Causal Knowledge Graphs for Actuarial Science: A Large-Language-Model and Machine learning Approach to reserving and predictive modeling

Group: KOFFI Konan Mardochee
DIBI Axel
MOHAMED EL HAFED Ismail

ISFA
Teacher: Aurélien Couloumy

April 25, 2025

Abstract

We present a framework that integrates causal knowledge graphs, large-language models (LLMs), and graph neural networks (GNNs) to advance actuarial risk modeling and reserving. First, we develop a six-step pipeline to transform unstructured text (e.g. NTSB accident reports) into structured causal graphs via text chunking, LLM-based event–relation extraction, fuzzy clustering, and graph aggregation. Next, we apply these meta-graphs to reserving—using loss-severity proxies for RBNS and targeted IBNR estimation—and to predictive modeling via case–case similarity graphs and GNN classifiers that predict safety recommendations. A comprehensive case study on 15 highway accidents validates our approach, achieving 76% accuracy and uncovering key driver–environment causal patterns. Our results demonstrate that machine-learning-augmented knowledge graphs provide a scalable, interpretable complement to traditional actuarial methods, enabling richer causal inference and enhanced data-driven decision support.

Contents

1 Introduction

The domain of actuarial science, traditionally reliant on established risk assessment and provisioning models, increasingly encounters challenges posed by the sheer volume and heterogeneity of modern data. Conventional approaches are being tested by the need to incorporate more granular and diverse information streams. In response to these evolving demands, knowledge graphs are emerging as a particularly potent methodological advancement. Offering an inherently intuitive framework, knowledge graphs possess the unique capacity to represent and logically connect disparate information sources, effectively weaving together structured quantitative data with rich, often underutilized, unstructured qualitative insights within a unified network of entities and their relationships [cf. 10]. This paper undertakes an investigation into the role of knowledge graphs as a sophisticated complement to existing actuarial techniques. We place particular emphasis on their capacity for illuminating causal relationships underlying loss events, enhancing data-driven risk analyses, and exploring their potential utility in supporting the complex task of estimating technical provisions.

The impetus for this research stems from recognizing specific limitations within long-established actuarial methodologies. While traditional approaches, such as loss development triangles and probabilistic frameworks like the Chain Ladder or Bornhuetter-Ferguson methods [cf. 3, 15], have served as cornerstones for decades, their efficacy is often constrained by a reliance on highly aggregated historical data. Consequently, the granularity and qualitative depth inherent in sources like accident reports, expert analyses, or textual descriptions of loss events frequently remain untapped by conventional models. It is precisely in this context that knowledge graphs demonstrate a significant advantage. They possess an intrinsic aptitude for consolidating disparate data streams, representing entities (as nodes) and their complex interrelations (as edges). This architectural flexibility is particularly conducive to embedding causal relationships [cf. 9], allowing for the inferential linking of event sequences or parameter effects. Within an actuarial framework, the ability to capture and model the underlying causal mechanisms driving insurance losses promises a more profound understanding of risk factors, potentially culminating in more accurate and robust estimations of necessary reserves and premiums.

To develop our argument systematically, this study proceeds through three interconnected stages. We commence by establishing the necessary theoretical groundwork, delving into the fundamental concept of causality and the structural properties of causal graphs [cf. 9, 10], which is crucial for appreciating how complex loss events can be modeled with greater explanatory power. Subsequently, we detail a comprehensive modeling approach designed to integrate multifaceted data layers. This involves not only leveraging structured datasets but also processing and enriching unstructured textual information through the sophisticated application of large language models (LLMs) [cf. 12, 4, 11]. We explore how these diverse components can be synergistically combined within a knowledge graph framework, aiming to facilitate more automated inference and robust decision support. Finally, the practical viability and implications of this approach are illustrated through an in-depth case study utilizing accident reports curated by the National Transportation Safety Board (NTSB). This empirical example serves to demonstrate how key variables extracted from real-world incident data can be effectively interconnected within a graph structure. Our analysis of the NTSB case considers the results from the dual perspectives of technical provisioning and preventative risk management. Critically, we also address the inherent limitations observed when applying such graph-based models purely for technical provisioning tasks, proposing alternative, potentially more impactful applications in enhancing risk prevention strategies and improving overall data quality management.

By seeking to integrate qualitative contextual understanding with rigorous quantitative methodologies, this research endeavors to contribute to a more holistic and anticipatory paradigm within actuarial science. The incorporation of knowledge graphs into risk management workflows not only facilitates the development of more dynamic modeling capabilities, responsive to new information, but also underscores the critical role of causal reasoning in truly comprehending the complex, multifaceted nature of insurance loss phenomena. Ultimately, our objective is to demonstrate that while knowledge graphs may not serve as a wholesale replacement for established actuarial techniques, their synergistic integration can significantly augment the predictive accuracy and analytical responsiveness of contemporary risk analysis frameworks.

2 What is Causality and What are Causal Graphs

2.1 Defining Causality

Understanding causality in the context of risk modeling and actuarial science is a multifaceted endeavor that requires a rigorous approach to discerning the relationships between events, their underlying factors, and their ultimate outcomes [1, 9, 17]. Jaimini and Sheth introduce CausalKG, a framework that enriches knowledge graphs with explicit cause-effect relations through interventional and counterfactual reasoning, thereby enhancing the explainability of artificial-intelligence models in sensitive domains [1]. This approach underlines the premise that a clear delineation of causal links is not only essential for predicting outcomes but also vital for validating and interpreting the decisions made by complex models, a notion that resonates with traditional actuarial methodologies. Early foundational work by Reid on claim reserves in general insurance established methodological insights into how uncertainty and the progression of loss events can be systematically analyzed, setting the stage for later developments that incorporate causal reasoning into risk estimation [17]. In a similar vein, Gamonet and Le Sujet offer an operational-risk-modeling perspective that leverages Bayesian networks—probabilistic graphical models that inherently rely on causality—to capture the sequential dependencies and interactions among various risk factors, thereby providing a coherent structure to analyze events that are both rare and high-impact [9]. Collectively, these research contributions underscore that integrating causality into risk assessment enables actuaries to simulate potential interventions and hypothetical scenarios, ensuring that the models not only achieve statistical accuracy but also adhere to a logical, rational understanding of cause and effect. This comprehensive synthesis, grounded in interventional, counterfactual, and probabilistic frameworks, advances the state of the art in both theory and practice, ultimately bridging the gap between abstract causal inference and its practical implementation in actuarial science.

2.2 Causal Graphs/Knowledge Graphs: Structure and Purpose

At their core, both causal graphs and knowledge graphs function as powerful network-based representations engineered to encapsulate the rich semantic interplay connecting various entities. Their fundamental structure is intrinsically linked to their purpose: enabling sophisticated reasoning and informed decision support. Building upon the conceptual framework advanced by Jaimini and Sheth (2022) [9], these graphs employ a systematic architecture where individual nodes signify variables or real-world concepts, while directed edges explicitly represent causal links or other semantic relationships. This design not only underpins the capacity for advanced interventional and counterfactual reasoning but also inherently supports the dynamic assimilation and evolution of knowledge as heterogeneous data sources are integrated over time. The architectural underpinnings of a knowledge graph, often adhering to established standards like RDF (Resource Description Framework) as noted by Knight (2025) [10], are specifically geared towards unifying disparate pieces of information into a coherent, context-preserving structure. Such unification is crucial for facilitating automated inference, thereby effectively bridging the often-significant gap between raw data accumulation and the generation of truly actionable insights. Moreover, as highlighted by Brack et al. (2022) [1], the practical utility of these graphs hinges critically on the adoption of robust construction methodologies and stringent quality assurance protocols. These are indispensable for ensuring the graph's fidelity in modeling complex, multidimensional phenomena—a necessity in high-consequence domains like actuarial science, where accurately mapping the intricate interplay between myriad risk factors is paramount. Ultimately, the synergistic design of causal and knowledge graphs serves a dual mandate. They provide not only a detailed semantic map illustrating the complex web of relationships between entities but also a potent predictive framework grounded in probabilistic and logical inference. This combination empowers practitioners not merely to assess risk through meticulously modeled cause-and-effect pathways, but also to dynamically update and enrich these sophisticated representations as new evidence and understanding emerge from the continuous influx of data.

2.3 Bayesian Network in Actuarial applications

Bayesian networks emerge at a compelling confluence of probability theory and graph theory, offering a framework particularly well-attuned to navigating the inherent complexities and uncertainties prevalent in actuarial science. Their application has proven highly valuable for capturing the intricate dependencies that exist among various risk factors and for modeling the temporal evolution of loss events over time. As effectively demonstrated by Gamonet and Le Sujet (2009) [6], these networks find significant utility in operational risk modeling. They allow actuaries to represent the complex interactions between contributing factors—such as exposure indicators, event frequencies, and potential loss severities—thereby enabling more accurate forecasting of claim distributions, especially when dealing with rare yet potentially extreme events. The conceptual lineage for such probabilistic modeling can be traced back to foundational investigations like Reid's (1978) [14] seminal work on claim reserving. Reid's exploration of the stochastic nature of claims highlighted the essential need for models that could handle uncertainty dynamically. In this respect, Bayesian networks provide a systematic and powerful mechanism for continuously updating risk assessments as new data points become available. Their structure, typically represented as a directed acyclic graph (DAG), explicitly maps out dependencies. This architecture facilitates the computation of conditional probabilities, precisely reflecting how the occurrence (or non-occurrence) of one event influences the likelihood of subsequent events—a calculation fundamental to making informed decisions under uncertain conditions. Furthermore, the true power of Bayesian networks in this context is significantly amplified when they are employed not just for predictive modeling but also for exploring causal relationships through counterfactual and interventional reasoning. These advanced techniques, as emphasized in the work of Jaimini and Sheth (2022) [9] concerning causal knowledge graphs, allow the network to simulate the potential effects of specific changes or interventions. This capability supports the critical 'what-if' analyses essential for robust underwriting decisions, proactive risk management strategies, and refining reserve adequacy assessments. Thus, Bayesian networks serve as a vital tool, bridging probabilistic inference with causal understanding within the actuarial domain.

2.4 Advantages of Using Causal Graphs

Employing causal graphs within the analytical toolkit brings forth substantial benefits, particularly when navigating the complex landscapes typical of actuarial science where untangling intricate risk interdependencies is fundamental. A primary advantage, powerfully illustrated by frameworks like CausalKG discussed by Jaimini and Sheth (2022) [9], lies in the explicit modeling of cause-and-effect relationships. This inherent focus on causality not only dramatically improves the transparency and interpretability of risk models but also unlocks sophisticated analytical capabilities. Practitioners gain the ability to perform advanced interventional and counterfactual reasoning, allowing them to simulate the likely impacts of potential strategic changes or external shocks and rigorously evaluate various 'what-if' scenarios—an essential capacity for sound decision-making in environments fraught with uncertainty. Beyond enhancing analytical depth, the very structure inherent to causal graphs—wherein nodes represent key variables and directed edges signify influence or dependence—greatly simplifies the challenging task of integrating diverse, heterogeneous data sources. This structured approach ensures that information from varied origins can be fused into a cohesive, unified schema. Importantly, contextual nuances embedded within qualitative or unstructured data are preserved during this integration, supporting more robust and reliable inferential processes, a point underscored in the construction strategies outlined by Brack et al. (2022) [1]. Such integrative power proves especially valuable when confronted with the dynamic and multifaceted datasets frequently encountered in risk modeling, enabling actuaries to continuously refine and update their models as fresh information becomes available. Furthermore, the capacity of causal graphs to incorporate established probabilistic frameworks, such as those embodied within the Bayesian networks discussed by Gamonet and Le Sujet (2009) [6] and building upon the foundational claim reserving principles explored by Reid (1978) [14], adds another layer of utility. This allows analysts to formally account for both the inherent stochasticity of risk events and the underlying causal drivers shaping those events. Consequently, the application of causal graphs extends well beyond improved model explainability and seamless data integration; it cultivates

enhanced predictive accuracy and fosters more adaptive, responsive risk assessment methodologies, positioning these graphs as a truly vital instrument in modern actuarial practice.

2.5 Challenges and Considerations

Despite the considerable promise offered by causal graphs in actuarial science, their effective implementation and sustained maintenance entail navigating a distinct set of practical hurdles and critical considerations. Harnessing their full potential requires acknowledging these potential impediments upfront. A primary concern, frequently emphasized in discussions of causal inference such as by Jaimini and Sheth (2022) [9], revolves around the fundamental issues of data quality and availability. Robust causal discovery and inference are intrinsically dependent upon access to accurate, comprehensive datasets; significant deficiencies or biases in the underlying data can propagate through the model, leading inevitably to skewed or unreliable conclusions. Furthermore, the inherent complexity involved in faithfully modeling multifaceted risk scenarios presents a substantial modeling challenge. Capturing the full spectrum of relevant variables and their intricate web of interdependencies, a difficulty highlighted even in early foundational work on claim reserving by Reid (1978) [14], can prove elusive. Oversimplification or inaccuracies in depicting these causal structures within the graph can significantly undermine the model's validity. Gamonet and Le Sujet (2009) [6] also draw attention to related difficulties when utilizing Bayesian networks; while these offer potent probabilistic reasoning capabilities, their outputs can be highly sensitive to the initial assumptions made and the specific parameterizations chosen—inputs that may not remain static or entirely accurate within the dynamic operational environment of risk management. An additional layer of complexity arises from the practical task of integrating heterogeneous data sources into a single, unified causal framework. As noted by Brack et al. (2022) [1], this is far from a trivial undertaking. It demands meticulous design choices, careful semantic mapping, and continuous recalibration efforts to preserve consistency and meaning across disparate data types. This process, often requiring significant domain expertise, can also be computationally intensive, particularly when dealing with large-scale datasets. Taken together, these considerations—spanning from ensuring data integrity and managing model complexity to addressing computational demands and the ongoing necessity for rigorous validation—underscore that leveraging causal graphs effectively requires more than just technical proficiency. It necessitates a thoughtful equilibrium between theoretical rigor and pragmatic applicability, ensuring that the resulting models are not only sophisticated but also robust, reliable, and genuinely useful for enhancing actuarial risk assessments in real-world settings.

3 Modeling Approach: Data, Large Language Models, and Processes

3.1 Data Acquisition and Preparation

The foundation for constructing our knowledge graph rests upon a systematic approach to acquiring and preparing data, primarily originating from unstructured textual sources such as PDF documents and incident reports. Identifying relevant source materials constitutes the initial phase. Once identified, a comprehensive extraction of all textual content from each file is undertaken; this broad initial capture ensures that potentially valuable risk-related insights embedded within the documents are not overlooked, aligning with the principles of thorough data gathering discussed in related research [1]. Following this initial extraction, the raw text is segmented into more granular, contextually coherent units—a technique commonly referred to as text chunking. This segmentation is critical for enabling detailed analysis by isolating specific events, descriptions, or causal sequences within the larger narrative of each report. To maintain traceability and facilitate later aggregation, each resulting text chunk is meticulously tagged with a unique case reference (UCR), clearly delineating its origin within a specific incident or document. Subsequently, these segmented chunks undergo a rigorous pre-processing pipeline. This involves essential cleaning steps to remove noise, normalization procedures to standardize formats, and the extraction or assignment of relevant metadata. Crucially, this stage leverages advanced natural language processing (NLP) techniques, potentially enhanced by

the capabilities of large language models (LLMs), to further standardize terminology and annotate the text with semantic information, reflecting modern approaches to text understanding [cf. 4, 5]. The outcome of this intensive pre-processing for each UCR's associated chunks is the generation of structured data representations, frequently in JSON format. These structured files aim to encapsulate the inferred causal relationships present in the original text, effectively translating narrative descriptions into a preliminary map of the event's causal structure. Recognizing the potential for variability in terminology and phrasing across different reports or even within a single document ('diversity of labels'), a final critical step involves applying fuzzy matching and clustering algorithms to the generated JSON representations. These techniques serve to identify and align semantically similar entities or relationships expressed differently, thereby refining and normalizing the information across the entire corpus. This normalization culminates in the aggregation of the processed information into a unified, cohesive dataset. This carefully orchestrated, multi-stage process not only mirrors effective data handling strategies seen in practice but also integrates contemporary methodologies [cf. 1, 6], establishing a solid and reliable groundwork for the subsequent generation and analysis of the causal knowledge graph.

3.2 Causal graph generation

With the data acquisition and preparation phases complete, the subsequent critical stage involves constructing the causal graph itself—a network designed to visually and structurally represent the intricate relationships discerned from the textual reports. The generation commences with the systematic identification of individual events within each pre-processed text chunk. In the graph's topology, each distinct event is designated as a node. Following identification, these event nodes are meticulously analyzed to uncover potential causal connections, ascertaining instances where one event might have directly precipitated, or indirectly contributed to, another. This requires a sophisticated blend of analytical techniques, leveraging both statistical measures of co-occurrence and deeper semantic analysis methods capable of interpreting the nuanced causal indicators embedded within the text. Parallel to identifying event nodes, the process focuses on extracting key entities and their associated attributes from each text chunk. Here again, the capabilities of natural language processing, augmented by large language models, are instrumental. These tools are employed to automatically annotate the identified entities and the relationships between them, typically producing a structured output format (such as JSON). This structured representation explicitly details the influential connections between events, thereby forming the foundational blueprint for the graph. Within this blueprint, directional links, or edges, are established to signify the flow of causality. The process aims to map not only straightforward cause-and-effect sequences but also more complex webs of influence, potentially incorporating probabilistic frameworks analogous to Bayesian networks to model scenarios where causality is less deterministic [cf. 1]. A crucial step following the initial extraction from individual chunks is the consolidation of information pertaining to a single Unique Case Reference (UCR). This aggregation merges potentially fragmented or overlapping representations derived from different text segments of the same report into a single, comprehensive causal map for that specific case. Advanced fuzzy matching and clustering techniques play a vital role here, ensuring that semantically equivalent events or entities described differently across chunks are correctly aligned and discrepancies are resolved. This normalization procedure is essential for guaranteeing the final graph's internal coherence and consistency across diverse cases, which in turn supports more reliable statistical analysis and inference drawing upon the entire dataset. Finally, once the core structure of the graph is established, its utility is further enhanced by applying logical rules and probabilistic reasoning mechanisms. These allow for the inference of new relationships or causal pathways that may not have been explicitly stated in the source text but are logically or probabilistically implied by the existing structure. Crucially, this phase incorporates the powerful interventional and counterfactual reasoning approaches highlighted in the literature [cf. 1]. By enabling the simulation of hypothetical scenarios—such as assessing the likely outcome had a specific risk factor been mitigated—the system moves beyond static representation. It allows for dynamic exploration and validation of the inferred causal chains, transforming the graph into a powerful analytical tool.

3.3 Integration of Large language models

The incorporation of Large Language Models (LLMs) represents a pivotal enhancement to our methodology, significantly augmenting the capacity to extract and interpret complex causal relationships embedded within unstructured textual data. Within this framework, LLMs function as highly sophisticated instruments for natural language understanding. Their power is particularly evident in the ability to parse raw text and distill structured representations, such as semantic triplets comprising subject-predicate-object constructions. These extracted triplets become the fundamental building blocks of the causal graph, identifying key events or entities and mapping the inferred relationships connecting them. Beyond simple entity recognition, LLMs demonstrate proficiency in discerning subtle, yet critical, contextual indicators of causal directionality—recognizing temporal sequencing markers, specific syntactic structures, or vocabulary choices that signal cause and effect. Once the initial text segmentation and pre-processing are completed, LLMs are applied systematically to each text chunk. Their task is to generate a structured output, often formatted as JSON, that meticulously captures the underlying causal architecture detected within the narrative. Such outputs typically feature detailed annotations identifying distinct events, their characterizing attributes, and the inferred relational links that establish potential cause-and-effect chains. For instance, through careful analysis of an incident report's narrative flow, an LLM can effectively identify an initial system failure and trace its subsequent propagation into operational disruptions, formally encoding this discovered relationship within the generated structured data. Furthermore, the advanced embedding capabilities inherent to modern LLMs offer substantial advantages for ensuring semantic consistency across the dataset. By transforming textual descriptions into dense vector representations in a high-dimensional space, these models allow the system to recognize and measure semantic similarity between different phrasings of the same underlying concept encountered across various reports or text chunks. This vector-based understanding directly facilitates the crucial processes of clustering and fuzzy matching, which are essential for accurately normalizing disparate data entries. Consequently, events or factors described using varied terminology can be correctly aligned, leading to a more coherent and internally consistent overall causal graph. These embeddings also prove invaluable in bridging inferential gaps where explicit causal language is absent, allowing relationships to be inferred from contextual proximity and semantic relatedness, thereby significantly enhancing the graph's completeness. Beyond structural extraction and normalization, LLMs contribute significantly to the graph's analytical capabilities by supporting advanced reasoning tasks. As explored in the literature concerning causal frameworks [cf. 1], their generative potential can be harnessed for counterfactual and interventional analyses. This involves generating plausible alternative scenarios or predicting outcomes based on hypothetical modifications to the input data (e.g., simulating the impact of a preventative measure). This transforms the knowledge graph from a static repository into a dynamic analytical environment. It allows for iterative refinement as new data or insights emerge and empowers actuaries to conduct sophisticated scenario modeling grounded in both empirical evidence extracted from text and deep semantic understanding [cf. 4, 5]. In essence, integrating LLMs achieves more than just streamlining the complex conversion of unstructured text into structured causal knowledge. It profoundly enhances the resulting graph's interpretability, adaptability, and analytical power. This synergistic integration lays a robust foundation for subsequent analytical endeavors, including the practical application demonstrated in the NTSB case study, where these sophisticated techniques are directly applied to domain-specific data for improved risk assessment and decision support.

3.4 Overall Workflow and Process Diagram

The end-to-end pipeline for building our causal knowledge graph is summarized in Figure 1 **??**. Starting from raw documents (PDFs, reports), we automatically chunk and pre-process text—cleaning, normalizing, and annotating it via large language models to extract event–relation triplets [12, 11]. We then harmonize labels and merge these structured fragments into per-case graphs, before consolidating all cases into a single meta-graph [1]. This meta-graph not only captures the intricate web of causal links, but also serves as the foundation for downstream analytics: for example, we can derive a case–case similarity graph (e.g. via Jaccard on shared factors) and train graph neural networks to

predict labels such as safety recommendations [6], or use sub-graph queries for scenario and counterfactual analysis [9]. Through careful tuning of embedding-based clustering and parallelized processing, the workflow scales to hundreds or thousands of cases, enabling rapid, data-driven risk assessment and decision support in actuarial contexts.

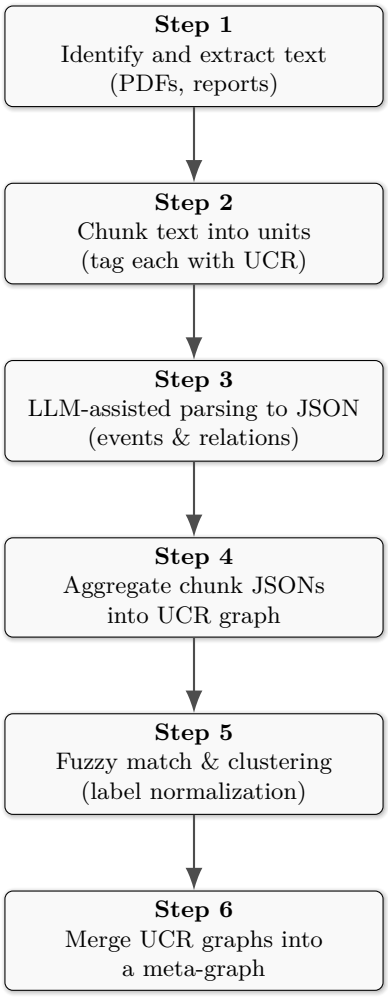

Figure 1: workflow for causal knowledge graph construction

4 Case Study: NTSB and Actuarial Applications

4.1 Overview of the NTSB Case

To ground our exploration of knowledge graphs in a practical actuarial context, we utilize investigation reports from the National Transportation Safety Board (NTSB) as a rich case study dataset. These NTSB investigations delve into major highway accidents across the United States, providing an exceptionally detailed repository of information crucial for advancing risk understanding, particularly when analyzed through the lens of knowledge graph methodologies. The specific reports examined within this study predominantly focus on complex incidents, including large-scale multivehicle collisions, hazardous chain-reaction pile-ups, and instances of catastrophic infrastructure failure, such as bridge collapses. The inherent value of these reports lies in their remarkable granularity and multidimensional nature. They typically contain meticulous factual descriptions of the accident sequence, precise event timelines, thorough documentation of prevailing environmental conditions, captured vehicle operational data (often from event data recorders or EDRs), detailed insights into driver behavior preceding the incident, and comprehensive post-incident analyses. This wealth of information collectively renders the NTSB archives an invaluable resource for constructing sophisticated risk models and probing the intricate causal pathways leading to severe loss events. Analysis across these se-

lected NTSB cases reveals several pervasive contributing factors and recurring accident typologies. Key themes frequently include the significant impact of driver impairment—often substantiated by toxicology reports identifying substances like alcohol, cocaine, PCP, or cannabis—alongside documented instances of excessive speed relative to conditions or posted limits. Adverse environmental factors, such as icy roadways or wet weather conditions, often act as critical amplifiers or triggers, especially in combination with high speeds leading to chain reactions or large pile-ups. Furthermore, systemic deficiencies related to roadway maintenance or inadequate infrastructure integrity emerge as contributing elements in certain catastrophic events. The NTSB data captures not only the immediate physics of the collisions but also crucial contextual elements like road geometry design, pre-crash communications or warnings, and the effectiveness of emergency response efforts. This profound depth and informational diversity make the NTSB dataset exceptionally well-suited for demonstrating the power of a knowledge graph framework. Such a framework can effectively integrate these multifaceted data points, bridging quantitative variables (e.g., vehicle speeds, impact forces, casualty counts) with qualitative determinants (e.g., documented driver negligence, identified infrastructure weaknesses). The resulting knowledge graph manifests as an interconnected network where nodes represent individual accidents, specific risk factors, vehicle characteristics, or environmental states. Actuaries can navigate this intricate network to uncover latent patterns, rigorously establish plausible causal links between contributing factors and outcomes, and ultimately infer potential loss characteristics with greater nuance. Leveraging the NTSB cases within this knowledge graph paradigm facilitates a significantly richer comprehension of risk by meticulously capturing the unique details and confluence of factors in each incident. This approach particularly highlights the capacity to differentiate between more commonplace accidents and truly extreme, high-severity events, thereby directly supporting the development of predictive models capable of better anticipating catastrophic losses. Furthermore, the detailed causal mapping aids significantly in processes like liability attribution and the precise identification of compensable exposures—both essential activities for accurate pricing and prudent reserving within non-life insurance operations. In essence, this overview of the NTSB case data provides not just the empirical bedrock for rigorous actuarial investigation but also illuminates the transformative potential of knowledge graphs for revealing hidden interdependencies within complex, real-world accident datasets.

4.2 Analysis of Extracted Risk Factors and Accident Characteristics

Our in-depth examination of the selected NTSB reports brought to light a consistent set of core risk factors and defining accident characteristics that fundamentally shape the causal narratives and severity indicators associated with these catastrophic events. This synthesis highlights the critical elements extracted from the data and explores their direct implications. Several driver-centric risk factors repeatedly emerged from the analysis. Driver impairment stands out as a predominant causal element, frequently corroborated by toxicology findings revealing the presence of alcohol or illicit substances such as cocaine, PCP, and cannabis. Such impairments often appear intertwined with documented histories of speeding violations or patterns of reckless driving behavior, manifesting critically through actions like excessive acceleration or a blatant disregard for traffic control signals and regulations. Independently, excessive speed itself proves to be a recurrent factor across diverse accident contexts, whether contributing to loss of control at urban intersections or escalating the severity of collisions on high-speed highways and toll lanes. Beyond driver actions, environmental conditions contribute significantly to accident causation and severity. Inclement weather, particularly the presence of icy road surfaces or conditions of reduced visibility due to wet weather, frequently acts as a catalyst or major contributing factor. In numerous documented instances, the dangerous combination of adverse weather and high vehicle speeds precipitated severe chain-reaction collisions or complex multivehicle pile-ups. Furthermore, identifiable deficiencies within the transportation infrastructure itself—ranging from the structural degradation of bridges to inadequate or confusing roadway signage—can substantially exacerbate risk, introducing elements of unpredictability that heighten the potential for accidents. The meticulous level of detail contained within the NTSB reports significantly deepens our understanding of the accident dynamics themselves. For instance, event data recorder (EDR) readouts, precise vehicle speed measurements, and documented driver actions immediately preceding impact allow for a

granular reconstruction of the event sequence. This high-resolution data facilitates the identification of critical operational thresholds—such as speeds drastically exceeding posted limits within specific zones—that show strong correlation with the likelihood of fatal outcomes. The reports also enable a clear characterization of recurring accident archetypes observed in the data, including multivehicle chain-reaction events, complex collision sequences unfolding at intersections, hazardous wrong-way driving incidents, and catastrophic structural failures impacting roadways or bridges. In nearly every case, the ultimate magnitude of the loss appears directly linked not only to the intensity of the primary risk factors (e.g., degree of impairment, speed delta) but also to the inherent complexity of the accident environment itself, such as the number of vehicles entangled or the overall severity of casualties. The profound richness and granularity of these NTSB datasets offer compelling support for adopting a predictive risk modeling approach particularly amenable to actuarial science applications. By systematically incorporating the identified risk factors (driver behavior, environmental states, infrastructure conditions) into a knowledge graph structure, intricate relationships between these elements and the resultant accident outcomes can be explicitly mapped. Such a graph, in turn, can power predictive models capable of estimating loss frequency and severity with potentially greater effectiveness than traditional models reliant solely on aggregated historical data. The demonstrable strong association between specific factors, like severe driver impairment, and high-fatality outcomes could directly inform the differentiation of premium pricing strategies or refine the calibration of claims reserves allocated for high-severity potential events. Moreover, the detailed granularity afforded by these reports permits a sophisticated segmentation of accidents based on type and specific risk profile. Multivehicle collisions occurring under icy conditions, for example, might be modeled as a distinct 'catastrophic event' cluster possessing unique loss distribution characteristics, separate from, say, isolated run-off-road incidents involving specific vehicle types. This capacity for fine-grained differentiation promises not only to enrich actuarial pricing and reserving accuracy but also to significantly enhance the strategic targeting of loss mitigation efforts, such as specialized driver safety training programs or prioritized infrastructure improvement initiatives. In conclusion, this analysis underscores that the NTSB reports constitute an exceptionally valuable source of data for enhancing predictive risk modeling, tarification processes, and reserving studies within actuarial science, particularly when leveraged through the structural and semantic capabilities of a knowledge graph framework. The consistently identified recurrent risk factors—driver impairment, excessive speed, adverse environmental conditions, and critical infrastructural deficiencies—along with the detailed accident characteristics, collectively furnish a robust empirical foundation essential for effectively modeling and ultimately mitigating catastrophic loss events.

4.3 Feasibility Study: Technical Reserving via an NTSB Knowledge Graph

Technical reserving in property-casualty (non-life) insurance consists in estimating the amounts an insurer will ultimately have to disburse to settle claims already incurred. This study investigates whether such *technical reserves* can be estimated by relying solely on a *knowledge graph* built from highway-accident investigation reports published by the U.S. National Transportation Safety Board (NTSB). Fifteen major reports (2021–2022) were analysed to **extract common variables** and evaluate their usefulness for modelling claim reserves.

We first review key reserving concepts—**RBNS**, **IBNR**, and **mathematical reserves** for annuities—to clarify the data normally required. We then identify the **relevant variables extracted from NTSB reports** (loss severity, risk factors, emergency-response delays, etc.) and assess how they might populate a knowledge graph for reserving purposes. Where feasible we outline an **estimation methodology**; otherwise we highlight limitations and indicate which reserve components could only be partially estimated. Finally, should the data prove insufficient for pure reserving, we propose **three alternative actuarial uses** of such a graph—loss prevention, early detection of severe claims, and improvement of claim-report quality—detailing an approach for each.

4.3.1 Recap of Basic Reserving Concepts in Non-Life Insurance

Technical reserves are *funds set aside* to pay for claims that have happened but are not yet fully settled. The main categories are[15, 3]:

- **RBNS (Reported But Not Settled)** — open claims known to the insurer, reserved individually in line with the best estimate of ultimate cost.
- **IBNR (Incurred But Not Reported)** — losses that have occurred but are unknown at the valuation date. IBNR is usually estimated in aggregate from historical reporting lags.
- **Mathematical reserves for annuities** — lifelong annuities following severe bodily-injury claims; the present value of future payments is calculated using mortality tables and discounting in line with statutory rules.

In practice, actuaries require detailed claim histories—occurrence and reporting dates, paid and outstanding amounts, claim descriptors, etc.[14] IBNR often uses *development triangles*; RBNS relies on claim-specific characteristics. Accurate data on accident *severity*, *circumstances*, and *timelines*—the kind we hope to obtain from NTSB reports—can materially improve reserve estimates.

4.3.2 Common Variables Extracted from NTSB Reports

NTSB highway reports supply factual narratives and causal findings for each serious crash. Our analysis identified the following **common variables** suitable for a knowledge graph:

1. *Accident identifiers*: date–time stamp; precise location; infrastructure type.
2. *Environmental conditions*: weather, light level, road surface, lane configuration.
3. *Vehicles and persons involved*: number and types of vehicles; make/model/year; total occupants.
4. *Human toll*: fatalities and injuries (by severity); safety-device usage; qualitative damage.
5. *Circumstances and risk factors*: accident scenario; causal factors such as driver error, DUI, adverse weather, infrastructure failure.
6. *Emergency response*: response time; intervention difficulties; implications for insurance reporting lags.

Graph representation. Each accident becomes a central node linked to nodes for date, location, vehicles, casualties, weather, causes, etc. Example triplets for case `HIR-23-09` (North Las Vegas, 29 Jan 2022):

Accident A – hasLocation – “North Las Vegas intersection”
Accident A – dateTime – “2022-01-29 15:12”
Accident A – involvesVehicle – “2021 Dodge Challenger”
Accident A – fatalities – “9”
Accident A – cause – “Excess speed”
Accident A – cause – “Drug impairment”

4.3.3 Feasibility of Reserve Estimation via the Graph

The graph provides a **rich accident description** but lacks direct financial data. Key usable elements include:

- **Human toll** — primary driver of claim cost; a severity score can map to cost bands.
- **Accident type and vehicle count** — multi-vehicle events imply multiple claim files and potential catastrophe treatment.

- **Risk factors** — causation informs liability and the potential for aggravated damages.
- **Timeline** — response delays and reporting lags affect reserve needs, especially for targeted IBNR.

A possible estimation approach:

1. Structure the graph linking accidents to insured entities.
2. Classify severity via rule-based or model-based scoring calibrated on historical cost data.
3. Use graph inference to detect aggregated events (e.g. 130-vehicle pile-up).
4. Compute reserves via deterministic rules or similarity search against an enriched cost database.
5. Book targeted IBNR for newly detected major accidents not yet reported.

Limitations. NTSB data contain no monetary figures, cover only major crashes, lack policy details, and provide no claim-development history. Hence, while the graph excels at detecting large losses and supporting case-by-case RBNS provisioning, it cannot replace comprehensive actuarial reserving—which still requires full claim histories and financial data.

4.4 Predictive Modeling Application

The NTSB investigation reports contain a wealth of structured information—accidents, contributing factors, recommendations—but when analyzed in isolation, these data remain under-utilized for proactive safety decision-making. The objective of this study is to build a predictive model that, for each investigation (*"case"*), estimates the likelihood that it will generate a safety recommendation. To achieve this, we followed a two-pronged approach:

- **Knowledge Graph (KG) construction**: modeling all entities (cases, factors, recs) and their semantic relationships (cause–effect, recommends, etc.).
- **Case–case similarity graph generation**: extracting shared factors from the KG and linking pairs of cases by computing Jaccard similarity (above a chosen threshold) to create a graph suited to Graph Neural Networks.

This dual strategy—combining the KG's semantic richness with an efficient graph-based learning pipeline—not only captures genuine thematic proximities between investigations but also supports an effective GNN (GCN/GAT) for node-level classification (predicting with or without a recommendation).

4.4.1 Approaches

Traditional predictive analysis on accident reports relies heavily on pure text processing: keyword extraction, TF–IDF vectorization or pretrained embeddings (word2vec, BERT), followed by classifiers such as SVMs or random forests. While these methods can surface broad trends—*e.g.*, terms frequently associated with severe crashes—they struggle to capture the deep structure and logical chains linking causes, contributing factors and safety recommendations. Two reports describing identical issues with different wording will be treated as dissimilar, and textual "noise" undermines the reliability of any similarity measure.

To overcome these shortcomings, we first built a **Knowledge Graph (KG)** that unifies all entities—investigations (`Case`), contributing factors (`Factor`), and safety recommendations (`Rec`)—and their semantic relations (e.g., `Case`→`Factor`, `Rec`→`Case`). This explicit structuring ensures that each edge represents a genuine causal or prescriptive link rather than a mere lexical co-occurrence. Moreover, the KG is fully extensible: one can add new entity types (locations, vehicles, timelines) without changing downstream workflows, and every edge remains traceable for full explainability.

From this enriched KG, we then extracted for each pair of cases their sets of shared factors, computed the Jaccard similarity

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|},$$

and generated a **case–case similarity graph** by retaining only those links whose similarity exceeds a chosen threshold. This second graph—tailored for graph-based learning—avoids noise by connecting only thematically close investigations. It feeds directly into **Graph Neural Networks** (GCN or GAT), which propagate information among neighboring nodes and jointly leverage local attributes (one-hot factor vectors) and global topology.

This two-step *KG → similarity graph* pipeline thus marries the rich semantic backbone of the Knowledge Graph with the effectiveness of GNNs, enabling both node classification (predicting whether a case warrants a recommendation) and link prediction (anticipating missing recommendations).

4.4.2 Knowledge Graph Construction

The first step was to extract from Carol only those NTSB records corresponding to completed highway investigations. Applying the filters "Investigation mode = Highway" and "Status = Completed" yielded the JSON file `cases2025-04-20_19-54.json`. Because the safety-recommendation export in Carol would hang if requested in one pass, we split it into two time-slices—before 1 January 2016 and on or after—and downloaded `safetyrecs2025-04-21_10-19.json` and `safetyrecs2025-04-21_10-21.json` respectively.

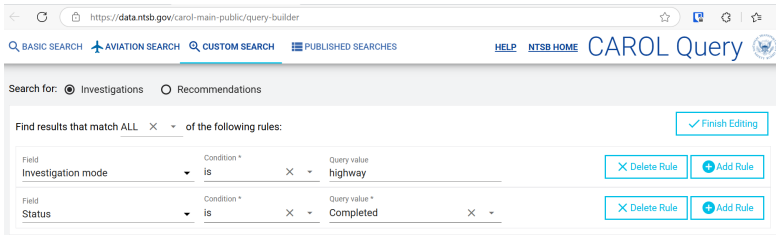

Figure 2: Carol Query Builder – filters applied: *Investigation mode = Highway* and *Status = Completed*.

Next, these two recommendation files were concatenated and sorted by their first notation date to produce `safetyrecs_merged.json`:

```
#!/usr/bin/env python3
import json
from dateutil import parser # pip install python-dateutil

# Load the two Carol exports
with open("safetyrecs2025-04-21_10-21.json", "r", encoding="utf-8") as f:
    recs_new = json.load(f)
with open("safetyrecs2025-04-21_10-19.json", "r", encoding="utf-8") as f:
    recs_old = json.load(f)

# Concatenate and sort by first notation date (newest first)
all_recs = recs_new + recs_old
all_recs.sort(
    key=lambda r: parser.isoparse(r["notations"][0]["dateIssued"]),
    reverse=True
)

# Write the merged file
with open("safetyrecs_merged.json", "w", encoding="utf-8") as f:
    json.dump(all_recs, f, ensure_ascii=False, indent=2)

print("Merged safety recommendations: safetyrecs_merged.json")
```

With both investigations and recommendations in unified JSON form, we constructed a directed NetworkX graph. Each investigation became a node labeled `Case:NTSB_Number` , each finding code a node `Factor:Code` , and each recommendation a node `Rec:SRID` . We also attach to each case node a boolean attribute `hasSafetyRec`, so that downstream processing can directly read whether a recommendation exists. Edges of type `hasFactor` link cases to their factors, and `recommends` edges link recommendations to the cases they address:

```
import json
import networkx as nx

# Load cases and merged recommendations
with open("cases2025-04-20_19-54.json","r",encoding="utf-8") as f:
    cases = json.load(f)
with open("safetyrecs_merged.json","r",encoding="utf-8") as f:
    recs = json.load(f)

KG = nx.DiGraph()

for case in cases:
    cid = case["cm_ntsbNum"]
    # Attach hasSafetyRec right at creation
    KG.add_node(
        cid,
        type="Case",
        hasSafetyRec=bool(case.get("cm_hasSafetyRec", False))
    )
    # Link to factors
    for unit in case.get("cm_vehicles", []):
        for finding in unit.get("cm_findings", []):
            code = finding["cm_findingCode"]
            KG.add_node(f"Factor:{code}", type="Factor")
            KG.add_edge(cid, f"Factor:{code}", relation="hasFactor")
    # Link to recommendations
    if case.get("cm_hasSafetyRec"):
        for r in recs:
            for note in r.get("notations", []):
                if note["mkey"] == case["cm_mkey"]:
                    KG.add_node(f"Rec:{r['srid']}", type="Rec")
                    KG.add_edge(f"Rec:{r['srid']}", cid, relation="recommends")

print(f"KG␣nodes:␣{KG.number_of_nodes()},␣edges:␣{KG.number_of_edges()}")
```

Finally, we visualized and exported the KG using a force-directed layout, coloring cases in blue, factors in green, and recommendations in red. The resulting graph is shown in Figure 3.

```
import matplotlib.pyplot as plt

pos = nx.spring_layout(KG, seed=42, k=0.15, iterations=50)
types = nx.get_node_attributes(KG, "type")
cases = [n for n,t in types.items() if t=="Case"]
factors = [n for n,t in types.items() if t=="Factor"]
recs = [n for n,t in types.items() if t=="Rec"]

plt.figure(figsize=(10,10))
nx.draw_networkx_edges(KG, pos,
    alpha=0.2, edge_color="lightgray", width=0.5)
nx.draw_networkx_nodes(KG, pos, nodelist=cases, node_color="tab:blue", node_size=20, label="
    Case")
nx.draw_networkx_nodes(KG, pos, nodelist=factors, node_color="tab:green", node_size=10,
    label="Factor")
nx.draw_networkx_nodes(KG, pos, nodelist=recs, node_color="tab:red", node_size=40, label="
    Rec")
```

```
plt.legend(scatterpoints=1, fontsize=12)
plt.axis("off"); plt.tight_layout()
plt.savefig("ntsb_kg.png", dpi=200)
```

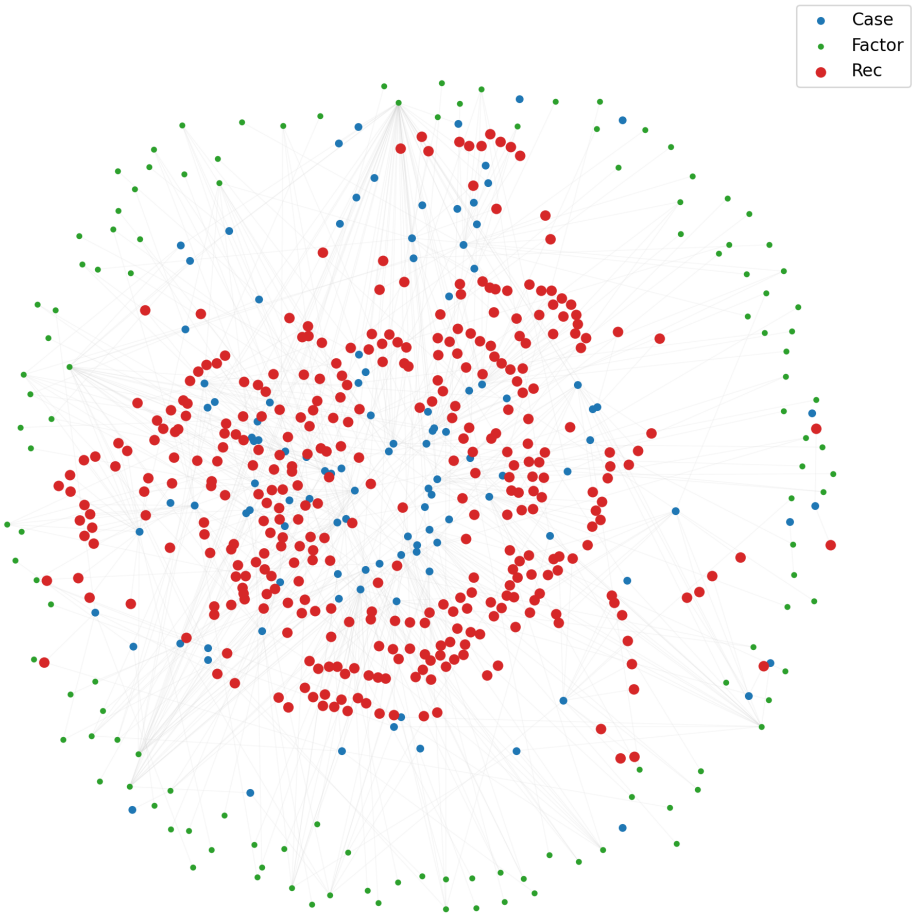

Figure 3: Knowledge Graph of NTSB highway investigations, showing `Case`, `Factor`, and `Rec` nodes.

4.4.3 Similarity Graph Construction and Predictive Modeling

Building upon the Knowledge Graph (KG) described in Section 4.4.2, we derive a *case–case similarity graph* tailored for graph-based learning. The conversion proceeds by first isolating each accident report ("Case") in the KG and gathering its set of contributory factors (`cm_findingCode`). A pairwise Jaccard similarity between these factor sets is computed for every duo of reports. When the similarity exceeds a chosen threshold (e.g. 0.3), we connect the two cases with bidirectional edges, thereby producing an undirected similarity graph in which each node denotes a Highway investigation and each edge signifies substantial thematic overlap.

Listing 1: Extracting case–case edges from the KG

```
import networkx as nx
import torch
import itertools
from torch_geometric.data import Data

# 1) Load the fully attributed KG
KG = nx.read_gexf("ntsb_kg.gexf")

# 2) Extract Case nodes and their Factor code sets
```

```
case_nodes = [n for n,d in KG.nodes(data=True) if d["type"]=="Case"]
case_to_factors = {
    case: {
      nbr.split("Factor:")[1]
      for nbr in KG.successors(case)
      if nbr.startswith("Factor:")
    }
    for case in case_nodes
}

# 3) Compute Jaccard similarities and assemble bidirectional edges
threshold = 0.3
edges = []
for u, v in itertools.combinations(case_nodes, 2):
    A, B = case_to_factors[u], case_to_factors[v]
    if A and B and len(A & B) / len(A | B) >= threshold:
        edges += [(u, v), (v, u)]

# 4) Map string IDs to integer indices and build edge_index for PyG
mapping = {c: i for i, c in enumerate(case_nodes)}
edge_index = torch.tensor([
    [mapping[u] for u,_ in edges],
    [mapping[v] for _,v in edges]
], dtype=torch.long)
```

From this process emerges a similarity graph with one node per completed Highway report and edges encoding substantive overlaps in reported factors.

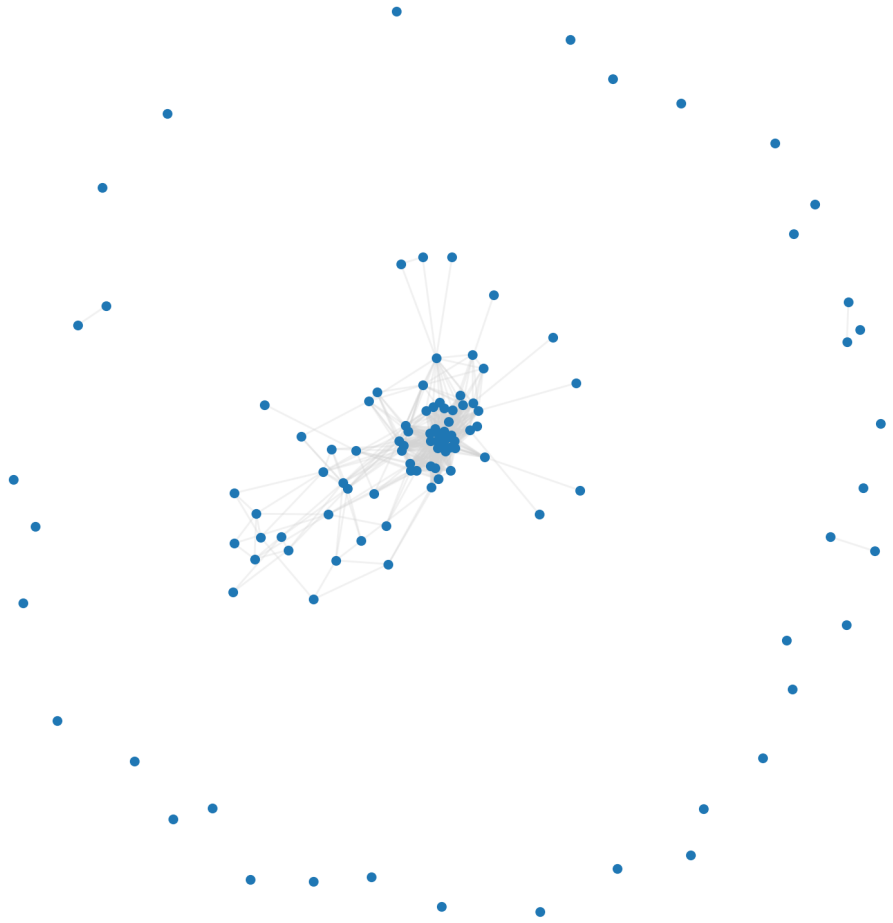

Figure 4: Case–case similarity graph for completed NTSB highway investigations. Nodes represent reports, edges indicate Jaccard similarity 0.3 on shared factors.

Next, each node is parameterized by a binary feature vector indicating the presence of every possible factor, and is labeled according to whether a recommendation exists on the original KG.

Listing 2: Building feature matrix `X` and label vector `y`

```
# 5) Index the complete set of Factor codes
all_codes = sorted({c for codes in case_to_factors.values() for c in codes})
code2idx = {c: i for i, c in enumerate(all_codes)}
N, C = len(case_nodes), len(all_codes)

# 6) Construct the feature matrix X (onehot encoding of factors)
X = torch.zeros((N, C), dtype=torch.float)
for case, idx in mapping.items():
    for code in case_to_factors[case]:
        X[idx, code2idx[code]] = 1.0

# 7) Build the binary label vector y from KG attribute hasSafetyRec
y = torch.tensor([
    int(KG.nodes[case]["hasSafetyRec"])
    for case in case_nodes
], dtype=torch.long)

# 8) Assemble the final PyG Data object
data = Data(x=X, edge_index=edge_index, y=y)
print(data)
```

With `data` in hand—where `x` contains factor-based features, `edge_index` defines the graph topology, and `y` holds the targets—we can proceed to any standard GNN training pipeline (e.g. GCN/GAT, train/val/test split, early stopping, accuracy/F1 monitoring) without further modification.

4.4.4 Training the GCN Model and Experimental Results

Building on the case–case similarity graph described in Section 4.4.2, we proceed to train a two-layer Graph Convolutional Network (GCN) in order to predict, for each investigation report node, whether it has an associated safety recommendation. To begin, the set of nodes is split into training, validation, and test subsets in a stratified manner so as to preserve the same proportion of positive (has recommendation) and negative examples across all splits. First, 30% of the data is held out as the test set, and then 20% of the remainder is used for validation. The resulting indices are used to build boolean masks stored in the `data` object of PyTorch Geometric:

Listing 3: Stratified splitting and mask creation

```
import torch
from sklearn.model_selection import StratifiedShuffleSplit

num_nodes = data.num_nodes
all_idx = torch.arange(num_nodes)

# First split: train+val vs. test (30%)
sss1 = StratifiedShuffleSplit(n_splits=1, test_size=0.3, random_state=42)
train_val_idx, test_idx = next(sss1.split(all_idx, data.y.numpy()))

# Second split: train vs. val (20% of train+val)
sss2 = StratifiedShuffleSplit(n_splits=1, test_size=0.2, random_state=42)
train_idx, val_idx = next(sss2.split(train_val_idx, data.y.numpy()[train_val_idx]))

# Create boolean masks
data.train_mask = torch.zeros(num_nodes, dtype=torch.bool)
data.val_mask = torch.zeros(num_nodes, dtype=torch.bool)
data.test_mask = torch.zeros(num_nodes, dtype=torch.bool)
data.train_mask[train_idx] = True
data.val_mask[val_idx] = True
data.test_mask[test_idx] = True
```

The GCN itself consists of two graph convolution layers with ReLU activations and a dropout of 50% to mitigate overfitting. We employ the `GCNConv` layer from PyTorch Geometric and optimize with Adam including L2 regularization:

Listing 4: Definition of the two-layer GCN model

```
import torch.nn.functional as F
from torch_geometric.nn import GCNConv

class GCN(torch.nn.Module):
    def __init__(self, in_channels, hidden_channels, out_channels, dropout=0.5):
        super().__init__()
        self.conv1 = GCNConv(in_channels, hidden_channels)
        self.conv2 = GCNConv(hidden_channels, out_channels)
        self.dropout = dropout

    def forward(self, x, edge_index):
        x = self.conv1(x, edge_index)
        x = F.relu(x)
        x = F.dropout(x, p=self.dropout, training=self.training)
        x = self.conv2(x, edge_index)
        return x

model = GCN(
```

```
    in_channels = data.num_node_features,
    hidden_channels = 16,
    out_channels = int(data.y.max().item()) + 1,
    dropout = 0.5
)
optimizer = torch.optim.Adam(model.parameters(), lr=0.01, weight_decay=5e-4)
criterion = torch.nn.CrossEntropyLoss()
```

Training proceeds for up to 200 epochs with an early-stopping criterion: if validation accuracy does not improve for 10 consecutive epochs, training halts and the best model state is restored. At each epoch, the cross-entropy loss on the training mask is minimized, and accuracy is monitored on the validation mask:

Listing 5: Training loop with early stopping

```
@torch.no_grad()
def eval_accuracy(mask):
    model.eval()
    out = model(data.x, data.edge_index)
    preds = out.argmax(dim=1)
    return (preds[mask] == data.y[mask]).float().mean().item()

best_val_acc = 0.0
patience, wait = 10, 0

for epoch in range(1, 201):
    model.train()
    optimizer.zero_grad()
    out = model(data.x, data.edge_index)
    loss = criterion(out[data.train_mask], data.y[data.train_mask])
    loss.backward()
    optimizer.step()

    val_acc = eval_accuracy(data.val_mask)
    if val_acc > best_val_acc + 1e-4:
        best_val_acc = val_acc
        best_state = model.state_dict()
        wait = 0
    else:
        wait += 1
        if wait >= patience:
            print(f"Early␣stopping␣at␣epoch␣{epoch}")
            break

    if epoch % 10 == 0:
        train_acc = eval_accuracy(data.train_mask)
        print(
            f"Epoch␣{epoch:03d}␣|␣"
            f"Loss:␣{loss:.4f}␣|␣"
            f"Train␣Acc:␣{train_acc:.4f}␣|␣"
            f"Val␣Acc:␣{val_acc:.4f}"
        )
```

After training, we load the best model parameters and evaluate on the held-out test set. The final performance metrics are:

```
Epoch 010 | Loss: 0.5913 | Train Acc: 0.8254 | Val Acc: 0.6875
Early stopping at epoch 14
 Test Accuracy: 0.7647
```

These results—approximately 76% accuracy on unseen reports—demonstrate that our two-stage

pipeline (knowledge graph to similarity graph, followed by GCN learning) effectively captures the latent relationships that govern the presence of safety recommendations.

4.4.5 Future Perspectives

In this final section, we reflect on the contributions of our two-stage methodology and outline possible avenues for improvement and extension. We first discuss the joint benefits of linking a rich semantic Knowledge Graph (KG) to a streamlined similarity graph designed for machine learning. We then identify the principal limitations encountered, and finally propose directions for future scientific and operational development.

Benefits of Dual Modeling The two-step approach—first constructing a semantic Knowledge Graph to capture the full richness of NTSB reports, then projecting it into a lean *case–case similarity graph* for graph-based learning—yields clear advantages. The KG preserves all domain relationships (cause–effect chains, recommendations, geographic and vehicle context, etc.), ensuring that similarity between cases reflects genuine thematic proximity rather than mere statistical correlation. Meanwhile, the resulting similarity graph, with its reduced average degree and undirected structure, is ideally suited to Graph Neural Networks, limiting noise and promoting rapid, reliable convergence.

Identified Limitations Despite these strengths, several constraints merit attention. The modest size of our Highway subset (on the order of a few hundred nodes) inherently restricts model generalization and inflates evaluation variance. The reliability of our pipeline also hinges on the consistency of factor codes and recommendation notations exported from the Carol platform, which can sometimes exhibit duplication or missing values. Finally, by symmetrizing edges during similarity graph construction, we sacrifice the directional semantics (e.g., `Rec→Case`) that are natively represented in the KG.

Directions for Extension Multiple promising directions can build on this work. First, one could apply link-prediction algorithms directly on the KG to infer and suggest missing recommendations or latent factor associations. Second, enriching the graph with temporal and spatial attributes—such as event dates and GPS coordinates—would enable seasonality- or region-specific analyses and improve similarity measures. Third, implementing an incremental learning pipeline, in which new reports are continuously ingested into the KG, the similarity graph periodically recomputed, and the GNN retrained, would support real-time prevention applications and maintain model relevance as the corpus grows.

This chapter thus sketches the contours of a research agenda at the intersection of graph mining and predictive analytics, offering a robust and adaptable framework for safety and prevention studies.

5 Conclusion

This study explored the integration of knowledge graphs into actuarial science as a means of advancing traditional risk modeling, reserving, and prevention strategies. Beginning with a detailed review of causality and causal graph theory, we demonstrated how these tools allow actuaries to move beyond correlation-based assessments and toward richer, causally-informed models. The use of Bayesian networks and causal inference frameworks such as CausalKG [9] was highlighted as pivotal in enabling predictive and counterfactual reasoning, providing not only greater explanatory power but also decision-making support in complex, uncertain environments [6, 14].

Building upon this theoretical foundation, we presented a rigorous modeling pipeline for generating causal knowledge graphs from raw textual data. Through careful data acquisition, pre-processing, semantic normalization, and the strategic integration of large language models [12, 11], we demonstrated how structured causal representations can be extracted and scaled efficiently across document corpora. The ability to perform this transformation at scale, while maintaining semantic integrity, is a critical innovation in enabling dynamic, data-rich actuarial applications [1].

Our case study on NTSB accident reports validated the practical applicability of this framework. The detailed event narratives provided in these reports allowed us to build knowledge graphs that encoded complex interrelations among vehicles, causes, consequences, and contextual variables. While we showed that the NTSB data was not sufficient to support complete technical reserving due to the lack of financial indicators and insurance policy metadata, we identified high-potential proxies—such as accident severity and causal attributions—that could aid in estimating reserves for major incidents and in detecting targeted IBNR events. This underscores the value of knowledge graphs for augmenting traditional actuarial models with deeper contextual insights, particularly in the face of high-severity but low-frequency events.

Beyond reserving, our prevention-focused modeling demonstrated how a knowledge graph and similarity graph hybrid can support predictive safety recommendation systems using graph neural networks (GNNs). The ability to automatically anticipate future safety actions from past case similarities reveals a new paradigm for proactive risk management, which is central to the evolving responsibilities of actuaries in modern enterprises.

In conclusion, while knowledge graphs may not yet replace foundational actuarial tools, they represent a robust and flexible extension to the actuarial toolbox. Their capacity to unify structured and unstructured data, support causal reasoning, and enable advanced analytics opens new frontiers for more granular, responsive, and intelligent actuarial modeling. Future research should aim to integrate real-time insurer data into these graphs, refine reserve estimation models via supervised learning, and further explore the synergy between semantic graph frameworks and emerging neural techniques. Through such work, knowledge graphs are poised to contribute meaningfully to a new, more holistic actuarial paradigm.

References

[1] A. Brack, A. Hoppe, M. Stocker, S. Auer, and R. Ewerth. Analysing the requirements for an open research knowledge graph: Use cases, quality requirements, and construction strategies. *International Journal on Digital Libraries*, 23(1):33–55, 2022. This article identifies the user requirements for open research knowledge graphs and discusses strategies for their construction, highlighting issues of completeness, precision, and quality assurance.

[2] A. Cho, G. C. Kim, A. Karpekov, A. Helbling, Z. J. Wang, S. Lee, et al. Transformer explainer: Interactive learning of text-generative models. `https://arxiv.org/abs/2408.04619`, 2024. arXiv preprint arXiv:2408.04619. This paper develops an interactive tool to visualize Transformer models, enhancing the understanding of attention mechanisms with practical examples from GPT-2.

[3] Institut des actuaires. Guide des bonnes pratiques de provisionnement des sinistres en assurance non-vie. Technical report, Institut des actuaires, 2023. This guide details deterministic and stochastic methods for estimating reserves in non-life insurance, focusing on both reported and incurred but not reported (IBNR) claims, and emphasizing data quality and regulatory compliance.

[4] PwC Middle East. Graph llms: The next ai frontier in banking and insurance transformation. `https://www.pwc.com/m1/en/publications/documents/2024/graph-llms-the-next-ai-frontier-in-banking-and-insurance-transformation.pdf`, 2024. This report examines the integration of Graph Neural Networks (GNNs) with large language models in the banking and insurance sectors, highlighting real-world use cases such as fraud detection and dynamic pricing.

[5] L. Feddoul, F. Löffler, and S. Schindler. A systematic literature review and classification of approaches for keyword search over graph-shaped data. Technical report, Heinz Nixdorf Chair for Distributed Information Systems, Friedrich Schiller University Jena, 2022. The authors review keyword search methods over graph databases, outlining existing limitations and suggesting avenues for future innovations in semantic information processing.

[6] J. Gamonet and S. Le Sujet. Modélisation du risque opérationnel dans l'assurance. Mémoire d'actuaire, CEA, 2009. This report explores methods for operational risk modeling in insurance, focusing on Bayesian networks to capture relationships between rare events and potential financial losses.

[7] M. Iantosca. The role of knowledge graphs in rag solutions. Medium, November 2024. This article explains how knowledge graphs enhance retrieval-augmented generation (RAG) solutions by explicitly modeling entity relationships, which improves answer accuracy and traceability.

[8] InsurAnalytics.ai. Deploying deep learning in claims reserving. Medium, March 2019. This article illustrates how deep learning can enhance claims reserving in property and casualty insurance, offering gains in precision and automation.

[9] U. Jaimini and A. Sheth. Causalkg: Causal knowledge graph explainability using interventional and counterfactual reasoning. *IEEE Internet Computing*, 26(1):43–50, 2022. This article introduces CausalKG, a framework that enriches knowledge graphs with complex causal relations via interventional and counterfactual reasoning, improving the explainability of AI models in sensitive domains.

[10] M. Knight. What is a knowledge graph? DATAVERSITY, February 2025. This article defines knowledge graphs as structures composed of nodes, edges, and properties to represent real-world entities and their interrelations, discussing their inferential and scalable capabilities.

[11] R. McDermott. From unstructured text to interactive knowledge graphs using llms. Medium, March 2025. This piece presents a complete pipeline for constructing knowledge graphs from unstructured text using large language models, including segmentation, extraction, normalization, and interactive visualization.

[12] S. Minaee, T. Mikolov, N. Nikzad, M. Chenaghlu, R. Socher, X. Amatriain, and J. Gao. Large language models: A survey. `https://arxiv.org/abs/2402.06196`, 2024. This survey provides a comprehensive overview of large language models, covering architectures, training data, performance benchmarks, and the associated technical and ethical challenges.

[13] Munich Re. Large language models in underwriting and claims. Munich Re, March 2024. This report discusses the application of large language models for streamlining underwriting and claims processing through automated extraction of relevant data from unstructured documents.

[14] D. H. Reid. Claim reserves in general insurance. *Journal of the Institute of Actuaries*, 105(3):211–315, 1978. This seminal article discusses the technical and accounting foundations of claim reserving in general insurance, addressing uncertainties in claim occurrence, reporting, and development, and establishing methodologies still used in modern actuarial practice.

[15] A. Saoudi, F. El Kassimi, and J. Zahi. Technical reserving in non-life insurance: A literature review of aggregated and individual methods. *Journal of Integrated Studies in Economics, Law, Technical Sciences & Communication*, 1(2), 2023. This review contrasts aggregated methods with individual approaches in non-life insurance reserving and advocates for more precise individual methodologies.

[16] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017. This foundational paper introduces the Transformer architecture, which relies solely on attention mechanisms, revolutionizing neural network models and laying the groundwork for modern LLMs like GPT and BERT.

[17] J. Yao, W. Sun, Z. Jian, Q. Wu, and X. Wang. Effective knowledge graph embeddings based on multidirectional semantics relations for polypharmacy side effects prediction. *Bioinformatics*, 38(8):2315–2322, 2022. This paper proposes an embedding model for knowledge graphs that

leverages multidirectional semantic relations to improve the prediction of polypharmacy side effects, reducing parameter counts and enhancing performance.

Appendix

A.1 Timeline and Milestones

Period	Milestone	Key Outcomes
05 Mar – 08 Mar	Kick-off meeting	Project scope defined; weekly cadence set; Notion workspace created.
09 Mar – 15 Mar	Literature review	Bibliography consolidated; six-step pipeline sketched; initial role allocation.
16 Mar – 31 Mar	NTSB exploration	Manual factor extraction; switch to JSON export; prototype pipeline tested end-to-end; figuring out which type of Actuarial study is the better.
01 Apr – 11 Apr	Modeling & reporting	reserving& prevention applications; switching from preventions to predictive modeling; GCN training (76)
11 Apr – 25 Apr	Finalization	Final steps of creating the Graph, exploring it. and writing the report

A.2 Task Allocation and Roles

- **K. Mardochee** – Lead on documentation, literature review and the reserving feasibility study; implemented RBNS/IBNR proxy rules and limitations analysis.
- **A. Dibi** – Lead on predictive-model branch; built case–case similarity graph and trained GCN classifier; produced performance diagnostics.
- **I. Mohamed El Hafed** – Lead author for theoretical sections (Sections 2 3); maintained bibliography and overall LaTeX integration.
- All members jointly did the necessary documentation in the beginning, performed manual NTSB exploration, shared Notion note-taking, and attended every weekly meeting.

A.3 Collaboration Tools and Workflow

- **Notion** – Central knowledge base (papers, meeting minutes, task Kanban).
- **Overleaf** – Version control for Python notebooks and LaTeX source.
- **Python / Jupyter** – Data extraction, clustering, GCN training (PyTorch Geometric).
- **Weekly video call** – 60-minute stand-up with agenda and action items; ad-hoc Slack channel for daily sync.

A.4 Interaction with Supervisor

- Weekly half-hour meeting supplemented by contact through email to check the advancement and request advice.
- Supervisor validated the research plan, suggested additional literature on knowledge graphs in risk contexts, and provided feedback on methodological pivots
- checkpoint, used to realign scope toward predictive modeling after reserving proven data-limited.
- Final draft reviewed before submitting the report

Overall, the structured workflow, clear role definition, and continuous supervisor engagement were instrumental in delivering the TER on time and in scope.