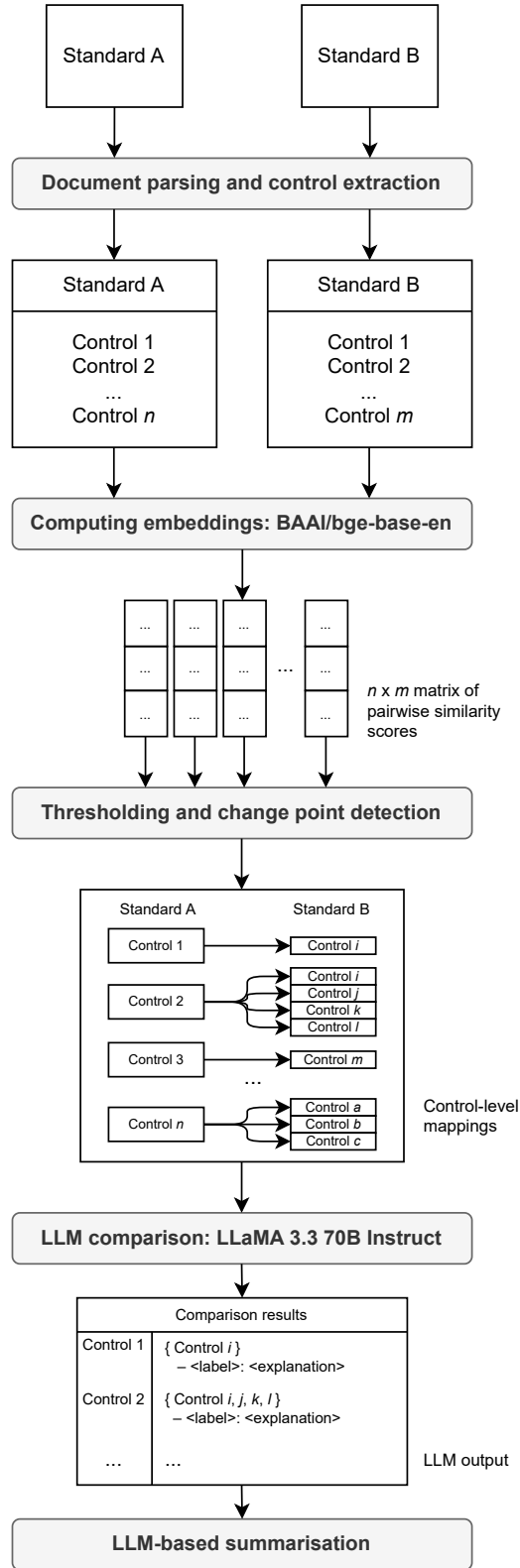


Figure 4.1: Overview of the thesis methodology.

our embeddings- and LLM-based analyses, we excluded all control enhancements, focusing exclusively on top-level controls. Excluding these control enhancements helps reduce variance in control granularity and ensures more equitable comparisons across standards.

In addition to the control text for each top-level control, we included the accompanying Discussion (Revision 5) or Supplemental Guidance (Revisions 0-4), when provided for each control. This section generally elaborates on the intent, context and implementation considerations of the control, which offers information beyond the terse language of the control itself. Since these explanatory sections are consistently present across revisions and largely consistent in length, their inclusion supports a more robust and context-aware comparison of controls over time. Including this supplementary content ensures that embeddings and LLM-based comparisons are informed by the full extent of each control, without too much additional variability in structure and specificity that further including control enhancements may introduce. This filtered, top-level-only control set and the accompanying discussion form the input for subsequent embedding and comparison stages.

4.2 Control-level mapping using embeddings

Embedding-based methods have been widely adopted in tasks like document retrieval, semantic search and paraphrase mining, and recent work has demonstrated their applicability in the alignment of legal and compliance texts [56, 49, 46]. Since the mapping between controls in one revision to the control in another revision is not one-to-one, we use embeddings to approximate the set of controls in one revision that are most semantically similar to each control in another. This enables a direct comparison between related, semantically similar controls across two different revisions.

4.2.1 *Embedding model selection*

We used the `BAAI/bge-base-en model`, available on Hugging Face [4], to transform each control and discussion text into a high-dimensional vector representation. This is the base general embeddings model; it is retrieval-optimised, meaning that output embeddings are close to one another in embedding space if the inputs are semantically similar. The model was accessed through the Sentence Transformer framework, which supports both encoding and retrieval workflows. The input to the model was formatted using a task-specific prefix ("**Represent this sentence for retrieval:**") to guide the model toward generating context-aware embeddings.

We also experimented with other sentence embedding models, but many other models produced results that failed to provide sufficient separation between semantically similar and dissimilar controls. In many cases, unrelated controls received moderately high similarity scores, resulting in ambiguous rankings that made reliable mapping difficult. In contrast, `bge-base-en` produced embeddings with sharper clustering behaviour and greater score differentials between relevant and irrelevant matches.

4.2.2 *Finding similar controls*

After using the embedding model to vectorise each control in both revisions, we use cosine similarity to compute a pairwise similarity metric between each control in Revision A and all the controls in Revision B. Because sentence embeddings map semantically similar sentences to vectors that are close to one another in the resulting vector space, computing the cosine similarity is appropriate for identifying the relatedness of two sentences. This involves taking the dot product between each pair of embeddings and unit-normalising the result. The output of this operation across all the controls in both standards is an $n \times m$ similarity matrix, where n and m represent the number of controls in Revision A and Revision B, respectively. If entry i, j has a numeric value close to unity, this indicates that control i in

Revision A is semantically similar to control j in Revision B; an analogous statement can be made for numeric values close to zero and semantic dissimilarity. This similarity matrix forms the basis of our control mapping strategy, capturing the semantic alignment between controls in different revisions.

4.2.3 *Generating mappings*

With this similarity matrix, we determine the mapping from each control in Revision A to the controls in Revision B. Rather than selecting a single matching control in Revision B for each Revision A control (for instance, by taking the corresponding control in Revision B with the maximum cosine similarity), we applied a hybrid approach to determine meaningful segments of high-similarity candidates. This approach enables one-to-many mappings, which account for controls that may be split into multiple granular controls between revisions, or for the consolidation of several overlapping controls into a single control. Because these mappings ultimately determine which controls get compared in the LLM-based comparison, restricting the mapping to a single match risks discarding relevant content and providing incomplete information in the downstream analyses.

Each control’s similarity vector was sorted in descending order, and the difference between adjacent values was analysed. We used both fixed threshold gaps and change point detection to detect shifts in similarity scores and segment controls accordingly. We used a fixed threshold of 0.02; if a drop of at least 0.02 was detected between adjacent similarity scores, the position of the drop was treated as a natural cutoff. For cases without clear drop-offs that satisfied this threshold, usually in cases where there was no direct match in the other revision, we used the ruptures library for change point detection. We used the PELT algorithm with an L2 cost model to identify change points in the similarity score distribution for each control. After both the fixed threshold gaps were analysed and change points were identified when necessary, we retained the first segment as the mapping for each control. This

is the segment containing the controls in the mapped-to revision with the highest similarity score, and hence, is the closest semantic match to the mapped-from control.

We performed this control-level mapping in both directions, mapping each Standard A control to one or more controls in Standard B, and vice versa. This bidirectional mapping helps capture the asymmetric relationship between controls, where a control in one revision may have no equivalent counterpart in the other, or where a control’s meaning was redistributed across multiple controls.

4.3 LLM-based comparison

Once each control in one revision is mapped to a corresponding set of candidate controls in another revision using the embedding-based approach, the next step is to determine the specific semantic relationship between each control and its corresponding mapped controls from the other revision. We used large language models for this task, prompting them to generate labels and explanations for the relationship between the controls.

4.3.1 *Model choice and parameters*

We used Meta’s LLaMA 3.3 70B Instruct model, accessed through the Together.ai API. This model was selected for its cost efficiency and performance. We invoked the model through a call to `client.chat.completions.create`. To ensure more consistent outputs and reduce variability in responses, we used a temperature of 0.0 to limit sampling randomness.

4.3.2 *Prompt structure*

We designed a prompt template that contextualises the task as a comparative analysis between two pieces of cybersecurity advice. The prompt presents the text of one control from Revision A alongside the corresponding mapped controls from Revision B obtained through

the embeddings-based mapping described above. The model is then asked to label the relationship of the advice from Revision A to the corresponding advice from Revision B into one of five categories: agrees, superset, subset, disagrees or incomparable. The criteria for each label are detailed within the prompt:

- **AGREES:** Excerpt A and Excerpt B offer compatible or equivalent guidance.
- **SUPERSET:** Excerpt A is broader or more comprehensive than Excerpt B, but some of the advice in Excerpt A is in Excerpt B. Excerpt B does not provide additional guidance beyond that in Excerpt A. That is, there are more control details in Excerpt A than in Excerpt B.
- **SUBSET:** Excerpt B is broader or more comprehensive than Excerpt A, but some of the advice in Excerpt B is in Excerpt A. Excerpt A does not provide additional guidance beyond that in Excerpt B. That is, there are more control details in Excerpt B than in Excerpt A.
- **DISAGREES:** Excerpt A contradicts or opposes Excerpt B in intent or recommendation.
- **INCOMPARABLE:** The two address unrelated domains, abstraction levels, or objectives, or if the model is uncertain or lacks confidence in its ability to determine a clear relationship.

To guide the model towards structured reasoning beyond a mere categorical judgement, we prompt the model to justify the label based on scope, overlap, technical differences or lack of comparable content. For relations other than **AGREES**, we also ask the model to specify all the differences between the two excerpts. For ease of interpretation and to reduce the risk of hallucination, we include explicit instructions to return only a valid Python dictionary containing the relation and explanation. The system message designates the LLM as a “security standards analyst” to provide context for the model’s task; this has been found to help better align the model’s response with the intention of the task [43, 38]. Furthermore, the prompt encourages abstention through the **INCOMPARABLE** label if the model is not at

least 95% confident in its decision. This is used to mitigate overconfident or incorrect labels, especially in cases where the true relationship between the controls may be ambiguous. The prompt template used for LLM-based comparisons can be found in Appendix A.1.

4.3.3 *Comparing iteratively*

We reconstruct the prompt for each control in Revision A, comparing it to the corresponding mapped controls in Revision B. The comparison for each control is conducted via a separate call to `chat.completions.create`, which ensures that this comparison is performed independently of the context and influence of prior comparisons. All the results are saved to a JSON file, with the individual dictionary-like output produced by the LLM converted to a Python dictionary object using `ast.literal_eval`. We repeat this iterative comparison for each control in Revision B, mapped to Revision A controls. We will use “forward mapping” to refer to comparisons from an older revision to a newer revision, and “backward mapping” to refer to comparisons from a newer revision to an older revision. This bidirectional comparison generates a complementary set of labels and explanations that allows for more robust examinations of the changes between revisions.

4.4 LLM-based summarisation

After performing the bidirectional comparison between revisions, we again use an LLM to generate a high-level summary of the semantic relationships observed across revisions. This summarisation step is designed to distill hundreds of granular control comparisons into a more concise, human-readable analysis that characterises the similarities and differences at a high level. This process allows an end user to better understand broader patterns and conceptual changes between the two standards.

We perform this LLM-based summarisation using the same Meta LLaMA 3.3 70B Instruct model used previously. As before, we use a temperature of zero to improve consistency in

the response. To construct the prompt, we aggregate the previously outputted comparison results, grouping each control and its corresponding explanations by its relationship label (AGREES, SUBSET, SUPERSET, DISAGREES or INCOMPARABLE). During this aggregation, we reclassify SUBSET-labelled controls, if they have a one-to-many mapping in the embeddings step. This reflects the idea that a single control may have been partitioned into more granular units in the other revision, rather than simply being less comprehensive. This distinction can help the model better identify restructuring between revisions, as controls are added, moved around and split up.

These label-grouped explanations—from both directions of the bidirectional comparisons—are then provided to the LLM, which is asked to identify similarities, differences, contradictions and broader thematic shifts in the control guidance. Specifically, we ask the model to address six questions:

1. What are the similarities between these two standards?
2. What are the differences between these two standards?
3. Are there any inconsistencies between these two standards? That is, would following the guidelines in one standard conflict with guidelines in the other standard?
4. What broad categories of guidance are generally missing from the `{std1_name}` standard? What types of guidance are generally missing from the `{std2_name}` standard? Please broadly characterize these missing controls. If I am a member of the security team at an organization, what differences should I be aware of?
5. Highlight what this might reveal about how the controls have evolved between `{std1_name}` and `{std2_name}`.
6. Summarize any notable patterns in the explanations.

We instruct the LLM to synthesise this information into a 3-5 paragraph summary, explicitly asking it to refrain from using any prior knowledge in order to reduce the likelihood of hallucinations. The output serves as a useful high-level summary of the broader conceptual

and structural shifts between revisions. The full prompt used for LLM-based summarisation can be found in Appendix A.2.

4.5 Evaluation

To evaluate the quality of the LLM-generated relationship classifications and explanations, we perform deductive coding using a predefined coding scheme consisting of six categories: Correct; Correct, but incorrect labelling; Omission; Addition; Incorrect; and Irrelevant. These categories are outlined in Table 5.2. This evaluation focuses on both the correctness of the LLM’s relationship label and the semantic fidelity of its justification. In particular, we assess whether the model accurately identifies the nature of the relationship between controls and whether the accompanying explanation captures all relevant elements of the control’s meaning without omitting important details or introducing fabricated content.

4.5.1 *Revision 5 and Revision 4 comparison*

For the comparison between NIST SP 800-53 Revision 4 and Revision 5, we leveraged the official change documentation released as a supplemental document by NIST alongside Revision 5 [32]. In particular, the *Mapping of Appendix J Privacy Controls (Rev. 4) to Rev. 5a* spreadsheet provides a definitive set of control-level correspondences and annotated change details, which allows for a direct verification of the LLM’s outputs for all the controls. This document serves as a ground truth against which we could validate the correctness of the LLM-generated mappings. Each LLM-generated relationship label and its associated explanation was reviewed against this ground truth and coded accordingly, with special attention paid to whether the model captured all relevant content, omitted key information or introduced hallucinated details. The results are presented in Table 5.3.

4.5.2 *Other revision comparisons*

For comparisons involving earlier revisions of SP 800-53, no official control-level mapping or detailed change log is available to define control-level mappings or describe the evolution of controls in a structured format. In the absence of such reference material, we employed a sampling-based evaluation strategy. A stratified random sample of controls was drawn across the LLM-produced labels, and the corresponding LLM outputs for those controls were manually reviewed. Coding, using the same six categories as previously described, is based on a close reading of the control texts and a judgement of the reasonableness of the LLM’s inferred relationships. The results of these evaluations are also presented in Table 5.3.

CHAPTER 5

RESULTS AND DISCUSSION

This chapter presents the results of our analysis and discusses key changes identified by applying our methodology to compare revisions of NIST SP 800-53. We begin by examining the structure and distribution of embeddings-based control mappings. We then evaluate the correctness of LLM-generated relationship labels and interpretability of LLM-generated explanations, and reflect on the broader implications of these findings for the viability of automated LLM-based approaches to regulatory comparison tasks.

5.1 Analysis of embeddings-based mappings

This section analyses the efficacy of the embeddings-based control mappings. Table 5.1 displays summary statistics from these mappings in both backward (mapping from Revision 5 to earlier revisions) and forward (mapping from earlier revisions to Revision 5) directions.

Table 5.1: Summary of control mapping metrics across revision comparisons.

Metric *	Backward mappings (from Rev. 5)					Forward mappings (to Rev. 5)				
	Rev. 4	Rev. 3	Rev. 2	Rev. 0	Avg	Rev. 4	Rev. 3	Rev. 2	Rev. 0	Avg
<i>Counts</i>										
Mappings made	298	298	298	298	298.00	265	198	171	163	199.25
1-to-1 mappings	254	201	170	159	196.00	239	188	156	148	182.75
1-to-1 matches **	232	182	148	142	176.00	231	182	152	141	176.50
1-to-many mappings	44	97	128	139	102.00	26	10	15	15	16.50
<i>Summary statistics of controls per mapping</i>										
Average count	2.55	3.77	10.03	6.67	5.76	4.16	1.29	1.44	1.50	2.10
Maximum count	25	25	170	162	95.50	296	15	15	15	85.25
Avg mean similarity	0.955	0.925	0.899	0.893	0.918	0.965	0.959	0.942	0.937	0.951
Avg max similarity	0.958	0.930	0.907	0.900	0.924	0.967	0.960	0.943	0.939	0.952

* For a control X in the source revision, its “mapped controls” refers the set of controls in the target revision that are most semantically similar to X (as measured by the cosine similarity between the embeddings). Then, the average number of mapped controls is the average of this number across all the controls in the source revision.

** The number of 1-to-1 matches refers to the number of 1-to-1 mappings, where the control identifier of the source revision matches the control identifier of the target revision. In order for a control in the source revision to have a 1-to-1 match, its control family and identifier must exist in the target revision.

5.1.1 *Backward mappings tend to be more diffuse*

The mapping, summarised in Table 5.1, reveals that controls introduced in Revision 5 tend to have more diffuse mappings when aligned to earlier revisions. That is, a single control from Revision 5 maps to several distinct controls in a previous revision, as opposed to having a one-to-one counterpart, indicating a diffusion of its content across multiple older controls. This is evident in the significantly higher average number of mapped controls per source control when mapping from Revision 5 to Revision 4 (2.55), when compared to mapping from Revision 5 to Revision 2 (10.03). These statistics suggest that, on average, the guidance provided in a single control in Revision 5 is covered by fewer than three controls from Revision 4, and just over ten controls from Revision 2. This suggests that newly introduced controls may include additional details that span multiple older controls, so it is more difficult to establish a clean one-to-one correspondence using the embeddings-based mapping.

This observation is especially prevalent when mapping Revision 5 controls to an older revision that does not contain any explicit equivalent to the control’s guidance. In such cases, the control may align only partially with a broad collection of thematically adjacent controls, the combination of which still does not capture the full intent of the newer control. In particular, modern controls related to privacy, supply chain risk management or continuous monitoring may only find approximate coverage scattered across disparate technical controls in older revisions. For instance, control PM-30, titled Supply Chain Risk Management Strategy, is a new base control introduced in Revision 5. It is mapped to PM-9 in both the Revision 3 and Revision 4 mappings, which discusses broader risk management strategy but does not explicitly mention supply chains. However, in revisions prior to Revision 3, PM-9 does not exist, and the entire Program Management control family is absent. As a result, PM-30’s guidance in Revision 5 lacks a direct or even approximate counterpart in these earlier revisions. It is mapped to a set of 15 thematically similar controls in Revision 2 and a different set of 15 controls in Revision 0. In these cases, the mapping serves more

as a note of thematic adjacency than as a reliable basis for backward equivalence.

On the other hand, controls that are consistently present across all revisions tend to exhibit much more stable and concentrated mappings. These controls often maintain similar functional intent over time despite changes in the level of detail provided by the guidance. As such, they are more likely to preserve semantic continuity across revisions, enabling clearer alignment between controls in different revisions when computing embedding similarities. This stability allows for more accurate mappings using the embedding-based approach. For example, SI-4, titled System Monitoring in Revision 5, appears in every revision of SP 800-53. The language and specificity of the control has evolved extensively from the original revision to Revision 5. In Revision 0, the control read:

SI-4 (Intrusion Detection Tools and Techniques): The organization employs tools and techniques to monitor events on the information system, detect attacks, and provide identification of unauthorized use of the system.

In Revision 5, the control is expanded to:

SI-4 (System Monitoring): a. Monitor the system to detect: 1. Attacks and indicators of potential attacks in accordance with the following monitoring objectives: [Assignment: organization-defined monitoring objectives]; and 2. Unauthorized local, network, and remote connections; b. Identify unauthorized use of the system through the following techniques and methods: [Assignment: organization-defined techniques and methods]; c. Invoke internal monitoring capabilities or deploy monitoring devices: 1. Strategically within the system to collect organization-determined essential information; and 2. At ad hoc locations within the system to track specific types of transactions of interest to the organization; d. Analyze detected events and anomalies; e. Adjust the level of system monitoring activity when there is a change in risk to organizational operations and assets, individuals, other organizations, or the Nation; f. Obtain legal opinion regarding system monitoring activities; and g. Provide [Assignment: organization-defined system monitoring information] to [Assignment: organization-defined personnel or roles] [Selection (one or more): as needed; [Assignment: organization-defined frequency]].

Despite the substantial changes in the presentation of the control and the amount of detail in the guidance, and even a change in the control title, our embeddings model consistently maps SI-4 in Revision 5 to SI-4 in all the revisions with a relatively high similarity score.

This suggests that our approach is capable of capturing conceptual continuity even when the control verbiage and length are vastly different. Embedding-based methods can robustly align semantically stable controls across revisions, but produce more diffuse mappings for controls that undergo significant changes in granularity and scope between revisions.

5.1.2 Mapping is largely asymmetric

Another observation from the embeddings-based mapping process is that mapping between revisions is asymmetric, reflecting the increasing scope and coverage of controls over time. That is, mapping from Revision 5 to earlier revisions produces different patterns than mapping from earlier revisions to Revision 5. Revision 5 introduces many controls that are broader in scope and incorporate security considerations across domains that were either absent or only implicitly addressed in earlier revisions. As a result, when a Revision 5 control is mapped to an older revision, its content is often dispersed across a wide set of thematically similar controls. On the contrary, Revision 5 still covers most of the controls in prior versions, usually extending the original control text with greater detail and supplemental guidance. As a result, the reverse mappings from earlier revisions to Revision 5 are more concentrated. Controls from earlier revisions often align with a single control in Revision 5, leading to a higher proportion of one-to-one mappings and a reduced average number of mapped controls.

This observation is especially prevalent for newly introduced controls. Take, for example, PT-1 in Revision 5, the Policy and Procedures control for the Privacy Authorisation control family. Prior to Revision 5, the Privacy Authorisation family did not exist altogether, so when mapping from Revision 5 to earlier revisions, PT-1 tends to get mapped to several generic policy and procedure-related controls from other control families, such as Personnel Security Policy and Procedures (PS-1) and Physical and Environmental Protection Policy and Procedures (PE-1). However, in the mappings from earlier revisions to Revision 5, no

control is mapped to PT-1 at all. This absence illustrates a conceptual gap between the revisions; earlier revisions may contain language that touches on privacy policies, but they lack the granularity of the new control. When a new control encapsulates concerns that were previously unaddressed or only implicitly distributed across the standard, forward mappings may result in a one-to-many relationship, while reverse mappings yield no relationship with the new control at all.

This trend can also be observed through the higher average number of mapped controls in the forward mapping in relation to the counterpart backward mapping to the same previous revision, with the exception of the mapping to Revision 4. The maximum number of mapped controls reveals a similar trend, with forward mappings generally mapping at most 15 other controls (again, with the exception of Revision 4) whereas backward mappings map to upwards of 25. This further indicates that backwards mappings from Revision 5 often span multiple tangentially-related controls in earlier revisions, while forward mappings from older revisions tend to concentrate on a few highly similar Revision 5 controls. The exception of Revision 4 is likely the result of the structural differences introduced in Revision 4 that were revoked in Revision 5.

However, the asymmetry is also present in the opposite direction, where a single Revision 5 control maps cleanly to an older control, but that same older control maps to multiple controls in Revision 5. For example, AU-2, titled Event Logging in Revision 5, is a focused and modular control that covers logging:

AU-2 (Event Logging): a. Identify the types of events that the system is capable of logging in support of the audit function: [Assignment: organization-defined event types that the system is capable of logging]; b. Coordinate the event logging function with other organizational entities requiring audit-related information to guide and inform the selection criteria for events to be logged; c. Specify the following event types for logging within the system: [Assignment: organization-defined event types along with the frequency of (or situation requiring) logging for each identified event type]; d. Provide a rationale for why the event types selected for logging are deemed to be adequate to support after-the-fact investigations of incidents; and e. Review and update the event types selected for logging [Assignment: organization-defined frequency].

When mapping from Revision 5 to earlier revisions, AU-2 maps exclusively to AU-2, formerly titled Audit Events / Auditable Events, in Revisions 0 through 4:

AU-2 (Auditable Events): The information system generates audit records for the following events: [Assignment: organization-defined auditable events].

AU-2 is much more detailed and comprehensive in Revision 5 about logging, whereas AU-2 in the previous revisions is very broad and rather vague. As such, in the forward mapping, this broader AU-2 control is aligned with multiple more granular audit-related controls in Revision 5, including AU-12 (Audit Record Generation), AU-2 (Event Logging) and AU-3 (Content of Audit Records). This one-to-many mapping in the forward direction arises because Revision 5 decomposes the functionality of the older AU-2 into several distinct components, each with more specific guidance and assignments. In the backward direction, however, the newer AU-2 control only maps to the broader AU-2 control in older revisions since that is the only remotely relevant control in the earlier revision. This illustrates a different asymmetry from what is observed with newly introduced controls such as PT-1.

Overall, this asymmetry of control mappings reflects overarching shifts in conceptual framing and scope between revisions; newer controls tend to be more granular or modular, whereas older controls are broader in scope. This effectively creates two causes of asymmetry, one from the expansion of scope and the other from increased modularity.

5.2 Evaluation of LLM comparisons

As introduced in Section 4.5, we use deductive coding to evaluate the quality of the LLM’s label and response. Based on the ground truth and close readings of the control texts, we manually coded the LLM-generated classification and explanation into one of six categories: Correct; Correct, but incorrect labelling; Omission; Addition; Incorrect; and Irrelevant. The individual criterion for each coding category is further described in Table 5.2, and the coding results are presented in Table 5.3. This section discusses the model’s performance using these

categories and examines the strengths and weaknesses in how the model interprets changes across different revisions.

Table 5.2: Deductive coding categories used to evaluate LLM-generated comparisons.

Code	Description
Correct	The LLM output matches ground truth with no substantive errors.
Correct, but incorrect labelling	The explanation correctly identifies the nature of the change, but the LLM assigns the wrong relationship label.*
Omission	The LLM omits important details or aspects of the change that appear in the ground truth.
Addition	The LLM hallucinates or introduces information not present in the ground truth.
Incorrect	The LLM explanation misrepresents the meaning or direction of change, including logical contradictions.
Irrelevant	The response is off-topic or too vague to meaningfully engage with the control content.

* For example, the LLM may identify that Excerpt A contains more specific guidance than that presented in Excerpt B, but incorrectly label Excerpt A as a **SUBSET** of Excerpt B (when it should be a **SUPERSET** under our definition).

Table 5.3: Coding counts for LLM comparison of Revision 5 against previous revisions.

Code	Rev. 4	Rev. 3	Rev. 2	Rev. 0
Correct	250	14	9	11
Correct, but incorrect labelling	20	1	8	4
Addition	0	0	0	0
Omission	20	3	2	3
Incorrect	8	2	1	2
Irrelevant	0	0	0	0
Total coded	298	20	20	20

5.2.1 *The LLM correctly identifies most changes, even subtle changes*

Our iterative comparison approach revealed that the LLM, on the whole, was able to detect both major semantic and fundamental changes, as well as the introduction of minor details

between revisions. Across all revisions, a majority of the controls are classified correctly, as seen in the coding results provided in Table 5.3. In most cases, the LLM correctly flags when one revision omits a detail that is explicitly present in another revision. This suggests a degree of contextual awareness in how the LLM parses meaning across comparable texts; rather than merely checking for lexical overlap, the LLM consistently demonstrates an ability to assess the functional overlap and gaps between two pieces of advice. Further, of the LLM comparisons that were manually verified, none of the comparisons blatantly contradicts the ground truth, with most of the incorrect coding labels stemming from an overly literal interpretation of the text rather than a substantive misunderstanding of the underlying control meaning.

Impressively, the LLM captures and explains even minor detail changes in the control text, such as the AC-8 control on System Use Notification. AC-8 from Revision 0 reads:

The information system notifies the user, upon successful logon, of the date and time of the last logon, and the number of unsuccessful logon attempts since the last successful logon.

while AC-8 from Revision 5 reads:

Notify the user, upon successful logon to the system, of the date and time of the last logon.

The LLM correctly identifies the additional guidance in Revision 0, noting that “Rev. 0 excerpt provides more detailed guidance by including the number of unsuccessful logon attempts since the last successful logon” and subsequently classifies the guidance in Revision 5 as a subset of the guidance in Revision 0. This illustrates the model’s ability to catch changes in security details, even when the overall structure and tone of the guidance remain consistent. In addition, the LLM, for the most part, does not overreact to surface-level differences in phrasing. Instead, it is capable of identifying when two versions of a control offer substantively equivalent guidance, even if one offers additional context, as is the case with

AC-21 in the comparison between Revisions 4 and 5. The control texts are essentially the same across the two revisions, but the supplemental guidance for AC-21 in Revision 4 reads:

[...] Information may be defined by content, type, security category, or special access program/compartment.

For Revision 5, the discussion for AC-21 expands to:

[...] Information may be defined by content, type, security category, or special access program or compartment. Access restrictions may include non-disclosure agreements (NDA). Information flow techniques and security attributes may be used to provide automated assistance to users making sharing and collaboration decisions.

In this example, the LLM correctly identifies their similarity and agreement, even pointing out the subtle difference in the additional examples provided in the Revision 5 discussion:

The main differences are minor and include additional related controls listed in Rev. 5 and a slightly expanded discussion of examples of restricted information, but these do not change the overall intent or recommendation of the control.

This illustrates the model's ability to differentiate between the overarching foundations of a control and the examples that accompany it. The efficacy with which the model identifies changes between revisions raises the possibility that it may have been potentially trained on, or influenced by, the publicly available change documentation provided by NIST. While testing the model for memorisation may be worthwhile, this explanation is still insufficient to account for the full range of behaviours observed. In particular, in some cases, the model correctly provides additional details about a change between revisions beyond what is described by the change documentation. For example, in the comparison of AC-7 in Revision 5 to AC-7 in Revision 4, the change details include:

Parameter includes additional selection options when the number of allowed consecutive invalid logon attempts threshold is exceeded. Discussion amplifies the control text with examples of additional actions to help prevent brute force attacks

However, the LLM identifies additional differences between the controls beyond these:

The Rev. 5 excerpt provides more comprehensive guidance than the Rev. 4 excerpt, including additional actions such as prompting the user to answer a secret question, invoking a lockdown mode, allowing users to only logon from specified IP addresses, requiring a CAPTCHA, or applying user profiles. The Rev. 5 excerpt also mentions considering a combination of other actions to help prevent brute force attacks and prompting users to respond to a secret question before the number of allowed unsuccessful logon attempts is exceeded, which are not present in the Rev. 4 excerpt.

Finally, in cases where the source control is truly incomparable with the set of controls it is mapped to, typically for newly introduced controls in Revision 5, the model accurately labels the pairing as **INCOMPARABLE**. These instances demonstrate that the model is capable of recognising when no meaningful conceptual overlap exists, rather than forcing a match based on linguistic or structural similarities. These results show promise for the use of large language models in the comparison of security advice.

5.2.2 The LLM is more sensitive to smaller changes and more susceptible to omission for substantial changes

For smaller, localised changes, the LLM performs comparison remarkably well. These smaller changes typically involve the addition or removal of a clause or a slight shift in phrasing. The LLM generally detects these edits with precision, correctly classifying the control as a subset or superset of the other, and providing explanations that accurately reflect the specific semantic shift. For example, when comparing PL-4 between Revisions 4 and 5, the LLM identifies “minor differences in wording, such as ‘documented acknowledgment’ in Rev. 5 versus ‘signed acknowledgment’ in Rev. 4, and the inclusion of ‘electronic agreement check boxes or radio buttons’ in Rev. 5.” This fine-grained awareness suggests that the current model is capable of identifying incremental updates that may otherwise go unnoticed in broader comparisons.

In contrast, for more substantial changes, such as major reorganisations or the introduction of entirely new implementation concepts, the LLM is more likely to omit some change

details when making the comparison. Rather than indicating a lack of understanding of the control, this pattern likely stems from the nature of the prompt structure and the constraints of the model’s focus. When faced with broader conceptual shifts, the LLM may struggle to compress all relevant information into a concise explanation, leading it to foreground high-level themes while omitting more granular content. In these cases, its responses generally capture the general direction of the revision, but fail to fully enumerate the specific change details. This is particularly the case for the Policy and Procedures controls; for each control family, Revision 5 introduces several new policies and procedures that are not presented in the Revision 4 document. For instance, the change details documentation for IR-1 identifies several change details:

Title changed from ‘Incident Response Policy and Procedures’. Requires the selection (one or more) of organization-level; mission/business process-level; system-level incident response policies. Adds text requiring consistency with applicable laws, executive orders, directives, regulations, policies, standards, and guidelines. Requires the designation of a specific official to manage the development, documentation, and dissemination of the incident response policy and procedures. New parameters require update to policy and procedures as a result of specified events in addition to specified frequency. Discussion amplifies the need for policy and procedures for risk management, and to help provide security and privacy assurance.

These details of these changes are non-exhaustively captured in the LLM’s response:

The Rev. 5 excerpt provides more detailed and comprehensive guidance on incident response policy and procedures, including the requirement for consistency with applicable laws and regulations, and the consideration of events that may precipitate updates to the policy and procedures, whereas the Rev. 4 excerpt provides more general guidance on the development, documentation, and dissemination of incident response policy and procedures.

While the model captures the high-level changes, it fails to enumerate all the details of the changes. This contrast highlights both the strengths and limitations of LLM-based comparison: the model excels at detecting fine-grained, syntactic edits but is less precise when trying to communicate more substantial changes. This also exemplifies the value of performing control-level mappings and comparisons as opposed to broader document-level comparisons;

by isolating individual controls, we can more effectively identify localised semantic shifts and scope changes that may be otherwise obscured in document-level analyses.

5.2.3 *The LLM can sometimes interpret the text too literally*

In a similar vein to the sensitivity to smaller changes, the LLM interprets control text too literally on several occasions, particularly when revisions introduce supplementary context or illustrative examples. While the model demonstrates overall comprehension, it sometimes misinterprets clarifying language in newer revisions as substantively narrowing the scope of a control, even when the intended function remains equivalent. The following example illustrates this tendency for the PE-15 (Water Damage Protection) control from Revision 5 when compared against PE-15 from Revision 0. The Revision 0 control reads:

PE-15 (Water Damage Protection): The organization protects the information system from water damage resulting from broken plumbing lines or other sources of water leakage by ensuring that master shutoff valves are accessible, working properly, and known to key personnel.

Meanwhile, the Revision 5 control reads:

PE-15 (Water Damage Protection): Protect the system from damage resulting from water leakage by providing master shutoff or isolation valves that are accessible, working properly, and known to key personnel. Discussion: The provision of water damage protection primarily applies to organizational facilities that contain concentrations of system resources, including data centers, server rooms, and mainframe computer rooms. Isolation valves can be employed in addition to or in lieu of master shutoff valves to shut off water supplies in specific areas of concern without affecting entire organizations.

In this scenario, it is evident that the guidance provided for this control in Revision 0 is effectively covered entirely by the guidance provided in Revision 5; the set of guidance in Revision 5 should be a superset of the advice in Revision 0. However, the LLM labels the guidance in Revision 5 as a subset of the guidance in Revision 0, with the explanation that:

The Rev. 0 excerpt is broader in the sense that it mentions protection from water damage resulting from broken plumbing lines or other sources of water leakage, while the Rev. 5 excerpt primarily applies to organizational facilities containing concentrations of system resources.

This response reflects an overly literal reading of the text in Revision 5. While the discussion does list organisational facilities such as data centres and server rooms, the purpose of including these facilities does not functionally narrow the control’s application. The misunderstanding likely stems from the LLM interpreting the discussion as a limiting clause rather than an explanatory one, potentially overly fixating on specific phrases that appear to narrow the scope. As a result, the model incorrectly infers that the newer control is less comprehensive, when in reality it is more flexible and technically detailed. This behaviour appears especially prevalent in cases where revisions introduce additional explanatory guidance, including examples and common implementations. Rather than recognising these as non-exclusive elaborations, the model may misinterpret these as constraints, thus misclassifying the relationship of the controls.

Interestingly, this appears to contradict the pattern observed in cases like AC-21 discussed above, where the model correctly identified that additional examples did not alter the overall intent or recommendation of the control. This may be a result of how the terminology and phrasing of the control influences how the model interprets functional scope. The PE-15 excerpt uses the phrase “primarily applies to,” which may be interpreted as limiting the applicability of the control, in contrast to the more permissive “may be used to” phrasing used in the AC-21 discussion. In these cases, the model may mistake clarifying language for a narrowing of scope. Since a majority of the LLM responses coded as “Incorrect” in Table 5.3 stems from this type of misinterpretation, future work could experiment with more structured prompts that more explicitly instruct the LLM to focus on the functional intent rather than other contextual elements.

5.3 The evolution of NIST SP 800-53

This section examines the major changes that the model captures in the evolution of the NIST SP 800-53 standard. Table 5.4 summarises the distribution of relationship labels for both backward and forward mappings. The prevalence and variation of these labels across pairs of revision comparisons provide insight into how the structure and semantics of the controls have shifted over time. The changes leading up to Revision 5 can broadly be categorised into two overarching trends: an expansion in conceptual scope and an increase in modularity and implementation detail.

Table 5.4: LLM labels for control mappings.

Metric*	Backward mappings (from Rev. 5)				Forward mappings (to Rev. 5)			
	Rev. 4	Rev. 3	Rev. 2	Rev. 0	Rev. 4	Rev. 3	Rev. 2	Rev. 0
AGREES	125	24	7	8	129	17	5	2
SUBSET	84	127	181	162	110	168	164	157
SUPERSET	84	121	83	96	26	13	2	4
DISAGREES	0	0	0	0	0	0	0	0
INCOMPARABLE	5	26	27	32	0	0	0	0
Total	298	298	298	298	265	198	171	163

5.3.1 An expansion in conceptual scope

Over successive revisions, NIST SP 800-53 has expanded its conceptual boundaries to incorporate broader organisational, privacy and risk management concerns that were largely absent in earlier versions. Controls introduced in Revision 5 largely address new domains, such as supply chain risk management and privacy governance, which were not formally represented in previous versions. As shown in Table 5.4, this conceptual broadening is reflected in the relative scarcity of **AGREES** labels as we compare Revision 5 with earlier versions. For instance, while 125 controls are labelled as **AGREES** in the backward comparison from Revision 5 to Revision 4, only seven and eight controls are considered functionally equivalent in comparison with Revision 2 and Revision 0, respectively. This sharp decline suggests

that Revision 5 substantially reframes and reorganises many existing controls to cover a broader security landscape. The large proportion of **SUBSET** labels in the backward mapping to older revisions further supports this observation, indicating that backward mappings from Revision 5 are subject to many one-to-many mappings characteristic of newly introduced controls, while the sizeable amount of **SUPERSET** labels also indicates that Revision 5 controls frequently subsume and extend the content of older controls. Furthermore, we observe more **INCOMPARABLE** labels in comparisons of Revision 5 against older versions; since **INCOMPARABLE** labels suggest that the content of the mapped controls addresses unrelated topics or domains, this is also an indication of the broader scope of Revision 5 and the increasing broadening of scope over time. The LLM-based summarisation results, presented in Appendix B, also capture these changes.

These observations align closely with two of the seven “most significant changes” to Revision 5, identified in the NIST Cybersecurity blog [53]:

- **Integrating supply chain risk management:** Revision 5 establishes a new Supply Chain Risk Management (SCRM) control family and integrates supply chain risk management aspects throughout the other control families to help protect system components, products, and services that are part of critical systems and infrastructures. The SCRM controls help ensure that security and privacy requirements, threats, and other concerns are addressed throughout the system development life cycle and the national and international supply chains.
- **Adding new state-of-the-practice controls:** As cyber threats evolve rapidly, new safeguards and countermeasures are needed to protect the critical and high value assets of organizations including individual’s privacy and personally identifiable information. The new controls in Revision 5 are based on the latest threat intelligence and cyber-attack data (e.g., controls to support cyber resiliency, secure systems design, security and privacy governance, and accountability).

5.3.2 *An increase in modularity*

Alongside the conceptual expansion, NIST SP 800-53 has evolved towards a more modular and granular structure, with newer revisions featuring more specific and outcome-based guidance. This trend is particularly evident in the forward mappings from older revisions

to Revision 5. As shown in Table 5.4, the SUBSET label dominates forward mappings from revisions prior to Revision 3, meaning that older controls are often subsets of their Revision 5 counterparts—that is, they tend to be less detailed or comprehensive in comparison. The prevalence of older controls being subsets of Revision 5 controls suggests two things:

1. Earlier controls persist in Revision 5, but now appear more narrowly scoped within a richer ecosystem of related guidance.
2. Revision 5 introduces finer distinctions between conceptual domains, often containing additional guidance for what may have previously collapsed into a single requirement.

Rather than prescribing monolithic controls, Revision 5 decomposes guidance into more well-defined, self-contained guidance with parameters tuneable for individual organisational needs. This makes it easier for implementers to apply controls selectively and systematically. These changes align closely with two other of the seven changes identified in the NIST Cybersecurity blog [53]:

- **Making controls outcome-based:** Revision 5 accomplishes this by removing the entity responsible for satisfying the control (i.e., information system, organization) from the control statement—thus focusing on the protection outcome to be achieved by the application of the control. Note that for historical continuity, Appendix C, Control Summaries now includes an “implemented by [system/organization]” column.
- **Improving descriptions of content relationships:** Revision 5 clarifies the relationship between requirements and controls as well as the relationship between security and privacy controls. These relationships are important to understand whether you are selecting and implementing controls at the enterprise level or as part of a life cycle-based systems engineering process.

Revision 5 also contains more comprehensive discussion and control enhancements compared to earlier revisions. Though our comparisons do not investigate changes to the control enhancements, these additions further provide elaboration on control intent and offer practical implementation considerations. Importantly, the increase in modularity across successive revisions does not introduce contradictory information or disagreements, so newer revisions tend to extend reorganise existing guidance rather than fundamentally altering its meaning.

5.3.3 Summary of changes

Together, the label distributions across forward and backward mappings reinforce the notion that NIST SP 800-53 has matured from a set of general-purpose controls into a complex, modular framework designed to address the evolving threat landscape. This transformation reflects the increased technical sophistication of the compliance landscape.

Notably, three of the seven “most significant changes” to Revision 5 outlined in the NIST Cybersecurity blog are not captured by the LLM comparison. These changes include [53]:

- **Consolidating the control catalog:** Information security and privacy controls are now integrated into a seamless, consolidated control catalog for systems and organizations. The privacy controls in Appendix J of Revision 4 have been incorporated into a new privacy family and the existing Program Management family. [...]
- **Separating the control selection process from the controls:** [...]
- **Transferring control baselines and tailoring guidance to a separate publication:** Control baselines have been moved to the new NIST SP 800-53B, Control Baselines for Information Systems and Organizations. [...]

Since the LLM-based comparisons occur at the control level, they inherently lack visibility into changes made to the higher-level structure of the standard. As a result, these higher-order modifications—such as consolidation, process decoupling and baseline removal—remain outside the scope of what this method can capture. While LLMs are effective at tracking semantic drift and refinement within individual controls, this approach makes them less sensitive to overarching structural changes across standards. We discuss additional limitations of our approach in Section 5.4 below.

5.4 Limitations

While our methodology demonstrates the utility of sentence embeddings and large language models for comparing cybersecurity standards, several limitations should be acknowledged.

First, the accuracy of our approach depends heavily on the quality of the initial control mappings. Although the embedding-based strategy enables flexible one-to-many mappings,

it may introduce noise, particularly when controls are diffuse or when a control has no true semantic counterpart in the other revision. In such cases, even a highly capable LLM may struggle to draw meaningful comparisons, resulting in incorrect or overly generic relationship labels. We observe this in our comparison results when a control that is mapped to a set of many other controls is labelled as a subset of that set; while true, this label is not particularly helpful for determining how the control aligns conceptually with the other revisions.

The accuracy of the comparisons is also limited by the accuracy of the model. We have seen that the LLM occasionally exhibits sensitivity to surface-level textual differences, sometimes misinterpreting elaborative examples or clarifying language as changes in scope or intent. This tendency toward literalism can lead to inaccurate comparisons, particularly in cases involving substantial explanatory additions in newer revisions. Similarly, the model occasionally omits relevant change details when faced with larger conceptual shifts, indicating limitations in reasoning. LLMs are also prone to hallucination, which introduces an additional aspect of unreliability. This poses a risk in high-stakes contexts such as standards compliance, where correctness is critical.

The evaluation of model outputs remains partially subjective, particularly for comparisons where no official ground truth exists. While deductive coding provides a structured approach to model evaluation, the classification of correctness, omissions, hallucinations or mislabelling is still subjective and potentially inconsistent. These limitations suggest important avenues for refinement, including improved alignment techniques and enhanced prompting strategies.

CHAPTER 6

CONCLUSION

This research presents and validates a methodology that combines sentence embeddings and large language models to compare and contrast revisions of NIST SP 800-53. Our results demonstrate that LLMs are capable of accurately classifying and explaining relationships between controls. The methodology reveals two primary trends in the evolution of NIST SP 800-53: a shift toward modularity and increased specificity in newer revisions, and the coverage of new domains such as privacy and supply chain risk management that lack historical equivalents. Despite some limitations, particularly in handling complex reorganisations or LLM errors, our approach achieves a high degree of semantic fidelity. More broadly, this research underscores the utility of LLMs in structured document comparison tasks and opens the door to generalising this framework for cross-standard comparison. As regulatory landscapes become more complex and interdependent, scalable tools for interpreting and aligning security standards will play an important role in security, compliance and risk management.

6.1 Future work

This work provides a foundation for using LLMs and sentence embeddings to assess semantic changes in cybersecurity standards over time. A number of avenues for future research and methodological refinement remain. While our analysis focuses on comparing across the revision history of NIST SP 800-53, our general methodology can be extended to comparisons between entirely different cybersecurity standards (e.g., FedRAMP, ISO/IEC 27001, SOC 2). Different standards have vastly different control families and document structures, but since we do not assume anything about the cross-revision similarities of NIST SP 800-53, the embeddings-based mapping and the LLM-based comparison and summarisation can still be applicable to cross-standard comparisons. However, as observed in our embeddings

for NIST controls, we have seen that controls that do not cleanly map to a single control in the other revision tend to have very diffuse mappings, making it difficult to ascertain specific relationships outside of being a subset of a large group of controls. This can dilute the semantic clarity needed to effectively compare controls across standards. One possible solution we’ve explored to address this challenge is by introducing an additional layer of abstraction. This involves mapping all the controls in both standards to an external security framework, such as the domains identified by the Secure Control Framework [7], to pre-cluster controls. Then, LLM-based comparisons can be applied within and across these clusters. This approach can support more scalable analyses of the interoperability of distinct sets of standards, which can be valuable for identifying gaps and contradictions that adhering to one standard over another can result in.

Prompting also remains a central challenge in LLM-based analysis. Future work could explore alternative prompt formulations, including chain-of-thought prompting [64], few-shot learning [21] or iterative prompting [74] strategies, to improve the reliability and interpretability of model outputs. Further, while our current methodology distinguishes the mapping and comparison steps, further work can experiment with the direct use of LLMs to generate mappings, either as a first step or as a refinement mechanism for diffuse mappings following the initial embedding-based matches.

This research used the **bge-base-en** model for the embeddings-based mappings. It would be beneficial to experiment with different sentence encoders to perform this mapping, as alternative models, especially those tuned on technical or legal documents, may offer improved domain-specific performance. We have also experimented with using LLMs to reconcile mismapped controls as an intermediate validation step before comparison to ensure accurate mappings and sensical comparisons. Additionally, the use of LLM-based embeddings [33, 58] may also allow us to integrate the mapping and comparison steps into a single workflow.

A broader goal of this project is the development of an end-to-end pipeline for standards

comparison by using semantic similarity, LLM-based reasoning and structured summarisation. By automating the identification of control additions, removals, restructurings and thematic shifts, such a system could help security professionals more efficiently assess the evolution of regulatory guidance and how those changes may influence organisational security implementations. This comparative analysis can enable the identification of areas where one framework's unique controls can complement another, offering a more comprehensive perspective for organisations looking to adopt or refine their security strategies.

REFERENCES

- [1] 2025. *2025 Data Breach Investigations Report*. <https://www.verizon.com/business/en-sg/resources/reports/dbir/>
- [2] 2025. *2.3 Federal Information Security Modernization Act (2002)*. <https://www.cio.gov/handbook/it-laws/fisma/>
- [3] 2009. About NIST. (2009). <https://www.nist.gov/about-nist> Last Modified: 2022-01-11T14:13-05:00.
- [4] 2025. *BAAI/bge-base-en* · *Hugging Face*. <https://huggingface.co/BAAI/bge-base-en>
- [5] 2025. *Compare models - OpenAI API*. <https://platform.openai.com>
- [6] 2025. *Cyber Security Tribe's Annual Report*. <https://www.cybersecuritytribe.com/annual-report>
- [7] 2025. *Cybersecurity & Data Privacy by Design (C|P) Principles*. <https://securecontrolsframework.com/domains-principles/>
- [8] 2022. *ISO/IEC 27002:2022*. <https://www.iso.org/standard/75652.html>
- [9] 2019. NIST History. (2019). <https://www.nist.gov/history> Last Modified: 2019-11-15T12:07-05:00.
- [10] 2025. NIST Timeline. (2025). <https://www.nist.gov/timeline>
- [11] Adebola Folorunso, Ifeoluwa Wada, Bunmi Samuel, and Viqaruddin Mohammed. 2024. Security compliance and its implication for cybersecurity. 24, 1 (2024), 2105–2121. doi:10.30574/wjarr.2024.24.1.3170
- [12] Hussain Al-Aqrabi, Lu Liu, Jie Xu, Richard Hill, Nick Antonopoulos, and Yongzhao Zhan. 2012. Investigation of IT Security and Compliance Challenges in Security-as-a-Service for Cloud Computing. In *2012 IEEE 15th International Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing Workshops* (2012-04). 124–129. doi:10.1109/ISORCW.2012.31
- [13] Meysam Alizadeh, Maël Kubli, Zeynab Samei, Shirin Dehghani, Mohammadmasiha Zahedivafa, Juan Diego Bermeo, Maria Korobeynikova, and Fabrizio Gilardi. 2024. Open-Source LLMs for Text Annotation: A Practical Guide for Model Setting and Fine-Tuning. doi:10.48550/arXiv.2307.02179 arXiv:2307.02179 [cs]
- [14] Felipe Almeida and Geraldo Xexéo. 2023. Word Embeddings: A Survey. doi:10.48550/arXiv.1901.09069 arXiv:1901.09069 [cs]

- [15] Benjamin Ampel, Chi-Heng Yang, James Hu, and Hsinchun Chen. 2025. Large Language Models for Conducting Advanced Text Analytics Information Systems Research. 16, 1 (2025), 2:1–2:27. doi:10.1145/3682069
- [16] Sourav Banerjee, Ayushi Agarwal, and Saloni Singla. 2024. LLMs Will Always Hallucinate, and We Need to Live With This. doi:10.48550/arXiv.2409.05746 arXiv:2409.05746 [stat] version: 1.
- [17] Stefano Bennati and Engin Bozdog. 2024. Automating Technical Privacy Reviews Using {LLMs}. (2024). <https://www.usenix.org/conference/pepr24/presentation/bennati>
- [18] Surayahani Hasnul Bhaharin, Umi Asma’ Mokhtar, Rossilawati Sulaiman, and Maryati Mohd Yusof. 2019. Issues and Trends in Information Security Policy Compliance. In *2019 6th International Conference on Research and Innovation in Information Systems (ICRIIS)* (2019-12). 1–6. doi:10.1109/ICRIIS48246.2019.9073645 ISSN: 2324-8157.
- [19] Paheli Bhattacharya, Kripabandhu Ghosh, Arindam Pal, and Saptarshi Ghosh. 2022. Legal case document similarity: You need both network and text. 59, 6 (2022), 103069. doi:10.1016/j.ipm.2022.103069
- [20] Mukund Bhole, Wolfgang Kastner, and Thilo Sauter. 2024. From Manual to Semi-Automated Safety and Security Requirements Engineering: Ensuring Compliance in Industry 4.0. In *IECON 2024 - 50th Annual Conference of the IEEE Industrial Electronics Society* (2024-11). 1–8. doi:10.1109/IECON55916.2024.10905636
- [21] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. doi:10.48550/arXiv.2005.14165 arXiv:2005.14165 [cs]
- [22] Haihua Chen, Lei Wu, Jiangping Chen, Wei Lu, and Junhua Ding. 2022. A comparative study of automated legal text classification using random forests and deep learning. 59, 2 (2022), 102798. doi:10.1016/j.ipm.2021.102798
- [23] Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. BGE M3-Embedding: Multi-Lingual, Multi-Functionality, Multi-Granularity Text Embeddings Through Self-Knowledge Distillation. doi:10.48550/arXiv.2402.03216 arXiv:2402.03216 [cs]
- [24] Shouyuan Chen, Sherman Wong, Liangjian Chen, and Yuandong Tian. 2023. Extending Context Window of Large Language Models via Positional Interpolation. doi:10.48550/arXiv.2306.15595 arXiv:2306.15595 [cs]

- [25] Yan Chen. 2022. Information security management: compliance challenges and new directions. 24, 4 (2022), 243–249. doi:10.1080/15228053.2022.2148979 Publisher: Routledge eprint: <https://doi.org/10.1080/15228053.2022.2148979>.
- [26] Yufan Chen, Arjun Arunasalam, and Z. Berkay Celik. 2023. Can Large Language Models Provide Security & Privacy Advice? Measuring the Ability of LLMs to Refute Misconceptions. In *Proceedings of the 39th Annual Computer Security Applications Conference* (New York, NY, USA, 2023-12-04) (*ACSAC '23*). Association for Computing Machinery, 366–378. doi:10.1145/3627106.3627196
- [27] Trieu C. Chieu, Manas Singh, Chunqiang Tang, Mahesh Viswanathan, and Ashu Gupta. 2012. Automation System for Validation of Configuration and Security Compliance in Managed Cloud Services. In *2012 IEEE Ninth International Conference on e-Business Engineering* (2012-09). 285–291. doi:10.1109/ICEBE.2012.53
- [28] Information Technology Laboratory Computer Security Division. 2016. *FISMA Background - NIST Risk Management Framework | CSRC | CSRC*. <https://csrc.nist.gov/projects/risk-management/fisma-background>
- [29] Yiran Ding, Li Lyna Zhang, Chengruidong Zhang, Yuanyuan Xu, Ning Shang, Jiahang Xu, Fan Yang, and Mao Yang. 2024. LongRoPE: Extending LLM Context Window Beyond 2 Million Tokens. doi:10.48550/arXiv.2402.13753 arXiv:2402.13753 [cs]
- [30] Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. 2024. Detecting hallucinations in large language models using semantic entropy. 630, 8017 (2024), 625–630. doi:10.1038/s41586-024-07421-0 Publisher: Nature Publishing Group.
- [31] Robert Filepp, Constantin Adam, Milton Hernandez, Maja Vukovic, Nikos Anerousis, and Guan Qun Zhang. 2018. Continuous Compliance: Experiences, Challenges, and Opportunities. In *2018 IEEE World Congress on Services (SERVICES)* (2018-07). 31–32. doi:10.1109/SERVICES.2018.00029 ISSN: 2378-3818.
- [32] Joint Task Force. 2020. Security and Privacy Controls for Information Systems and Organizations. NIST Special Publication (SP) 800-53 Rev. 5 (2020). doi:10.6028/NIST.SP.800-53r5
- [33] Matthew Freestone and Shubhra Kanti Karmaker Santu. 2024. Word Embeddings Revisited: Do LLMs Offer Something New? doi:10.48550/arXiv.2402.11094 arXiv:2402.11094 [cs] version: 2.
- [34] Stefano Gurciullo and Slava Mikhaylov. 2017. Detecting Policy Preferences and Dynamics in the UN General Debate with Neural Word Embeddings. doi:10.48550/arXiv.1707.03490 arXiv:1707.03490 [cs]
- [35] Peyman Hosseini, Ignacio Castro, Iacopo Ghinassi, and Matthew Purver. 2024. Efficient Solutions For An Intriguing Failure of LLMs: Long Context Window Does Not Mean LLMs Can Analyze Long Sequences Flawlessly. doi:10.48550/arXiv.2408.01866 arXiv:2408.01866 [cs] version: 3.

- [36] Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2024. A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions. doi:10.48550/arXiv.2311.05232 arXiv:2311.05232
- [37] Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2025. A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions. 43, 2 (2025), 1–55. doi:10.1145/3703155 arXiv:2311.05232 [cs]
- [38] Minbyul Jeong, Jungho Cho, Minsoo Khang, Dawoon Jung, and Teakgyu Hong. 2025. System Message Generation for User Preferences using Open-Source Models. doi:10.48550/arXiv.2502.11330 arXiv:2502.11330 [cs]
- [39] Hongye Jin, Xiaotian Han, Jingfeng Yang, Zhimeng Jiang, Zirui Liu, Chia-Yuan Chang, Huiyuan Chen, and Xia Hu. 2024. LLM Maybe LongLM: Self-Extend LLM Context Window Without Tuning. doi:10.48550/arXiv.2401.01325 arXiv:2401.01325 [cs]
- [40] Joint Task Force Interagency Working Group. 2020. Security and Privacy Controls for Information Systems and Organizations. (2020). doi:10.6028/NIST.SP.800-53r5 Edition: Revision 5.
- [41] Arshia Kermani, Veronica Perez-Rosas, and Vangelis Metsis. 2025. A Systematic Evaluation of LLM Strategies for Mental Health Text Analysis: Fine-tuning vs. Prompt Engineering vs. RAG. doi:10.48550/arXiv.2503.24307 arXiv:2503.24307 [cs]
- [42] Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent Retrieval for Weakly Supervised Open Domain Question Answering. doi:10.48550/arXiv.1906.00300 arXiv:1906.00300 [cs]
- [43] Seongyun Lee, Sue Hyun Park, Seungone Kim, and Minjoon Seo. 2024. Aligning to Thousands of Preferences via System Message Generalization. doi:10.48550/arXiv.2405.17977 arXiv:2405.17977 [cs] version: 1.
- [44] Bohan Li, Hao Zhou, Junxian He, Mingxuan Wang, Yiming Yang, and Lei Li. 2020. On the Sentence Embeddings from Pre-trained Language Models. doi:10.48550/arXiv.2011.05864 arXiv:2011.05864 [cs]
- [45] Tianle Li, Ge Zhang, Quy Duc Do, Xiang Yue, and Wenhui Chen. 2024. Long-context LLMs Struggle with Long In-context Learning. doi:10.48550/arXiv.2404.02060 arXiv:2404.02060 [cs] version: 3.
- [46] João Alberto de Oliveira Lima. 2024. Unlocking Legal Knowledge with Multi-Layered Embedding-Based Retrieval. doi:10.48550/arXiv.2411.07739 arXiv:2411.07739 [cs] version: 1.
- [47] Daniel Machlab and Rick Battle. 2024. LLM In-Context Recall is Prompt Dependent. doi:10.48550/arXiv.2404.08865 arXiv:2404.08865 [cs] version: 1.

- [48] Nishanth Madhusudhan, Sathwik Tejaswi Madhusudhan, Vikas Yadav, and Masoud Hashemi. 2024. Do LLMs Know When to NOT Answer? Investigating Abstention Abilities of Large Language Models. doi:10.48550/arXiv.2407.16221 arXiv:2407.16221 [cs]
- [49] Mariana Y. Noguti, Eduardo Vellasques, and Luiz S. Oliveira. 2020. Legal Document Classification: An Application to Law Area Prediction of Petitions to Public Prosecution Service. In *2020 International Joint Conference on Neural Networks (IJCNN)* (2020-07). 1–8. doi:10.1109/IJCNN48605.2020.9207211 arXiv:2010.12533 [cs]
- [50] Bowen Peng, Jeffrey Quesnelle, Honglu Fan, and Enrico Shippole. 2023. YaRN: Efficient Context Window Extension of Large Language Models. doi:10.48550/arXiv.2309.00071 arXiv:2309.00071 [cs]
- [51] Kamalkumar Rathinasamy, Jayarama Nettar, Amit Kumar, Vishal Manchanda, Arun Vijayakumar, Ayush Kataria, Venkateshprasanna Manjunath, Chidambaram GS, Jaskirat Singh Sodhi, Shoeb Shaikh, Wasim Akhtar Khan, Prashant Singh, Tanishq Dattatray Ige, Vipin Tiwari, Rajab Ali Mondal, Harshini K, S. Reka, Chetana Amancharla, Faiz ur Rahman, Harikrishnan P. A, Indraneel Saha, Bhavya Tiwary, Navin Shankar Patel, Pradeep T. S, Balaji A. J, Priyaprasava, and Mohammed Rafee Tarafdar. 2024. EnterpriseEM: Fine-tuned Embeddings for Enterprise Semantic Search. doi:10.48550/arXiv.2406.00010 arXiv:2406.00010 [cs]
- [52] David Rodriguez, Ian Yang, Jose M. Del Alamo, and Norman Sadeh. 2024. Large language models: a new approach for privacy policy analysis at scale. 106, 12 (2024), 3879–3903. doi:10.1007/s00607-024-01331-9
- [53] Ron Ross. 2020. The Next Generation Security and Privacy Controls—Protecting the Nation’s Critical Assets. (2020). <https://www.nist.gov/blogs/cybersecurity-insights/next-generation-security-and-privacy-controls-protecting-nations> Last Modified: 2025-03-19T09:30-04:00 Publisher: Ron Ross.
- [54] Ron Ross, Stuart Katzke, L. Johnson, Marianne Swanson, Gary Stoneburner, George Rogers, and Annabelle Lee. 2005. Recommended Security Controls for Federal Information Systems. NIST Special Publication (SP) 800-53 (Withdrawn) (2005). doi:10.6028/NIST.SP.800-53
- [55] Kratika Sharma and Satya Kiranmai Tadepalli. [n. d.]. Detecting Duplicate Questions in Online Forums Using Machine Learning Techniques. 10 ([n. d.]).
- [56] Yun Shen and Gianluca Stringhini. 2019. ATTACK2VEC: Leveraging Temporal Word Embeddings to Understand the Evolution of Cyberattacks. doi:10.48550/arXiv.1905.12590 arXiv:1905.12590 [cs]
- [57] Adi Simhi, Itay Itzhak, Fazl Barez, Gabriel Stanovsky, and Yonatan Belinkov. 2025. Trust Me, I’m Wrong: High-Certainty Hallucinations in LLMs. doi:10.48550/arXiv.2502.12964 arXiv:2502.12964 [cs] version: 1.

- [58] Eric Tang, Bangding Yang, and Xingyou Song. 2025. Understanding LLM Embeddings for Regression. doi:10.48550/arXiv.2411.14708 arXiv:2411.14708 [cs]
- [59] Xiaoyu Tian, Sitong Zhao, Haotian Wang, Shuaiting Chen, Yunjie Ji, Yiping Peng, Han Zhao, and Xiangang Li. 2025. Think Twice: Enhancing LLM Reasoning by Scaling Multi-round Test-time Thinking. doi:10.48550/arXiv.2503.19855 arXiv:2503.19855 [cs]
- [60] Christian Tomani, Kamalika Chaudhuri, Ivan Evtimov, Daniel Cremers, and Mark Ibrahim. 2024. Uncertainty-Based Abstention in LLMs Improves Safety and Reduces Hallucinations. doi:10.48550/arXiv.2404.10960 arXiv:2404.10960 [cs]
- [61] Kazi Wali Ullah, Abu Shohel Ahmed, and Jukka Ylitalo. 2013. Towards Building an Automated Security Compliance Tool for the Cloud. In *2013 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom 2013)*. 1587–1593. doi:10.1109/TrustCom.2013.195 ISSN: 2324-9013.
- [62] Dave Van Veen, Cara Van Uden, Louis Blankemeier, Jean-Benoit Delbrouck, Asad Aali, Christian Bluethgen, Anuj Pareek, Malgorzata Polacin, Eduardo Pontes Reis, Anna Seehofnerová, Nidhi Rohatgi, Poonam Hosamani, William Collins, Neera Ahuja, Curtis P. Langlotz, Jason Hom, Sergios Gatidis, John Pauly, and Akshay S. Chaudhari. 2024. Adapted large language models can outperform medical experts in clinical text summarization. 30, 4 (2024), 1134–1142. doi:10.1038/s41591-024-02855-5 Publisher: Nature Publishing Group.
- [63] Zhiqiang Wang, Yiran Pang, and Yanbin Lin. 2024. Smart Expert System: Large Language Models as Text Classifiers. arXiv:2405.10523 <http://arxiv.org/abs/2405.10523> version: 1.
- [64] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. doi:10.48550/arXiv.2201.11903 arXiv:2201.11903 [cs]
- [65] Bingbing Wen, Bill Howe, and Lucy Lu Wang. 2024. Characterizing LLM Abstention Behavior in Science QA with Context Perturbations. doi:10.48550/arXiv.2404.12452 arXiv:2404.12452 [cs]
- [66] Bingbing Wen, Jihan Yao, Shangbin Feng, Chenjun Xu, Yulia Tsvetkov, Bill Howe, and Lucy Lu Wang. 2024. The Art of Refusal: A Survey of Abstention in Large Language Models. doi:10.48550/arXiv.2407.18418 arXiv:2407.18418 [cs] version: 1.
- [67] Bingbing Wen, Jihan Yao, Shangbin Feng, Chenjun Xu, Yulia Tsvetkov, Bill Howe, and Lucy Lu Wang. 2024. Know Your Limits: A Survey of Abstention in Large Language Models. arXiv:2407.18418 [cs] <http://arxiv.org/abs/2407.18418>

- [68] Yingsheng Wu, Yuxuan Gu, Xiaocheng Feng, Weihong Zhong, Dongliang Xu, Qing Yang, Hongtao Liu, and Bing Qin. 2024. Extending Context Window of Large Language Models from a Distributional Perspective. doi:10.48550/arXiv.2410.01490 arXiv:2410.01490 [cs]
- [69] Shaochen Xu, Zihao Wu, and Huaqin Zhao. [n. d.]. *Reasoning before Comparison: LLM-Enhanced Semantic Similarity Metrics for Domain Specialized Text Analysis*. <https://arxiv.org/html/2402.11398v2>
- [70] Ziwei Xu, Sanjay Jain, and Mohan Kankanhalli. 2025. Hallucination is Inevitable: An Innate Limitation of Large Language Models. doi:10.48550/arXiv.2401.11817 arXiv:2401.11817 [cs]
- [71] Yasin Abbasi Yadkori, Ilja Kuzborskij, David Stutz, András György, Adam Fisch, Arnaud Doucet, Iuliya Beloshapka, Wei-Hung Weng, Yao-Yuan Yang, Csaba Szepesvári, Ali Taylan Cemgil, and Nenad Tomasev. 2024. Mitigating LLM Hallucinations via Conformal Abstention. doi:10.48550/arXiv.2405.01563 arXiv:2405.01563 [cs]
- [72] Xiaoju Ye, Zhichun Wang, and Jingyuan Wang. 2025. Infinite Retrieval: Attention Enhanced LLMs in Long-Context Processing. doi:10.48550/arXiv.2502.12962 arXiv:2502.12962 [cs] version: 1.
- [73] Hanning Zhang, Shizhe Diao, Yong Lin, Yi R. Fung, Qing Lian, Xingyao Wang, Yangyi Chen, Heng Ji, and Tong Zhang. 2024. R-Tuning: Instructing Large Language Models to Say ‘I Don’t Know’. doi:10.48550/arXiv.2311.09677 arXiv:2311.09677 [cs]
- [74] Lechen Zhang, Tolga Ergen, Lajanugen Logeswaran, Moontae Lee, and David Jurgens. 2024. SPRIG: Improving Large Language Model Performance by System Prompt Optimization. doi:10.48550/arXiv.2410.14826 arXiv:2410.14826 [cs]
- [75] Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Fei Wu, and Guoyin Wang. 2024. Instruction Tuning for Large Language Models: A Survey. doi:10.48550/arXiv.2308.10792 arXiv:2308.10792 [cs]
- [76] Wenhao Zhu, Hongyi Liu, Qingxiu Dong, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. 2024. Multilingual Machine Translation with Large Language Models: Empirical Results and Analysis. doi:10.48550/arXiv.2304.04675 arXiv:2304.04675 [cs]
- [77] Anni Zou, Wenhao Yu, Hongming Zhang, Kaixin Ma, Deng Cai, Zhuosheng Zhang, Hai Zhao, and Dong Yu. 2024. DOCBENCH: A Benchmark for Evaluating LLM-based Document Reading Systems. doi:10.48550/arXiv.2407.10701 arXiv:2407.10701 [cs] version: 1.

APPENDIX A

PROMPT TEMPLATES

This appendix presents the prompt templates used for the LLM-based analysis, particularly for control comparison and summarisation.

A.1 LLM comparison prompt template

We used the following prompt template for comparing a single pair of controls:

LLM comparison prompt template

System message: You are a security standards analyst.

Prompt: Given one piece of cybersecurity advice (Excerpt A), analyze how Excerpt A relates to Excerpt B. Determine the relationship using one of the following labels:

- **AGREES:** Excerpt A and Excerpt B offer compatible or equivalent guidance.
- **SUBSET:** Excerpt A is a narrower or more specific instance of Excerpt B.
- **SUPERSET:** Excerpt A is broader or more comprehensive than Excerpt B.
- **DISAGREES:** Excerpt A contradicts or opposes Excerpt B.
- **INCOMPARABLE:** The excerpts are unrelated, or the relationship is unclear.

Use your knowledge of cybersecurity frameworks (e.g., NIST SP 800-53, ISO/IEC 27001, CIS Controls, FedRAMP) to assess technical scope, control intent, and implementation.

Excerpt from Rev. `{stdA_name}` (Excerpt A):
`{stdA_excerpt}`

Excerpt from Rev. `{stdB_name}` (Excerpt B):
`{stdB_excerpt}`

Return a dictionary with:

- **relation:** one of AGREES, SUBSET, SUPERSET, DISAGREES, INCOMPARABLE
- **explanation:** a brief justification based on scope, overlap, or differences

If you are not at least 95% confident due to ambiguity or insufficient detail, return `'relation': 'INCOMPARABLE'` and explain why.

This prompt was dynamically generated for each control in Revision A and its mapped candidates from Revision B.

A.2 LLM summarisation prompt template

We used the following prompt template to instruct the LLM to synthesise the results of pairwise control comparisons into a high-level summary:

LLM summarisation prompt template

System message: You are a security standards analyst.

Prompt: The comparison data below maps each control from `{std1_name}` to its semantically similar counterparts in `{std2_name}`. For each pair, the model outputs a **relation** label (AGREES, SUBSET, SUPERSET, DISAGREES, or INCOMPARABLE) and a natural language explanation of why the label was assigned. I am interested in how `{std1_name}` compares to `{std2_name}` at a high level based on these labels and explanations.

The following controls in `{std1_name}` AGREE WITH their corresponding control(s) in `{std2_name}`: `{std1_agrees}`

The following controls in `{std1_name}` ARE MORE COMPREHENSIVE THAN their corresponding control(s) in `{std2_name}`: `{std1_superset}`

The following controls in `{std1_name}` ARE LESS COMPREHENSIVE THAN their corresponding control(s) in `{std2_name}`: `{std1_subset}`

The following controls in `{std1_name}` DISAGREE WITH their corresponding control(s) in `{std2_name}`: `{std1_disagree}`

The following controls in `{std1_name}` ARE INCOMPARABLE WITH their corresponding control(s) in `{std2_name}`: `{std1_incomparable}`

The following controls in `{std1_name}` HAVE BEEN SPLIT UP INTO MULTIPLE CONTROLS in `{std2_name}`: `{std1_split}`

The comparison data below maps each control from `{std2_name}` to its semantically similar counterparts in `{std1_name}`. As above, [...]

Using only the above comparison data, without reference to your previous knowledge, please answer the following questions:

1. What are the similarities between these two standards?
2. What are the differences between these two standards?
3. Are there any inconsistencies between these two standards? That is, would following the guidelines in one standard conflict with those in the other?
4. What broad categories of guidance are generally missing from `{std1_name}`? What types are missing from `{std2_name}`? If I am on a security team, what differences should I be aware of?
5. Highlight what this might reveal about how the controls have evolved between `{std1_name}` and `{std2_name}`.
6. Summarize any notable patterns in the explanations.

Respond with a concise 3-5 paragraph summary.

APPENDIX B

LLM-GENERATED SUMMARIES

This appendix presents the LLM-generated summaries that result from the methodology described in Chapter 4.4. There is a large degree of overlap between the summaries for each of the comparisons. The contents of the following subsections are entirely generated by LLMs, with additional emphasis added to highlight key findings and takeaways.

B.1 Revision 4, compared to Revision 5

The comparison data between Revision 5 and Revision 4 standards reveals several similarities and differences. Similarities include the agreement between corresponding controls in both standards, such as access control, audit and accountability, and incident response. **Many controls in Revision 5 have equivalent guidance in Revision 4**, indicating consistency in the fundamental security and privacy principles. Differences, on the other hand, are evident in the level of detail, scope, and specific requirements. For instance, **Revision 5 provides more comprehensive guidance on security and privacy controls**, including additional related controls and considerations for supply chain risk management, privacy impact assessments, and continuous monitoring.

The differences between the two standards are notable, with Revision 5 generally providing more detailed and expansive guidance on various security and privacy controls. **Revision 5 includes additional requirements, such as the consideration of privacy risks, supply chain risk management, and continuous monitoring**, which are not explicitly mentioned in Revision 4. In contrast, Revision 4 provides more general guidance on certain topics, such as security awareness training and incident response. There are **no apparent inconsistencies between the two standards**, as the guidance in one standard does not conflict with the guidance in the other. However, the **broader scope and more detailed requirements in Revision 5** may require organizations to reassess and update their security and privacy controls to ensure compliance.

The broad categories of guidance generally missing from the Revision 5 standard include specific details on certain topics, such as **security awareness training and incident response, which**

are more comprehensively covered in Revision 4. In contrast, Revision 4 is missing guidance on supply chain risk management, privacy impact assessments, and continuous monitoring, which are more extensively covered in Revision 5. As a member of the security team, it is essential to be aware of these differences and to ensure that the organization's security and privacy controls align with the more comprehensive and detailed requirements in Revision 5. The evolution of controls between Revision 5 and Revision 4 reveals a shift towards more comprehensive and detailed guidance on security and privacy controls, with a greater emphasis on supply chain risk management, privacy impact assessments, and continuous monitoring.

The notable patterns in the explanations include the consistent agreement between corresponding controls in both standards, indicating a strong foundation in fundamental security and privacy principles. The more detailed and expansive guidance in Revision 5 suggests a growing recognition of the importance of supply chain risk management, privacy impact assessments, and continuous monitoring in ensuring the security and privacy of organizational systems and data. The explanations also highlight the importance of ongoing review and update of security and privacy controls to ensure compliance with evolving standards and requirements. Overall, the comparison data reveals a maturation of security and privacy controls, with Revision 5 providing more comprehensive and detailed guidance to support organizational security and privacy posture.

B.2 Revision 3, compared to Revision 5

The comparison data between Revision 5 and Revision 3 standards reveals several similarities and differences. Similarities include the shared control intent and implementation guidance for various security controls, such as access control, incident response, and risk management. Both standards emphasize the importance of developing and implementing security policies, procedures, and controls to protect organizational information systems. The differences between the two standards lie in the **level of detail, scope, and specificity of the guidance provided**. The Revision 5 standard tends to be more comprehensive, providing additional details, examples, and related controls, whereas the Revision 3 standard is more general and focused on specific aspects of security.

The Revision 5 standard includes **more detailed guidance on topics such as security and**

privacy risk management, supply chain risk management, and incident response. In contrast, the **Revision 3 standard provides more general guidance on security planning, security assessments, and security authorization.** There are **no apparent inconsistencies** between the two standards, and following the guidelines in one standard does not conflict with guidelines in the other. However, the Revision 5 standard tends to be more prescriptive, providing specific requirements and examples, whereas the Revision 3 standard is more flexible and allows for organization-defined controls and procedures.

The broad categories of guidance generally missing from the Revision 5 standard include more general and high-level guidance on security planning and security assessments. In contrast, the Revision 3 standard lacks detailed guidance on specific security controls, such as security and privacy risk management, supply chain risk management, and incident response. As a member of the security team, it is essential to be aware of these differences and to understand that the Revision 5 standard provides more comprehensive and detailed guidance on specific security controls, whereas the Revision 3 standard provides more general guidance on security planning and assessments. The evolution of controls between Revision 5 and Revision 3 reveals a shift towards more comprehensive and detailed guidance on specific security controls, reflecting the increasing complexity and sophistication of cyber threats.

The comparison data also reveals notable patterns in the explanations. Many of the Revision 5 controls are more comprehensive and detailed, providing additional examples, related controls, and specific requirements. In contrast, the Revision 3 controls tend to be more general and focused on specific aspects of security. This pattern suggests that the Revision 5 standard is more focused on providing detailed guidance on specific security controls, whereas the Revision 3 standard is more focused on providing general guidance on security planning and assessments. Overall, the comparison data highlights the importance of understanding the differences between the two standards and the need for organizations to adapt their security controls and procedures to the evolving cyber threat landscape.

The differences between the two standards may also **reflect changes in the regulatory and threat landscape.** The Revision 5 standard may be more comprehensive and detailed due to the

increasing sophistication of cyber threats and the need for more specific guidance on security controls. In contrast, the Revision 3 standard may be more general due to its focus on providing high-level guidance on security planning and assessments. As the cyber threat landscape continues to evolve, it is likely that security standards will continue to adapt and provide more comprehensive and detailed guidance on specific security controls.

B.3 Revision 2, compared to Revision 5

The comparison data between Revision 2 and Revision 5 standards reveals several similarities and differences. Similarities include the emphasis on access control, audit and accountability, configuration management, and incident response. Both standards provide guidance on these topics, indicating a consistent approach to information security and privacy. However, the differences lie in the level of detail, specificity, and scope of the guidance. **The Revision 5 standard provides more comprehensive and detailed guidance on various security controls, including security and privacy planning, risk management, and supply chain risk management.**

The differences between the two standards are notable, with **Revision 5 providing more detailed and specific guidance on various security controls.** For example, Revision 5 includes more detailed requirements for access control, audit and accountability, and configuration management. In contrast, Revision 2 provides more general guidance on these topics. Additionally, Revision 5 includes new controls and guidance on topics such as privacy, supply chain risk management, and cybersecurity. **The Revision 2 standard, on the other hand, focuses on more traditional security controls, such as access control and incident response.** There are **no apparent inconsistencies** between the two standards, and following the guidelines in one standard does not conflict with guidelines in the other.

The broad categories of guidance generally missing from the Revision 5 standard include more detailed technical specifications for certain security controls, such as cryptography and network security. In contrast, the Revision 2 standard lacks guidance on newer security topics, such as privacy, supply chain risk management, and cybersecurity. **The missing controls in Revision 5 can be characterized as technical implementation details, while the missing controls**

in Revision 2 can be characterized as emerging security topics. As a member of the security team, it is essential to be aware of these differences and to understand the evolving nature of security controls and guidance.

The evolution of controls between Revision 5 and Revision 2 reveals a shift towards more comprehensive and detailed guidance on security and privacy topics. The Revision 5 standard includes more **emphasis on privacy, supply chain risk management, and cybersecurity**, indicating a recognition of the increasing importance of these topics in modern information security. The notable patterns in the explanations include the consistent emphasis on access control, audit and accountability, and configuration management, as well as the increasing focus on emerging security topics. Additionally, the explanations highlight the importance of continuous monitoring, risk management, and incident response in both standards.

Overall, the comparison data highlights the evolving nature of security controls and guidance. The Revision 5 standard provides more comprehensive and detailed guidance on various security controls, while the Revision 2 standard focuses on more traditional security controls. Understanding these differences is essential for security teams to ensure they are implementing effective security controls and staying up-to-date with emerging security topics. The patterns in the explanations reveal a consistent emphasis on fundamental security topics, as well as a growing recognition of the importance of emerging security topics, such as privacy and supply chain risk management.

B.4 Revision 0, compared to Revision 5

The comparison data between Revision 5 and Revision 0 standards reveals several similarities and differences. Similarities include the shared goals and objectives of various controls, such as access control, incident response, and risk assessment. Both standards provide guidance on these topics, indicating a common foundation for security and privacy practices. However, the differences lie in the level of detail, specificity, and comprehensiveness of the guidance. The Revision 5 standard tends to provide more detailed and specific requirements, whereas the Revision 0 standard offers more general guidance.

The differences between the two standards are notable, with **Revision 5 providing more**

comprehensive guidance on various controls, such as configuration management, continuous monitoring, and supply chain risk management. In contrast, the **Revision 0** standard lacks the same level of detail and specificity, but provides more general guidance on certain topics. There are **no apparent inconsistencies** between the two standards, as the guidelines in one standard do not conflict with those in the other. However, the differences in guidance may require security teams to adapt their practices to align with the more comprehensive and detailed requirements of the r5 standard.

The broad categories of guidance generally missing from the Revision 5 standard include **specific details on certain topics, such as the use of FIPS 140-2 compliant cryptography and the implementation of specific security controls**. In contrast, the Revision 0 standard lacks guidance on topics like configuration management, continuous monitoring, and supply chain risk management. The **missing controls in Revision 5 tend to be more technical and specific, whereas those in Revision 0 are more general and high-level**. Security teams should be aware of these differences and adapt their practices accordingly. The evolution of controls between Revision 5 and Revision 0 suggests a **shift towards more comprehensive and detailed guidance**, with a greater emphasis on risk management, continuous monitoring, and supply chain risk management.

The comparison data reveals notable patterns in the explanations, with the Revision 5 standard consistently providing more detailed and specific requirements. This suggests that the Revision 5 standard is more comprehensive and up-to-date, reflecting the evolving security and privacy landscape. The explanations also highlight the importance of risk management, continuous monitoring, and supply chain risk management in the Revision 5 standard, indicating a greater emphasis on these topics. Overall, the comparison data provides valuable insights into the similarities and differences between the Revision 5 and Revision 0 standards, and highlights the importance of adapting security practices to align with the more comprehensive and detailed requirements of the Revision 5 standard.

The differences between the two standards may indicate a **maturation of security and privacy practices**, with the Revision 5 standard reflecting a more nuanced and comprehensive

approach to security and privacy. The increased emphasis on risk management, continuous monitoring, and supply chain risk management in the Revision 5 standard suggests a recognition of the evolving security threats and the need for more proactive and adaptive security practices. Security teams should be aware of these differences and adapt their practices to align with the more comprehensive and detailed requirements of the Revision 5 standard, ensuring that their organizations remain secure and resilient in the face of emerging threats.