electronics

MDPI

Article

Information Extraction of Aviation Accident Causation Knowledge Graph: An LLM-Based Approach

Lu Chen [1], Jihui Xu [1,*], Tianyu Wu [2] and Jie Liu [2]

[1] Equipment Management and Unmanned Aerial Vehicle Engineering School, Air Force Engineering University, Xi'an 710051, China; chen_lu2@163.com
[2] Data Science and Intelligence Analysis Laboratory, Beijing Information Science and Technology University, Beijing 100092, China; bistuwty@gmail.com (T.W.); zztong_181012@163.com (J.L.)
* Correspondence: skyline6018@126.com

Abstract: Summarizing the causation of aviation accidents is conducive to enhancing aviation safety. The knowledge graph of aviation accident causation, constructed based on aviation accident reports, can assist in analyzing the causes of aviation accidents. With the continuous development of artificial intelligence technology, leveraging large language models for information extraction and knowledge graph construction has demonstrated significant advantages. This paper proposes an information extraction method for aviation accident causation based on Claude-prompt, which relies on the large-scale pre-trained language model Claude 3.5. Through prompt engineering, combined with a few-shot learning strategy and a self-judgment mechanism, this method achieves automatic extraction of accident-cause entities and their relationships. Experimental results indicate that this approach effectively improves the accuracy of information extraction, overcoming the limitations of traditional methods in terms of accuracy and efficiency in processing complex texts. It provides strong support for subsequently constructing a structured knowledge graph of aviation accident causation and conducting causation analysis of aviation accidents.

Keywords: knowledge graphs; large language models (LLMs); information extraction; aviation safety; accident causation

check for updates

Citation: Chen, L.; Xu, J.; Wu, T.; Liu, J. Information Extraction of Aviation Accident Causation Knowledge Graph: An LLM-Based Approach. *Electronics* **2024**, *13*, 3936. https://doi.org/10.3390/electronics13193936

Academic Editor: George Angelos Papadopoulos

Received: 4 September 2024
Revised: 1 October 2024
Accepted: 2 October 2024
Published: 5 October 2024

1. Introduction

Aviation activities, as a mode of transportation characterized by high speed, efficiency, and frequency, are increasingly prevalent, playing a significant role in social life [1]. Although the probability of aviation accidents is low, once they occur, they can result in significant casualties and property losses, thereby exerting substantial negative societal impacts. Given the escalating number of aviation accidents and the severity of their consequences, ensuring the safety of aviation activities is of utmost importance. Consequently, the issue of how to reduce the occurrence probability of aviation accidents has garnered increasing attention and emphasis [2].

Causation analyses based on extensive accident data are an effective method for accident prevention [3]. Aviation accident reports serve as a valuable source for accident causation analyses, enabling the effective identification of potential aviation safety threats and preventing accidents from occurring [4]. However, aviation accident reports are characterized by their large volume, rapid increment, and low utilization rate [5]. Therefore, it becomes particularly urgent to employ intelligent means for the efficient mining and structured storage of accident information, objectives which facilitate the exploration of inherent patterns behind accident occurrences [6]. Furthermore, with the development of artificial intelligence methods, technologies such as knowledge graphs and large language models (LLMs) are bringing about new opportunities for research in aviation safety [7]. Knowledge graphs can structurally represent and connect the unstructured and complex texts found in aviation accident reports [8], extract causal factors, and form a network of causal relationships, thereby expressing information on complex accident causation [9]. This allows for a more efficient and comprehensive exploration of inherent correlations [10], enabling structured storage and visualization, and providing a solid data foundation for subsequent analyses of aviation accident causation. Furthermore, in the context of big data, there is still a lack of systematic application frameworks in the field of safety [11]. Establishing a knowledge graph of aviation accidents is conducive to developing an application framework for aviation safety.

Currently, research on knowledge graphs in the field of aviation safety is still in its developmental phase, primarily focusing on aspects, such as knowledge graph construction [12], information extraction [4,13], and the application of knowledge graphs [14]. Among these, given that the essence of a knowledge graph is the process of effectively organizing and intelligently processing vast amounts of information, the core of constructing a knowledge graph lies in information extraction [15]. However, aviation accident causes are often presented in long texts, resulting in a large number of extracted entities and complex logical relationships between them. Traditional information extraction methods mainly rely on manual analyses or supervised learning [4,16]. When dealing with unstructured and complex long-text information, a manual analysis is inefficient [17], costly [18], susceptible to subjective interference [19], and challenging for large-scale sample mining. Supervised learning models, on the other hand, depend on high-quality annotated data and have limited generalization capabilities. The development of large model technologies has provided new possibilities for information extraction in knowledge graphs. LLMs can automatically extract information from unstructured texts, decreasing the workload of manual labeling and improving extraction efficiency. They are gradually becoming a powerful tool for constructing knowledge graphs [20,21].

Therefore, addressing the current challenges in the construction of aviation accident causation knowledge graphs (AACKG), including low efficiency in information extraction, dependence on high-quality annotated data, and the need for improved extraction accuracy, this paper collects 5364 aviation accident reports since 2000 to construct an ontology in the domain of aviation accident causes. We propose a Claude-prompt method for extracting aviation accident causation information, leveraging the powerful natural language understanding capabilities of LLMs to extract textual descriptions of accident causes from aviation accident reports, thereby obtaining accident causation entities and their relationships. This method is then compared with traditional supervised methods and existing LLM-based methods. Experimental results demonstrate significant improvements in precision, recall, and F1-score with our proposed method. Compared to supervised learning methods, LLMs exhibit strong generalization capabilities, enabling them to handle complex and variable aviation accident reports while reducing dependence on annotated data and, consequently, lowering the cost of data annotation. Compared to existing LLM-based methods, the few-shot prompting strategy and self-judgment mechanism proposed in this paper effectively enhance model performance. The structure of the subsequent sections is as follows: In Section 2, we introduce knowledge graph construction, information extraction methods, and LLM-based methods, describing the preliminary work such as data collection and ontology construction for the knowledge graph. In Section 3, experiments are carried out utilizing 5364 global aviation accident investigation reports spanning from 2000 to 2023 as the foundational dataset. Information extraction of accident causation is performed using the Claude-prompt method. A comparison is made between the effectiveness of this approach and that of traditional information extraction methods for knowledge graphs, as well as large model-based methods. The results lead to the conclusion that the Claude-prompt method exhibits higher efficiency and greater accuracy. This lays the groundwork for subsequent analysis of aviation accidents and the establishment of a data-driven accident prevention system. Section 4 discusses the advantages and disadvantages of the proposed method as well as the subsequent work, while Section 5 provides a summary of the research presented in this paper.

2. Materials and Methods

2.1. Construction of AACKG

2.1.1. Construction Process of AACKG

The Knowledge Graph is a structured form of knowledge representation composed of nodes and edges, which stores knowledge in the form of triples ("Entity-Relationship-Entity") [22]. As an aeronautical domain-specific knowledge graph, AACKG possesses characteristics such as specialization and dynamic evolution [23]. The construction of AACKG enables the transformation of textual descriptions of aircraft accident causes into a connection of entities and relationships. Through the semantic graph of nodes and edges, it forms a comprehensive, structured, dynamically updated, and easily accessible knowledge base [8], which facilitates the subsequent structured management of aircraft accident data [10]. This allows for a more in-depth exploration of the potential relationships and inherent laws between accident causes.

Figure 1 illustrates the fundamental process of constructing and applying a domain-specific knowledge graph, which is primarily divided into six steps: data collection, ontology construction, information extraction, knowledge fusion, knowledge storage and visualization, and knowledge graph application. Among these, the ontology construction primarily focuses on categorizing the factors causing aircraft accidents. Information extraction is the core of knowledge graph construction, mainly involving the extraction of entities and relationships from text-based data. The knowledge storage module is responsible for storing the extracted knowledge triples and facilitating their visualization. This paper primarily discusses the methods of information extraction in the process of knowledge graph construction; therefore, it is necessary to first complete the steps of data acquisition and ontology construction.

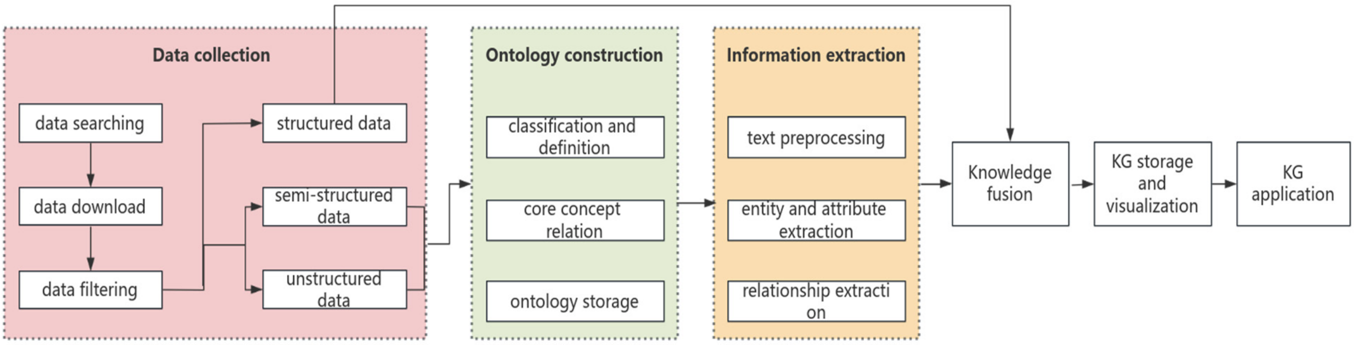


Figure 1. Process of knowledge graph building and application.

2.1.2. Data Collection

Accident investigation reports represent the most direct source of data for accident analyses. Currently, there are two primary sources of such data globally. The first category originates from governmental agencies, aviation accident investigation boards, and civil aviation authorities of various countries, which are responsible for publishing official aviation accident investigation reports. Examples include the National Transportation Safety Board (NTSB) of the United States of America, the United Kingdom Accident Investigation Board (UKAB), the Japan Transport Safety Board (JTSB), and the German Federal Bureau of Aircraft Accident Investigation (BFU) of Germany. The second category consists of online aviation accident databases, such as the NTSB Aviation Accident Database and the Aviation Safety Network (ASN), which comprehensively compile aviation accident data from around the world. These databases derive their data primarily from governmental agencies and news sources.

This paper selects the database of ASN, an aviation accident data statistics website affiliated with the Flight Safety Foundation, which provides comprehensive information on aircraft accidents and records descriptions of aviation accidents involving various aircraft types such as airliners, military transport category aircraft, and corporate jet aircraft (capable of carrying at least 12 passengers) since 1919 [24]. The ASN is chosen as the core data source for this paper primarily because the accident report data entered on this website is relatively comprehensive and updated in a timely manner. The database is primarily structured and semi-structured, facilitating easy collection, analysis, and management.

This paper selects a total of 5364 aviation accident reports from the years 2000 to 2022 on the ASN as sample data, a method that has been employed in data collection in references [25,26]. The data collection process is illustrated in Figure 2. Initially, the ASN is identified as the primary source for data acquisition. Subsequently, data are collected using Octopus and stored in Excel, yielding 6062 accident records. These records include structured data such as time, location, nature of flight, flight phase, and accident consequences, as well as unstructured data such as accident narratives, causes, and types. Then, to address issues such as missing, duplicate, erroneous, or anomalous data encountered during the collection process—including duplicate entries, special characters, or garbled text—the data are cleaned, refined, and augmented by cross-referencing it with original data published by various national aviation investigation authorities. This step enhances the overall quality and integrity of the data. Among the 6062 collected records, an assessment is conducted on the quality of the descriptions of the accident causes, and entries with overly short texts or those that do not mention causal information are removed. The primary reason for this is that, through testing, it is found that Probable Cause texts lacking causal information are more prone to Hallucination in the analysis of LLMs. Ultimately, 5364 data records are obtained, forming an aviation accident database stored in Excel. This database includes structured data fields such as accident time, location, accident type, accident consequences (including casualties and aircraft damage), nature of flight, flight phase, and unstructured data fields such as accident descriptions, causes, consequences, and improvement measures. These serve as the data input for subsequent information extraction tasks.

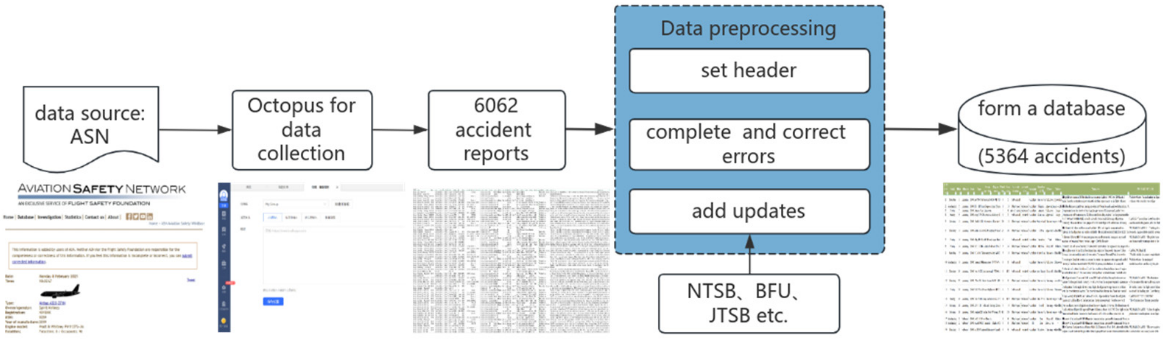


Figure 2. Process of accident data collection.

The research content of this paper focuses on the information extraction of aviation accident causes, with relevant data solely contained within the two fields of "narrative" and "probable cause". These two fields consist of unstructured and complex textual data, primarily ranging from 200 to 1000 words, and mainly describe information such as accident causes, types, and consequences. Table 1 presents the specific content regarding accident description and causes from report number 5021.

Table 1. Sample of accident narrative and probable cause.

Number	Narrative	Probable Cause
5021	A Challenger 604 corporate jet sustained damage during a crosswind landing on runway 22 at London-Stansted Airport, UK. Control was lost during an attempted landing in a strong crosswind, following an ILS approach. The left wingtip struck the runway several times and remained in contact with the ground as the aircraft departed the paved surface into the grass area at the side of the runway. The aircraft's stick pusher also activated, resulting in a hard landing that damaged the aircraft's nose landing gear assembly. The flight then diverted to London-Gatwick Airport, where it landed at 00:51 UTC.	Conclusion The aircraft yawed and rolled rapidly following a long float with insufficient airspeed in strong gusting wind conditions. Stick shaker activation was followed almost immediately by stick pusher activation, resulting in the aircraft landing on its nosewheel. A crosswind exceeding the commander's personal limit was forecast before departure. It would have been possible to delay departure or select an alternative arrival aerodrome with more favorable conditions. The commander reflected that although there was an opportunity to discontinue the approach earlier, he had felt compelled to continue with the landing by a degree of plan continuation bias. Fatigue, commercial pressure, and the nature of their interactions may have made the pilots more susceptible to this bias.

2.1.3. Ontology Construction

An ontology serves as the foundational framework for constructing a knowledge system. It defines an abstract model encompassing entities, attributes, relationships, and their constraints within a specific domain. By describing the structure of domain knowledge and the hierarchical relationships between concepts through a formalized language, it enables computers to comprehend and reason about complex information in the real world. Commonly employed methods for ontology construction include manual, automatic, and semi-automatic approaches. This paper adopts a manual method to construct the ontology based on the characteristics of accident-cause entities and the HFACS model.

The ontology construction process is illustrated in Figure 3. Step 1: establish the preliminary framework of the aviation accident causation ontology based on data characteristics. Firstly, given that the "Human-Aircraft-Environment-Management" (HAEM) framework serves as the prevailing and authoritative classification approach [27], it is selected as the primary classification level for the aviation accident causation ontology. However, the granularity of the HAEM classification is too coarse to facilitate in-depth exploration of accident causes. Consequently, further subdivision and expansion are necessary within the HAEM framework. Secondly, the Human Factors Analysis and Classification System (HFACS) model is adopted as the basis for the second-level classification, integrating the elements of HFACS into the HAEM dimensions. Thirdly, based on the HFACS classification framework, some elements are optimized and adjusted, and a third-level classification is applied to certain elements. Ultimately, the preliminary framework of the aviation accident causation ontology is completed. Step 2: convene an expert review meeting based on the established accident ontology framework. Five experts in the field of aviation safety are invited to discuss the framework, adhering to the classification principles of no overlap, no omission, and highlighting research priorities. Based on the feedback from aviation safety experts, the framework is refined and modified accordingly. Step 3: annotate aviation accident data with triple instances. When identified annotation instances do not correspond to the existing ontology, further improvements and supplements are made to the ontology. This iterative process ensures that the ontology evolves to comprehensively encompass all relevant aspects of aviation accidents. Ultimately, the finalized aviation accident ontology is established.

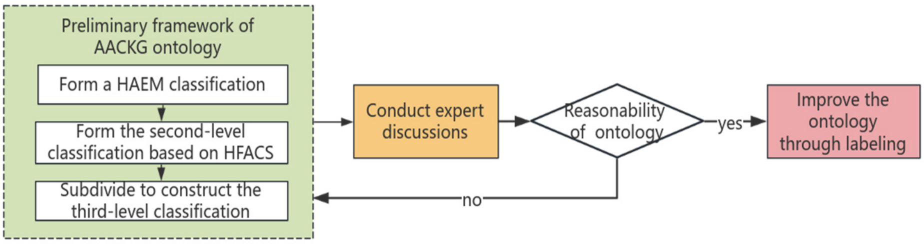

Preliminary framework of AACKG ontology
Form a HAEM classification
Form the second-level classification based on HFACS
Subdivide to construct the third-level classification
Conduct expert discussions
Reasonability of ontology
yes
no
Improve the ontology through labeling

Figure 3. Construction of AACKG ontology.

Figure 4 shows the ontology concepts of the aviation accident causation, which consists of 14, level 1 concepts, and 30, level 2 concepts. When constructing the ontology for the AACKG, level 2 factors were selected; however, considering that aircraft factors are only at level 1 and the importance and frequency of occurrence of artificial and social environments among environmental factors are relatively low, only level 1 was selected for these. Consequently, a total of 27 core concepts emerged as the ontology, as indicated by the orange area in Figure 4. Additionally, this paper provides definitions and codes for the core concepts in the ontology. Due to space constraints, these are presented in Appendix A.

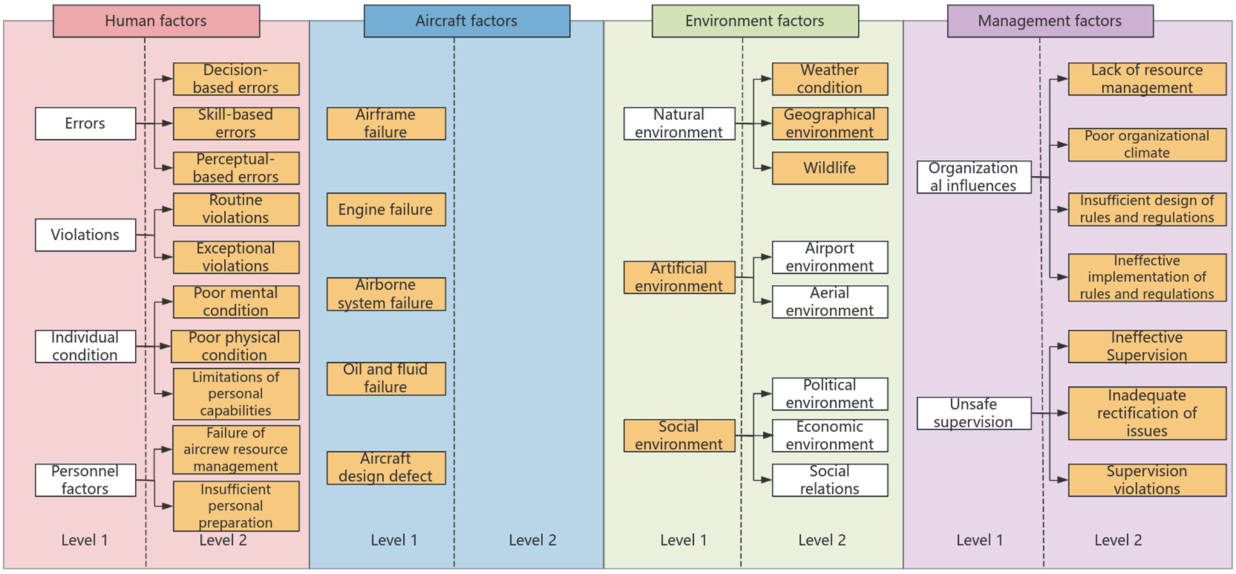


Figure 4. Concepts in AACKG ontology.

2.1.4. Information Extraction

Information extraction is the process of extracting structured data from large volumes of unstructured text, aiming to transform textual data into knowledge triplets. This process encompasses three primary steps: entity extraction, relation extraction, and event extraction. As a core component of constructing knowledge graphs, the quality of information extraction directly influences the quality of the final knowledge graph, with the accuracy of entity and relation extraction being particularly critical.

(1) Entity Extraction: It is also known as Named Entity Recognition (NER), which is the process of identifying words or phrases (entities) within unstructured text and categorizing them into specific classes (entity types) [28]. The technology of entity extraction has continuously evolved, progressing from rule-based extraction to unsupervised learning, then to supervised learning based on feature engineering, and, ultimately, to research methods based on deep learning [1]. However, the performance of entity extraction conducted through traditional methods heavily relies on the quality of sample annotation. In the process of entity extraction for aviation accident causes, issues such as a large number of entities, ambiguous boundaries for some entities, and

excessively long entities, pose challenges in ensuring annotation quality. Additionally, the annotation of long texts is inefficient, requiring significant human and time resources. Therefore, it is imperative to explore more effective extraction methods.

(2) Relation Extraction: This process aims to identify the relationships that exist between pairs of entities within a text, essentially categorizing the potential relationships between entity pairs. Through the analysis of the constructed aviation accident dataset, it has been found that the primary relationship in the causes of aviation accidents is causal. Additionally, there are other relationships, such as parallel, sequential, conditional, reversal, subordinate, and compositional. The specific meanings of these relationships are presented in Table 2.

Table 2. Relationship in AACKG.

Relationship Type	Meaning	Formalization	Words
Causal relationship	One event triggers the occurrence of another event.	A leads to B	lead_to
Parallel relationship	One event occurs simultaneously with another event.	A is B at the same time	and
Sequential relationship	One event occurs immediately following another event.	A is followed by B	follow_by
Conditional relationship	Under the condition of one event, another event occurs.	If A, then B	if_then
Reverse relationship	One event is in opposition to another event.	Although A, B	although
Subordinate relationship	One event is a superordinate or subordinate event to another event.	A is a type of B	type_of
Compositional relationship	One event is a constituent part of another event.	A constitutes B	constitute

Note: One event may be a causative factor or an accident type.

Deep learning currently represents the mainstream approach to information extraction, with LSTM [29], BERT [30], Global Pointer [31], and W2NER [32] each exhibiting unique strengths and limitations in their application. Among these, Long Short-Term Memory (LSTM), a specialized form of Recurrent Neural Network (RNN), efficiently captures long-range dependencies through its gating mechanisms and memory cells, demonstrating remarkable performance in sequence labeling tasks. BERT (Bidirectional Encoder Representations from Transformers), as a pre-trained language model, leverages bidirectional encoding and robust contextual representation capabilities to excel in NLP tasks. The Global Pointer framework reframes entity extraction as a global pointer prediction problem, utilizing matrices to represent all possible entity spans, effectively tackling complex entity issues. Meanwhile, W2NER translates the information extraction task into a relation classification problem between words, constructing a word graph and employing neural networks to learn node representations, thus naturally accommodating diverse entity types. These methods significantly outperform traditional methods in handling complex entity structures.

However, traditional information extraction methods rely heavily on various forms of data annotation, with extraction performance largely contingent upon the quality of sample annotations. In the context of aviation accident causal factor extraction, challenges such as numerous entities, unclear entity boundaries, and excessively long entities compromise annotation quality. Furthermore, annotating long texts is inefficient, necessitating significant human and time resources. Consequently, exploring more efficient extraction methods is imperative.

2.2. Overview of Claude–Prompt Method

2.2.1. Overview of LLMs

The rapid development of artificial intelligence, particularly LLMs represented by ChatGPT, has offered new possibilities for advancements in various fields [33]. In November 2022, ChatGPT, released by OpenAI, demonstrated question-answering capabilities that approached or even surpassed the average human level. Subsequently, other LLMs, such as Claude, were also introduced. LLMs, which originated from research in natural language processing and rely on deep learning with extensive parameters [34], are renowned for their powerful general knowledge and language processing capabilities. As LLMs continue to evolve and technological iterations accelerate, they are playing an increasingly significant role in the construction of knowledge graphs. John Dagdelen et al. proposed a method for jointly extracting entities and relations by fine-tuning pre-trained LLMs. This method can extract records from single sentences or entire texts in simple English sentences or more structured formats, providing a simple, accessible, and highly flexible approach for acquiring large, structured, domain-specific scientific knowledge bases from research papers [20]. Zhang et al. utilized LLMs to construct a Traditional Chinese Medicine (TCM) knowledge graph, accurately organizing and presenting various entities, attributes, and relationships in the TCM field, thereby providing a solid and promising foundation for the learning, research, application, and modernization of TCM [21].

Given the current situation in the field of aviation accident causation research, where traditional methods for information extraction suffer from low efficiency and accuracy that require improvement, the introduction of LLMs for information extraction presents significant advantages. Firstly, LLMs possess powerful natural language understanding capabilities. Through pre-training on large-scale corpora, they excel at comprehending and generalizing complex long texts for downstream tasks. Secondly, compared to traditional information extraction processes, methods based on LLMs do not necessitate extensive data annotation. This avoids the issue of inconsistent quality in the manual annotation of complex texts and the high labor costs associated with extensive labeling, thereby enhancing processing efficiency.

2.2.2. Claude–Prompt Method

To fully harness the potential of LLMs, it is necessary to undertake effective, prompt engineering. This involves designing and optimizing prompt inputs and, in the iterative process, examining factors such as context, phrasing, and grammar to effectively guide LLMs in producing the desired output and achieving optimal performance [35]. LLMs in specific tasks can be influenced by the nature of the prompts used, indicating that customized prompt techniques may be required for different LLMs [36].

In this study, the Claude-prompt method is adopted, utilizing Claude 3.5 as the model for information extraction. Claude 3.5, released in June 2024 by Anthropic, represents the company's latest model. Compared to its predecessor, Claude 3 Opus, Claude 3.5 offers advantages in terms of speed and cost and achieves notable performance enhancements in various aspects, such as capturing subtle differences and handling complex instructions [37]. As illustrated in Table 3 [37], Claude 3.5 exhibits superior text reasoning capabilities (reasoning over text) when compared to other mainstream models, and it also supports long text inputs.

Table 3. Reasoning over text performance between Claude 3.5 and other models.

Claude 3.5 Sonnet	Claude 3 Opus	GPT-4o	Gemini 1.5 Pro	Llama-400b (Early Snapshot)
87.1 3-shot	83.1 3-shot	83.4 3-shot	74.9 Variable shots	83.5 shot (Pre-trained model)

3. Experimental Design and Analysis

3.1. Construction of the Claude-Prompt Method

Figure 5 illustrates the working principle of the large model presented in this paper. It initially converts the Prompt and accident data into vector representations through an Embedding layer, which are then processed by the underlying Transformer blocks for general feature extraction and continuous learning. Subsequently, the data flows through two parallel sequences of task-specific Transformer blocks optimized for language understanding tasks. This design combines general representations with task-specific representations, enabling the model to comprehend and analyze complex aviation accident information effectively, even when the training data are limited. The original inputs (Prompt and Aviation accident data) are first transformed into Embedding vectors. These vectors traverse the general Transformer blocks, where they learn broad linguistic features and patterns. Following this, the data passes through specialized Transformer blocks, optimized for language understanding tasks. Ultimately, the model outputs the understanding and analysis results of the aviation accident data, namely, entity-relationship triplets.

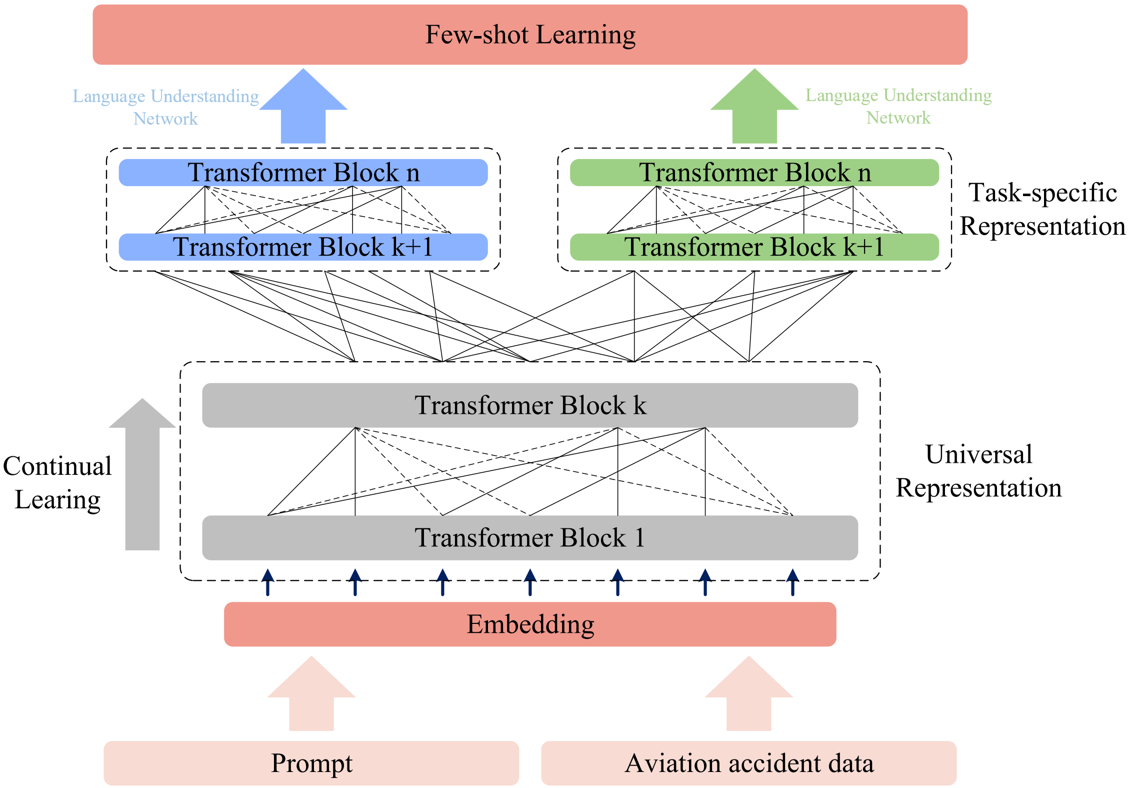
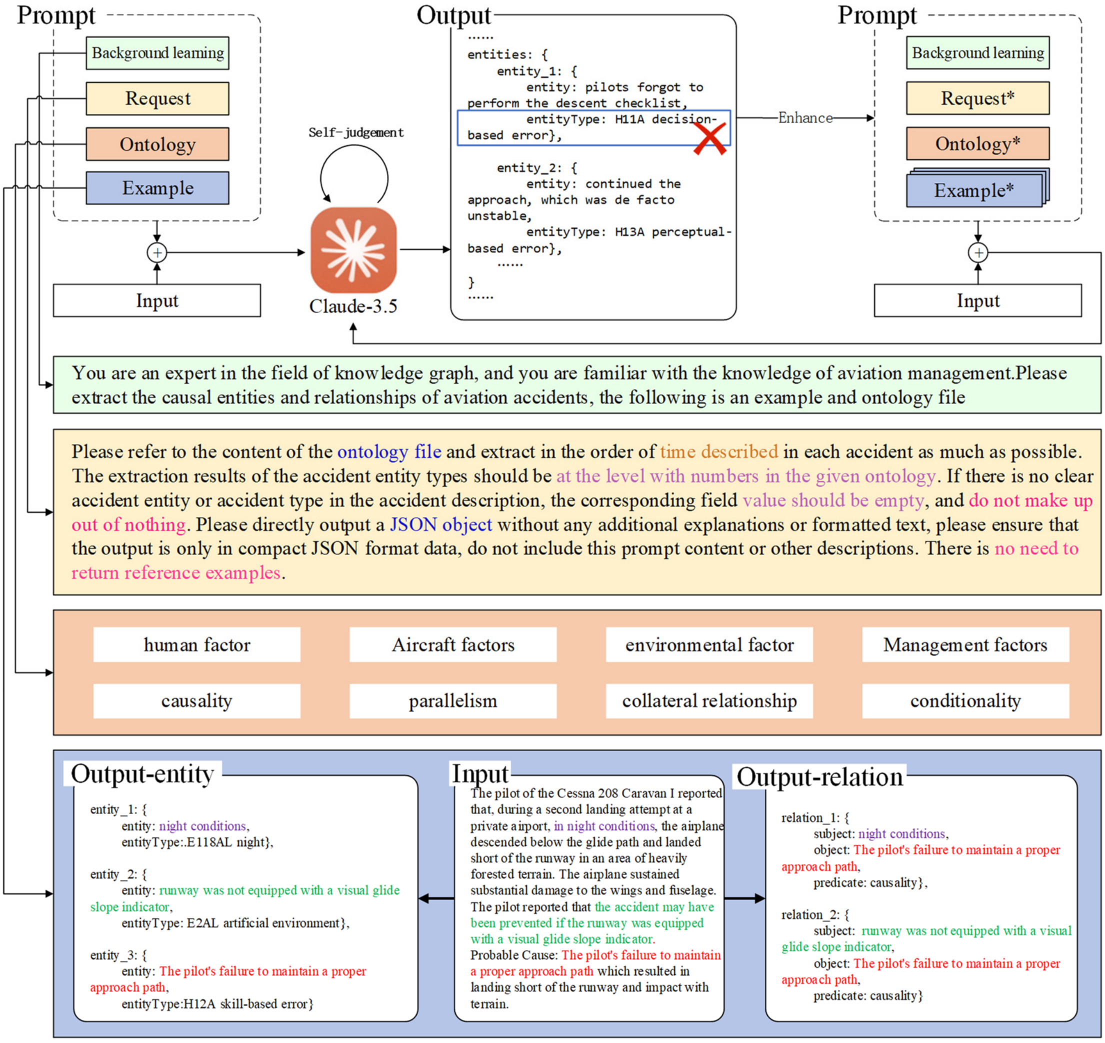


Figure 5. Principles of large language models processing raw data.

In this research, we introduce a novel methodology that incorporates prompting and a self-judgment mechanism tailored to tackle the intricate task of information extraction pertaining to the causation of complex aviation accidents. This methodology addresses the difficulties encountered by conventional models in handling complex relation extraction, especially when the complexity of accident causation entities and their relationships escalates, frequently constraining the efficacy of traditional methods. Our proposed approach fully capitalizes on the advanced context comprehension capabilities of the Claude 3.5 language model. Through the implementation of a strategic prompt design, the model is encouraged to delve deeper into the text content, thereby producing outputs with a higher degree of accuracy. This strategy not only streamlines the model's adaptation process for specific tasks but also markedly bolsters its performance stability and generalization capacity when confronted with intricate and variable relationship structures. Figure 6 depicts the architectural framework for constructing an LLM based on the Claude-prompt ethodology.



```
……
entities: {
    entity_1: {
        entity: pilots forgot to
perform the descent checklist,
        entityType: H11A decision-
based error},

    entity_2: {
        entity: continued the
approach, which was de facto
unstable,
        entityType: H13A perceptual-
based error},
    ……
}
……
```

You are an expert in the field of knowledge graph, and you are familiar with the knowledge of aviation management.Please extract the causal entities and relationships of aviation accidents, the following is an example and ontology file

Please refer to the content of the ontology file and extract in the order of time described in each accident as much as possible. The extraction results of the accident entity types should be at the level with numbers in the given ontology. If there is no clear accident entity or accident type in the accident description, the corresponding field value should be empty, and do not make up out of nothing. Please directly output a JSON object without any additional explanations or formatted text, please ensure that the output is only in compact JSON format data, do not include this prompt content or other descriptions. There is no need to return reference examples.

```
entity_1: {
    entity: night conditions,
    entityType:.E118AL night},

entity_2: {
    entity: runway was not equipped with a visual glide
slope indicator,
    entityType: E2AL artificial environment},

entity_3: {
    entity: The pilot's failure to maintain a proper
approach path,
    entityType:H12A skill-based error}
```

The pilot of the Cessna 208 Caravan I reported that, during a second landing attempt at a private aiport, in night conditions, the airplane descended below the glide path and landed short of the runway in an area of heavily forested terrain. The airplane sustained substantial damage to the wings and fuselage. The pilot reported that the accident may have been prevented if the runway was equipped with a visual glide slope indicator. Probable Cause: The pilot's failure to maintain a proper approach path which resulted in landing short of the runway and impact with terrain.

```
relation_1: {
    subject: night conditions,
    object: The pilot's failure to maintain a proper
approach path,
    predicate: causality},

relation_2: {
    subject:  runway was not equipped with a visual
glide slope indicator,
    object: The pilot's failure to maintain a proper
approach path,
    predicate: causality}
```

Figure 6. Framework of Claude-prompt method-based information extraction.

3.1.1. Few-Shot Prompt

In contextual learning, the goal of achieving better results is pursued by experimenting with different prompt words. By assigning the identity of a "knowledge graph and aviation safety domain expert" to the LLMs and inputting an ontology file containing entity types, the LLMs are instructed to identify and extract specific entities from the provided statements. This study involves various types of causation entities; therefore, the extraction process requires multiple iterations to be completed. The essence of this process is the transformation of a multi-class classification problem into multiple binary classification problems.

In the case presentation, to enhance the relevance of the prompted case content, we designed multiple samples for each prompt category. To achieve effective extraction of different entities and relationships, we crafted ten prompt samples for each entity type and relationship type. Specifically, for a given entity type, each set of prompt samples encompassed two distinct coverage scenarios: (a) sentences that do not contain any entities and (b) sentences that contain one or more entities. Each example consists of an input sequence in the JSON format and its corresponding output sequence. Five samples are included in each scenario to ensure a comprehensive evaluation of the model's performance across various contextual environments. During the sample selection process, the SimCSE model [38] is employed to calculate the semantic similarity between the target text and candidate samples, thereby determining the most suitable prompts. The KNN method was used within the SimCSE model to select the most similar K cases. The input for testing was a specific sentence from the test set, and the accuracy of the model's recognition was calculated by comparing the expected answer with the answer in the test set. Compared to other models, the SimCSE model demonstrated significant advantages in computational efficiency. Through this approach, we were able to ensure that the selected prompt samples were highly semantically relevant to the target text, thereby improving the accuracy and efficiency of named entity recognition. Furthermore, the diversified coverage scenarios helped the model better adapt to different textual environments and entity types, enhancing its generalization ability. Ultimately, the input for testing was a sentence from the test set, and the accuracy of the model's recognition was evaluated by comparing the expected answer with the provided answer in the test set.

The design of prompt information is a pivotal aspect in task completion, serving to inform the model of the required tasks and provide certain examples to deepen its understanding of those tasks. The methodology presented in this paper divides the entire prompt into three components: contextual learning, case presentation (with few-shot examples), and input data.

(1) Contextual Learning

Contextual learning clarifies the role and task requirements of the LLM, while precisely expressing and defining the knowledge to be extracted. It informs the LLM of the nature of the current task, sets a precise working framework, and taps into the model's potential to adapt to the task, ensuring that the model can accurately understand and complete the task in a targeted manner.

(2) Case Presentation

By introducing the few-shot learning method, specific task examples are provided for the model's reference. The aim is to further enhance the model's understanding of the task structure and expected output format, while standardizing the model's output results to optimize its comprehension and execution of the task.

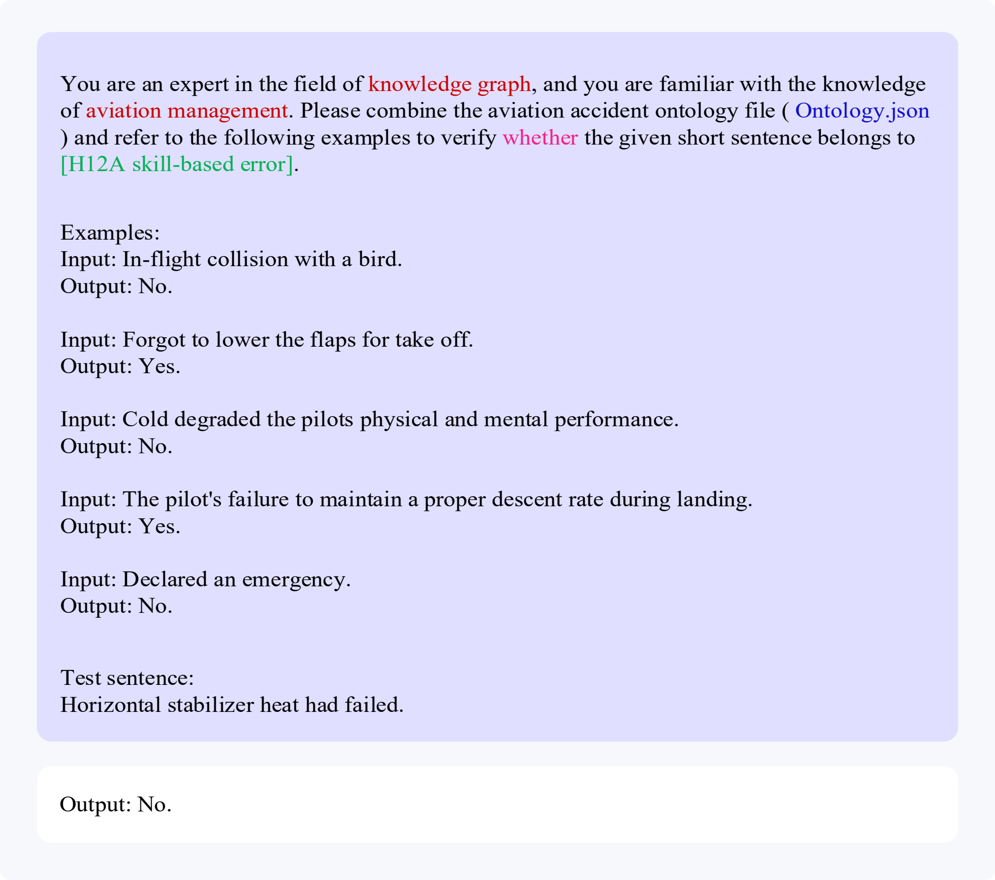
(3) Input Data

This refers to the data provided to the model for testing purposes.

3.1.2. Self-Judgment Mechanism

After a round of prompt-based training, the LLM is capable of learning to predict entities related to the causes of aviation accidents, but its accuracy may be insufficient. We have observed variations in the output results of the large model across different time periods. Taking into account the potential randomness of the outputs from generative models and the normativity of named entity recognition tasks, a self-judgment mechanism is introduced to verify whether the entities extracted from a given sentence belong to a specific entity type. This enhances the Claude-prompt model's ability to self-judgment the correctness of its inputs and reduces the impact of inherent model deficiencies, thereby optimizing the accuracy of the output results. Consequently, the performance of the generative model in the task of identifying entities related to aviation accident causes is effectively improved. Figure 7 illustrates an example of the self-judgment mechanism.

Figure 8 demonstrates the workflow of the self-judgment mechanism within the Claude-prompt model, which constitutes a meticulously designed iterative process. Initially, the input and prompts are fed into the Claude 3.5 model for processing, yielding preliminary outputs such as "In-flight collision with a bird". Subsequently, this output, along with an evaluation prompt, is reintroduced into the model for self-verification. The model assesses the result, triggering a re-reasoning process if it deems the result unreliable (with an evaluation answer of "No"); conversely, if the model considers the result reliable (with an evaluation answer of "Yes"), the result is saved in the JSON format. This self-checking and iterative mechanism significantly enhance the accuracy and reliability of the outputs, making it particularly suitable for complex tasks requiring high precision, such as an aviation accident causation analysis. Through continuous self-assessment and necessary re-reasoning, this mechanism ensures the superior quality of the ultimately generated knowledge graph data, providing a solid foundation for subsequent analyses and decision-making.



You are an expert in the field of knowledge graph, and you are familiar with the knowledge of aviation management. Please combine the aviation accident ontology file (Ontology.json) and refer to the following examples to verify whether the given short sentence belongs to [H12A skill-based error].

Examples:
Input: In-flight collision with a bird.
Output: No.

Input: Forgot to lower the flaps for take off.
Output: Yes.

Input: Cold degraded the pilots physical and mental performance.
Output: No.

Input: The pilot's failure to maintain a proper descent rate during landing.
Output: Yes.

Input: Declared an emergency.
Output: No.

Test sentence:
Horizontal stabilizer heat had failed.

Output: No.

Figure 7. An example of self -judgment mechanis.

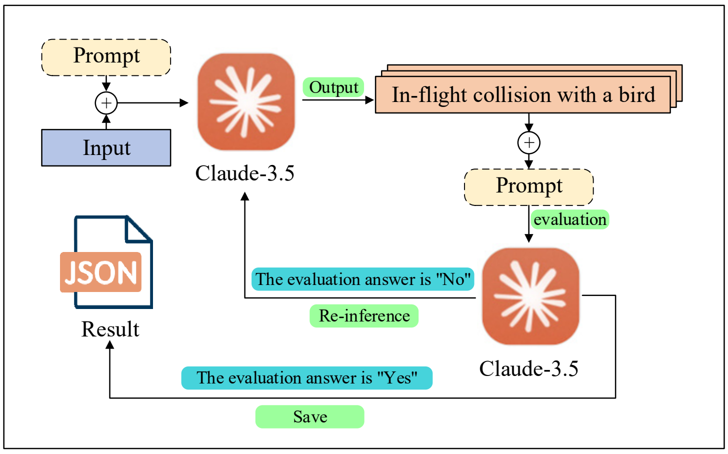

Prompt
Input
Claude-3.5
Output
In-flight collision with a bird
Prompt
evaluation
JSON
Result
The evaluation answer is "No"
Re-inference
Claude-3.5
The evaluation answer is "Yes"
Save

Figure 8. Claude-prompt model self-judgment mechanism workflow.

3.2. Experimental Results and Analysis

3.2.1. Evaluation Index

This paper employs the F1-score as the primary evaluation metric to test the performance and effectiveness of the information extraction model on the dataset, with Precision (P) and Recall (R) serving as auxiliary evaluation metrics. The F1-score takes into account both precision and recall in its calculation. The specific calculation methods for the evaluation metrics are as follows, as indicated by the formulas:

$$\mathrm{Precision} = \frac{\mathrm{n_p}}{\mathrm{n_p} + \mathrm{n_t}} \tag{1}$$

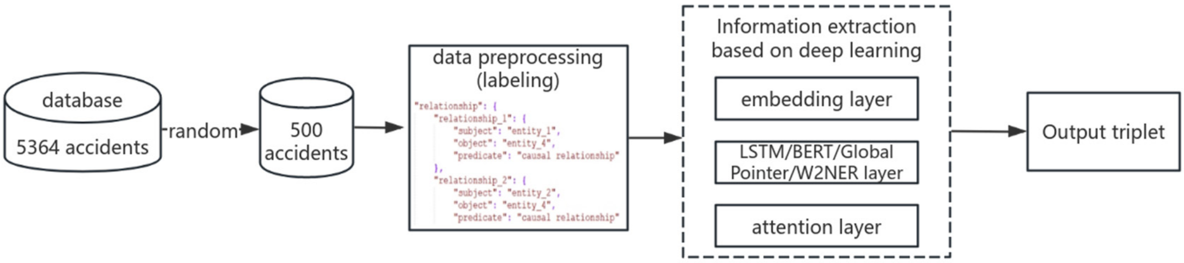
$$\mathrm{Recall} = \frac{\mathrm{n_p}}{\mathrm{n_c}} \tag{2}$$

$$\mathrm{F1} = \frac{2\mathrm{Precision} \times \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}} \tag{3}$$

In the formula, $\mathrm{n_p}$ represents the number of correctly identified instances, $\mathrm{n_t}$ represents the number of incorrectly identified instances, and $\mathrm{n_c}$ represents the total number of labels to be identified.

3.2.2. Performance Comparison

After conducting information extraction based on a Claude-prompt, traditional deep learning models such as LSTM, BERT, Global Pointer, and W2NER were utilized for further information extraction. Among the 5364 accident data obtained from Figure 2, approximately 10% of the data were randomly selected, manually annotated by aviation domain experts, and employed as the test set for evaluation. The resulting information extraction triplets were then acquired. The extraction process is illustrated in Figure 9. Notably, during the annotation phase, the approach based on deep learning is identical to that of large-scale model-based methods.


database
5364 accidents
random
500 accidents
data preprocessing (labeling)
Information extraction based on deep learning
embedding layer
LSTM/BERT/Global Pointer/W2NER layer
attention layer
Output triplet

```
"relationship": {
    "relationship_1": {
        "subject": "entity_1",
        "object": "entity_4",
        "predicate": "causal relationship"
    },
    "relationship_2": {
        "subject": "entity_2",
        "object": "entity_4",
        "predicate": "causal relationship"
```

Figure 9. Information Extraction Process Based on Deep Learning.

In the task of extracting entities related to aviation accident causes, experiments were conducted based on approximately 5364 data entries obtained through web scraping. To ensure the accuracy of the test results, we randomly selected about 10% of the data and annotated it through experts in the aviation field, using this as the test set. The effectiveness of the method proposed in this paper was verified through comparison experiments with traditional deep learning models, existing high-performance large models, ablation studies, and variations in the number of few-shot prompts. The experimental results are detailed in Tables 4–7.

Table 4. Comparison of evaluation metrics between Claude-prompt method and traditional deep learning models.

Models	Precision	Recall	F1
LSTM	55.84%	56.79%	56.31%
BERT	61.87%	59.08%	60.44%
Global Pointer	64.11%	64.54%	64.32%
W2NER	68.33%	69.15%	68.74%
Ours (Claude-prompts)	74.51%	68.73%	71.50%

Table 5. Comparison of evaluation metrics between Claude-prompt method and other LLMs.

Models	Precision	Recall	F1	Average Spending/Data	Average Time Taken/Data
ChatGLM-6B-prompts	65.26%	65.04%	65.15%	0.003 $	6.7 s
ChatGPT 3.5-prompts	69.66%	67.48%	68.55%	0.006 $	5.1 s
ChatGPT 4.0-prompts	72.73%	69.28%	70.96%	0.196 $	4.8 s
Ours (Claude-prompts)	74.51%	68.73%	71.50%	0.016 $	7.6 s

Table 6. Blation experiment comparison.

Models	Precision	Recall	F1
Claude 3.5	62.36%	65.69%	63.98%
Claude 3.5 (+self judgment)	65.16%	64.01%	64.57%
Claude 3.5 (+compute few-shot prompts)	68.92%	70.11%	69.51%
Ours (Claude-prompts)	74.51%	68.73%	71.50%

Table 7. Comparison of few-shot prompts methods.

Models	K	Random			Compute		
		Precision	Recall	F1	Precision	Recall	F1
Claude-prompt	1 shot	65.85%	64.12%	64.97%	68.01%	64.07%	65.98%
	3 shot	67.37%	65.31%	66.32%	69.74%	65.70%	67.66%
	5 shot	69.43%	66.13%	67.74%	70.79%	68.05%	69.39%
	10 shot	72.51%	66.57%	69.41%	74.51%	68.73%	71.50%

(1) Comparison with deep learning models

In Table 4, we compare our proposed model, Claude-prompt, with existing mainstream deep learning models for information extraction, such as BERT [30] and W2NER [32]. The experimental results demonstrate that the proposed Claude-prompt model in this paper outperforms other deep learning models across various indicators. From the perspective of addressing data dependency issues, the large model architecture is capable of overcoming the reliance on high-quality annotated data that exists in traditional deep learning neural network models, providing theoretical support for the promotion of the model in practical applications. Furthermore, even when trained with only ten sample data points for each entity type, Claude-prompt's performance scores surpass those of other models, indicating its efficiency and generalization ability. This further confirms the value of large language models in real-world application scenarios, particularly in situations where the data volume is limited, and features are not prominent, as large models can comprehend their meanings and complete tasks effectively.

(2) Comparison with High levels LLMs

In Table 5, we compare the accuracy of the Claude-prompt method with four existing LLMs: ChatGLM-6B, ChatGPT 3.5, and ChatGPT 4.0. The GPT models are trained on large-scale language data to generate coherent and logical dialogs; their training process emphasizes the models' natural language understanding and fluent conversational expression capabilities, covering a wide range of topics. However, Claude 3.5 adopts different training methods and techniques, focusing on the model's safety, ethics, and reliability during training. Additionally, it can handle longer text inputs, making it more suitable for large-scale and complex information extraction tasks related to aviation accident causes. We used the same prompting method and self-judgment strategy for all the aforementioned models. The results indicate that the Claude-prompts model outperforms the other models in multiple evaluation metrics.

Specifically, we have also conducted a comparative analysis of the usage costs and response times across various Large Language Models (LLMs), where "Average spending/data" represents the average cost incurred per data input, and "Average time taken/data" denotes the average response time for each input-output transaction. It is observable that Claude 3.5 exhibits a relatively longer average response time of 7.6 s among all models, which can be interpreted as reflecting its higher capacity to handle complexities. Notably, Claude 3.5's average cost stands at $0.016, which is more than ten times lower than the highest cost incurred by ChatGPT 4.0. Taking into account the F1 score, precision, recall, average cost, and time comprehensively, our model demonstrates the best performance in terms of overall effectiveness, particularly excelling in accuracy. Although it exhibits some disadvantages in terms of cost and time compared to ChatGPT 3.5, these concessions are deemed worthwhile in light of the significant performance enhancements achieved.

(3) Blation experiment

In Table 6, we compare the performance of the Claude-prompt model under four different learning scenarios: zero-sample prompt direct generation, the introduction of a self-judgment mechanism, the introduction of few-shot prompts, and the simultaneous introduction of both few-shot prompts and a self-judgment mechanism. Based on the experimental results, the following conclusions can be drawn: Firstly, the introduction of a self-judgment mechanism improves model performance. Although there is a slight decrease in recall, the precision (P) and the overall performance metric, F1 score, are significantly improved. This indicates that the self-judgment strategy reduces false positives while optimizing overall efficiency, demonstrating a positive effect, and improving the system's overall recognition efficiency. Secondly, the introduction of sample prompts selected through a similar text calculation method enhances model performance, with improvements in precision, recall, and F1 score. Compared to the original model that generates outputs directly from zero-sample prompts, providing the model with specific task examples for reference further enhances its understanding of task structure and expected output format. Simultaneously, it regulates the model's output results, optimizing its comprehension and execution of the task. Additionally, our experiments reveal that if the prompt template requires the model to repeat the entire test statement and use special symbols to mark the target entities, it increases the workload for the language model, resulting in poorer task performance. Conversely, adopting a strategy of directly answering with the entities aligns more closely with the tasks of LLMs during the pre-training stage, yielding better results. Finally, the simultaneous introduction of the few-shot prompt strategy and the self-judgment mechanism achieves optimal model performance, effectively addressing the task of information extraction for aviation accident causes.

(4) Comparison of few-shot prompts methods

Table 7 compares the impact of the number of few-shot prompt cases and the text selection method on model performance. Here, "Random" indicates randomly selecting K samples from the training data, while "Compute" represents screening samples using the SimCSE method. It can be observed that, regardless of whether the random method or the SimCSE model method proposed in this paper are used for sample selection, as the number of provided shots increases, the model's performance improves to some extent. Specifically, the SimCSE model method, which identifies K detailed cases similar to the test data by calculating text similarity for the model to learn from, outperforms the randomly selected samples. This demonstrates the effectiveness of the method proposed.

Based on these four experimental results, it can be concluded that LLMs exhibit poor performance in the absence of specific prompts and constraints. These models tend to incorrectly identify unrelated data as specific entities, resulting in a phenomenon akin to "fabrication" out of thin air, and their performance may even be inferior to traditional deep learning models. The method proposed in this paper, which combines few-shot prompts with a self-judgment mechanism, effectively constrains and fine-tunes existing

LLMs, serving as an efficacious means to mitigate this issue. Its performance on the dataset used in this paper surpasses that of traditional deep learning models and other LLMs.

4. Discussion

4.1. Implications of Our Findings

This paper proposes an AACKG (Aeronautical Accident Causal Knowledge Graph) information extraction method based on Claude-prompt. This method enhances the efficiency and accuracy of information extraction through prompt engineering, thereby promoting the effectiveness of AACKG. By comparing Claude-prompt, a large model approach with traditional knowledge graph extraction methods, with other large-scale models, it is demonstrated that Claude-prompt addresses the limitations of traditional methods in complex text processing to a certain extent, surpassing the performance and cost advantages of previous extraction methods. This showcases the effectiveness and potential of complex text tasks based on prompt engineering, enabling the efficient extraction of aviation accident causal information without a need for costly model training or fine-tuning. This conclusion aligns with the research that achieved performance optimization of the OpenMedLM platform through prompt engineering [39].

Overall, we find that using LLMs for aviation accident causal information extraction requires less effort while achieving high-quality extraction results, fulfilling the current demands for AACKG accuracy. Certainly, fine-tuning generally enables better performance [40]. However, due to the limitations of requiring a large amount of annotated data for model training and significant computational resource consumption, fine-tuning was not investigated in this study. As research progresses and more stringent requirements for information extraction are imposed, continued model training and optimization of current extraction results through fine-tuning will be pursued.

It is noteworthy that our research indicates that Claude-prompt exhibits exceptional performance without further model training or fine-tuning. By using only a few labeled samples for each entity and relationship type, the model can achieve performance close to that of fine-tuned models which require hundreds or even thousands of training samples. Generative models have the ability to learn from limited training data and generalize to new, unseen texts, making them excel in handling data with high diversity.

Despite the notable flexibility and generalization capabilities exhibited by LLMs in information extraction tasks, challenges persist in terms of accuracy, computational resource requirements, and the ability to process long texts. Firstly, due to the potential of LLMs to generate erroneous or nonsensical entities and relationships, their precision often falls short of that of specially designed rule-based or classification models when confronted with complex sentence structures or ambiguous contexts. Secondly, the "black-box" nature of LLMs renders their decision-making processes difficult to interpret, which is detrimental for applications requiring high interpretability. Lastly, although LLMs are capable of handling longer texts, their performance in accurately extracting entities and relationships within such texts may decline, particularly when contextual information is scattered throughout the entire document.

LLMs have demonstrated their formidable flexibility and generalization capabilities in information extraction tasks for knowledge graphs, particularly in the processing of diversified and complex texts. However, these models continue to face challenges in terms of accuracy, interpretability, and computational resource requirements. Subsequent efforts will focus on optimizing these models to enhance their accuracy and interpretability.

4.2. More Applications of AACKG

Constructing a knowledge graph based on big data plays a significant role in conducting analyses of aviation accident causation and in preventing such accidents [3].

(1) A more profound and multi-dimensional analysis of aviation accident causation can be conducted. Traditional accident causation analyses often rely on qualitative methods such as cause models [41] and indicator systems [42] for safety analyses, which struggle with the quantitative analysis of accident causes and fail to meet the demands for multi-dimensional

causation analyses and accident prevention. After constructing the AACKG, a large volume of cause-related texts is efficiently structured, extracting 14,768 aviation accident causal factors and 9655 causal relationships, laying the foundation for the subsequent analysis of key causes and relationships between causal factors. Furthermore, it enables the exploration of correlations between accident causes and other crucial accident factors such as accident types, flight phases, and flight times, thereby revealing potential accident patterns and occurrence regularities in aviation accidents.

(2) In the context of the expanding data volume in the aviation safety field and the continuous development of artificial intelligence technologies, establishing a new paradigm for the design, storage, mining, and knowledge transformation of massive safety data is crucial for enhancing aviation safety management capabilities and achieving digital transformation in the era of big data [11]. Through subsequent digital management and the establishment of a communication platform for data integration and fusion, an accident information search system can be developed to facilitate queries on accident cause relationships and accident case retrievals, as well as provide intelligent decision support for current safety management. Furthermore, it has the potential to play a more prominent role in emerging accident prevention models such as resilience and emergency response systems.

5. Conclusions

This paper focuses on unstructured long texts of aviation accident causes as the research object and proposes an information extraction method for aviation accident causes based on Claude-prompt. Leveraging the large-scale pre-trained language model, Claude 3.5, this method enhances the accuracy of entity and relationship extraction through a few-shot prompting and a self-judgment mechanism. Experimental results demonstrate that this approach can effectively identify relevant entities and their relationships in aviation accident reports, significantly improving the efficiency and accuracy of information extraction. This provides strong support for the subsequent construction of a structured aviation accident cause knowledge graph and facilitates deep mining, analysis, and prevention of aviation accidents.

Furthermore, the knowledge graph holds significant application value in the field of aviation safety. In response to the current deficiencies in research, future work will continue to be refined from the following dimensions: in the construction of AACKG, the diversity of aviation accident report data sources will be expanded, the ontology concepts of accident causes will be refined, the credibility of the data will be enhanced, and a larger-scale aviation accident knowledge graph will be developed. In terms of information extraction tasks, there is still room for improvement in accuracy. The extraction models will be further refined to enhance accuracy and interpretability. Moreover, given the superior performance of LLMs in information extraction tasks, future considerations include migrating LLMs to ontology construction, knowledge fusion, knowledge completion, and knowledge reasoning tasks in the construction of the knowledge graph, further leveraging LLMs to strengthen various stages of the knowledge graph construction process. Through continuous updates and improvements to the procedures and methods of knowledge graph construction, the accuracy and automation level of the aviation accident knowledge graph can be enhanced, facilitating subsequent analyses and prevention of accidents.

Author Contributions: Conceptualization, J.X. and L.C.; methodology, L.C. and T.W.; software, T.W. and J.L.; validation, J.L.; formal analysis, J.X.; investigation, L.C.; resources, T.W.; data curation, J.L.; writing—original draft preparation, L.C. and T.W.; writing—review and editing, J.L.; visualization, J.L.; supervision, J.X.; project administration, L.C.; funding acquisition, J.X. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China, grant number 52074309.

Data Availability Statement: The data presented in this study are available in [Aviation Safety Network] at [https://aviation-safety.net/about/ (accessed on 1 April 2024)], reference number [24] in our manuscript.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A. Construction of AACKG Ontology Based on HFACS Model

When classifying the ontology of aviation accident causation, a subdivision is made based on the HAEM framework, with reference to the structural framework of the HFACS model. On one hand, the HFACS classification is incorporated into the ontology classification; on the other hand, the logical relationship of the HFACS model is referenced, dividing all factors into active failures and latent failures. This facilitates subsequent analysis based on the HFACS model. Taking decision-based errors as an example, it is coded as H11A, where 'H' represents human factors, '11' signifies level 2 within human causes, 'A' indicates an active failure, 'L' stands for a latent failure, and 'AL' denotes either an active or latent failure.

Table A1. Concepts in AACKG ontology and code.

Level 1	Level 2	Definition
	Human factors	
H1A Errors	H11A Decision-based errors	It refers to the deliberate actions that have been carried out according to a plan, but the plan is inadequate or inappropriate for the situation; these actions are intentional and purposeful.
	H12A Skill-based errors	It refers to errors that occur due to inadequate attention and memory dysfunction, which constitute highly automated behaviors.
	H13A Perceptual-based errors	It refers to the occurrence of perceptual errors when an individual's perception does not align with the actual situation, particularly when the input of sensory information is reduced or differs from the usual.
H2A Violations	H21A Routine violations	It refers to deliberate acts of non-compliance with regulations and procedures, such as unauthorized approaches, violations of training rules, failure to inspect the aircraft after warning lights illuminate, and disregard for the takeoff manual.
	H22A Exceptional violations	It refers to isolated incidents that deviate significantly from established regulations, which are not typical behaviors of individuals and are not tolerated by management.
H3AL Individual condition	H31AL Poor mental condition	It refers to factors encompassing psychological conditions (such as stress, mental fatigue, and motivation) that affect job performance, leading to human errors or unsafe conditions.
	H32AL Poor physical condition	It refers to medical or physiological conditions (such as internal diseases, physical fatigue, and hypoxia) that impact job performance, constituting factors leading to human errors or unsafe situations.
	H33AL Limitations of personal capabilities	It refers to the operator's lack of physical or mental capacity to respond to a given situation, which impacts job performance, constituting a factor that leads to human errors or unsafe conditions.
H4AL Personnel factors	H41AL Failure of aircrew resource management	It refers to factors encompassing communication, coordination, planning, and team issues that influence an individual's practices, conditions, or actions, ultimately leading to human errors or unsafe situations.
	H42AL Insufficient personal preparation	It refers to the failure to adhere to off-duty activities necessary for optimal performance at work, such as complying with crew rest requirements and alcohol restrictions, which subsequently leads to human errors or unsafe conditions.

Table A1. *Cont.*

Level 1	Level 2	Definition
Aircraft factors		
A1AL Airframe failure		It refers to various factors related to the design and automation issues inherent in the aircraft itself.
A2AL Engine failure		
A3AL Airborne system failure		
A4AL Oil and fluid failure		
A5AL Aircraft design defect		
Environment factors		
E1AL Natural environment	E11AL Weather condition	Severe weather or climatic events: factors associated with hurricanes, winter storms, droughts, tornadoes, thunderstorms, lightning, and wind shear.
	E12AL Geographical environment	It refers to factors associated with mountainous terrain, airport elevation, airport terrain, and large water bodies such as oceans.
	E13AL Wildlife	It refers to wildlife factors encountered during flight and at airports.
E13AL Wildlife	E21AL Issues with the airport environment	Factors related to the artificial environment, such as issues with the infrastructure of the airport environment and abnormal activities like aerial combat.
	E22AL Abnormalities in the aerial environment	
E3AL Social environment	E31AL Political environment	Factors encompassing the political environment, economic environment, and social relationships.
	E32AL Economic environment	
	E33AL Social relations	
Management factors		
M1L Organizational influences	M11L Lack of resource management	It refers to organizational-level decisions regarding the allocation and maintenance of organizational assets.
	M12L Poor organizational climate	It refers to the internal work atmosphere of an organization, encompassing elements such as structure, policies, and culture.
	M13L Insufficient design of rules and regulations	It refers to issues in organizational decision-making and rule-setting for managing daily activities within the organization, such as operations, procedures, and supervision.
	M14L Ineffective implementation of rules and regulations	It refers to issues identified in the process of implementing and enforcing organizational decisions and rules related to the management of daily activities within the organization, such as operations, procedures, and supervision.
M2L Unsafe supervision	M21L Ineffective Supervision	Managers should provide appropriate guidance, training, leadership, supervision, or motivation to ensure that tasks are completed safely and effectively.
	M22L Inadequate rectification of issues	It refers to situations where managers are aware of defects but allow them to persist.
	M23L Supervision violations	The term refers to circumstances in which managers have knowledge of defects yet choose to permit their continued existence.

References

1. Xiong, M.; Wang, H.; Wong, Y.D.; Hou, Z. Enhancing Aviation Safety and Mitigating Accidents: A Study on Aviation Safety Hazard Identification. *Adv. Eng. Inf.* **2024**, *62*, 102732. [CrossRef]
2. Huesler, J.; Strobl, E. Predicting the Number of Fatalities in Extreme Civil Aviation Accidents. *J. Air Transp.* **2023**, *31*, 150–160. [CrossRef]
3. Jia, Q.; Fu, G.; Xie, X.; Xue, Y.; Hu, S. Enhancing Accident Cause Analysis through Text Classification and Accident Causation Theory: A Case Study of Coal Mine Gas Explosion Accidents. *Process Saf. Environ. Prot.* **2024**, *185*, 989–1002. [CrossRef]
4. Wang, X.; Gan, Z.; Xu, Y.; Liu, B.; Zheng, T. Extracting Domain-Specific Chinese Named Entities for Aviation Safety Reports: A Case Study. *Appl. Sci.* **2023**, *13*, 11003. [CrossRef]
5. Gao, Y.; Zhu, G.; Duan, Y.; Mao, J. Semantic Encoding Algorithm for Classification and Retrieval of Aviation Safety Reports. *IEEE Trans. Autom. Sci. Eng.* **2024**, 1–8. [CrossRef]
6. Jiao, Y.; Dong, J.; Han, J.; Sun, H. Classification and Causes Identification of Chinese Civil Aviation Incident Reports. *Appl. Sci.* **2022**, *12*, 10765. [CrossRef]
7. Tamašauskaitė, G.; Groth, P. Defining a Knowledge Graph Development Process Through a Systematic Review. *ACM Trans. Softw. Eng. Methodol.* **2023**, *32*, 1–40. [CrossRef]
8. Zhang, X.; Srinivasan, P.; Mahadevan, S. Sequential Deep Learning from NTSB Reports for Aviation Safety Prognosis. *Saf. Sci.* **2021**, *142*, 105390. [CrossRef]
9. Peng, C.; Xia, F.; Naseriparsa, M.; Osborne, F. Knowledge Graphs: Opportunities and Challenges. *Artif. Intell. Rev.* **2023**, *11*, 13071–13102. [CrossRef]
10. Gan, L.; Ye, B.; Huang, Z.; Xu, Y.; Chen, Q.; Shu, Y. Knowledge Graph Construction Based on Ship Collision Accident Reports to Improve Maritime Traffic Safety. *Ocean. Coast. Manag.* **2023**, *240*, 106660. [CrossRef]
11. Niu, Y.; Fan, Y.; Ju, X. Critical Review on Data-Driven Approaches for Learning from Accidents: Comparative Analysis and Future Research. *Saf. Sci.* **2024**, *171*, 106381. [CrossRef]
12. Liu, P.; Qian, L.; Zhao, X.; Tao, B. The Construction of Knowledge Graphs in the Aviation Assembly Domain Based on a Joint Knowledge Extraction Model. *IEEE Access* **2023**, *11*, 26483–26495. [CrossRef]
13. Wang, J.; Qu, J.; Zhao, Z.; Dong, X. SMAAMA: A Named Entity Alignment Method Based on Siamese Network Character Feature and Multi-Attribute Importance Feature for Chinese Civil Aviation. *J. King Saud Univ. Comput. Inf. Sci.* **2023**, *35*, 101856. [CrossRef]
14. Gong, W.; Guan, Z.; Sun, Y.; Zhu, Z.; Ye, S.; Zhang, S.; Yu, P.; Zhao, H. Civil Aviation Travel Question and Answer Method Using Knowledge Graphs and Deep Learning. *Electronics* **2023**, *12*, 2913. [CrossRef]
15. Ji, S.; Pan, S.; Cambria, E.; Marttinen, P.; Yu, P.S. A Survey on Knowledge Graphs: Representation, Acquisition, and Applications. IEEE Trans. Neural Netw. *Learning Syst.* **2022**, *33*, 494–514. [CrossRef]
16. Milosevic, N.; Thielemann, W. Comparison of Biomedical Relationship Extraction Methods and Models for Knowledge Graph Creation. *J. Web Semant.* **2023**, *75*, 100756. [CrossRef]
17. Jia, Q.; Fu, G.; Xie, X.; Hu, S.; Wu, Y.; Li, J. LPG Leakage and Explosion Accident Analysis Based on a New SAA Method. *J. Loss Prev. Process Ind.* **2021**, *71*, 104467. [CrossRef]
18. Perboli, G.; Gajetti, M.; Fedorov, S.; Giudice, S.L. Natural Language Processing for the Identification of Human Factors in Aviation Accidents Causes: An Application to the SHEL Methodology. *Expert Syst. Appl.* **2021**, *186*, 115694. [CrossRef]
19. Dechy, N.; Dien, Y.; Funnemark, E.; Roed-Larsen, S.; Stoop, J.; Valvisto, T.; Arellano, A.L.V. Results and Lessons Learned from the ESReDA's Accident Investigation Working Group: Introducing Article to "Safety Science" Special Issue on "Industrial Events Investigation. *Saf. Sci.* **2012**, *50*, 1380–1391. [CrossRef]
20. Dagdelen, J.; Dunn, A.; Lee, S.; Walker, N.; Rosen, A.S.; Ceder, G.; Persson, K.A.; Jain, A. Structured Information Extraction from Scientific Text with Large Language Models. *Nat. Commun.* **2024**, *15*, 1418. [CrossRef]
21. Zhang, Y.; Hao, Y. Traditional Chinese Medicine Knowledge Graph Construction Based on Large Language Models. *Electronics* **2024**, *13*, 1395. [CrossRef]
22. Pan, S.; Luo, L.; Wang, Y.; Chen, C.; Wang, J.; Wu, X. Unifying Large Language Models and Knowledge Graphs: A Roadmap. *IEEE Trans. Knowl. Data Eng.* **2024**, *36*, 3580–3599. [CrossRef]
23. Abu-Salih, B. Domain-Specific Knowledge Graphs: A Survey. *J. Netw. Comput. Appl.* **2021**, *185*, 103076. [CrossRef]
24. Aviation Safety Network. Available online: https://aviation-safety.net/about/ (accessed on 1 April 2024).
25. Madeira, T.; Melício, R.; Valério, D.; Santos, L. Machine Learning and Natural Language Processing for Prediction of Human Factors in Aviation Incident Reports. *Aerospace* **2021**, *8*, 47. [CrossRef]
26. Nogueira, R.P.R.; Melicio, R.; Valerio, D.; Santos, L.F.F.M. Learning Methods and Predictive Modeling to Identify Failure by Human Factors in the Aviation Industry. *Appl. Sci.* **2023**, *13*, 4069. [CrossRef]
27. Wiegmann, D.A.; Shappell, S.A. The Human Factors Analysis and Classification System (HFACS). In *A Human Error Approach to Aviation Accident Analysis*; Routledge: England, UK, 2017; pp. 45–71. [CrossRef]
28. Liu, J.; Luo, H.; Fang, W.; Love, P.E.D. A Contrastive Learning Framework for Safety Information Extraction in Construction. *Adv. Eng. Inform.* **2023**, *58*, 102194. [CrossRef]
29. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding. *arXiv*, **2019**, arXiv:1810.04805. [CrossRef]

30. Zhao, H.; Pang, H.; Feng, S.; Han, D. Overview of Chinese Named Entity Recognition Technology. *J. Chang. Univ. Technol.* **2021**, *42*, 444–450.
31. Su, J.; Murtadha, A.; Pan, S.; Hou, J.; Sun, J.; Huang, W.; Wen, B.; Liu, Y. Global Pointer: Novel Efficient Span-Based Approach for Named Entity Recognition. *arXiv* **2022**, arXiv:2208.03054. [CrossRef]
32. Li, J.; Fei, H.; Liu, J.; Wu, S.; Zhang, M.; Teng, C.; Ji, D.; Li, F. Unified Named Entity Recognition as Word-Word Relation Classification. *arXiv* **2021**, arXiv:2112.10070. [CrossRef]
33. Patil, R.; Heston, T.F.; Bhuse, V. Prompt Engineering in Healthcare. *Electronics* **2024**, *13*, 2961. [CrossRef]
34. Wang, D.; Wang, Y.; Jiang, X.; Zhang, Y.; Pang, Y.; Zhang, M. When Large Language Models Meet Optical Networks: Paving the Way for Automation. *Electronics* **2024**, *13*, 2529. [CrossRef]
35. Venerito, V.; Lalwani, D.; Del Vescovo, S.; Iannone, F.; Gupta, L. Prompt Engineering: The next Big Skill in Rheumatology Research. *Int. J. Rheum. Dis.* **2024**, *27*, e15157. [CrossRef] [PubMed]
36. Yuan, M.; Bao, P.; Yuan, J.; Shen, Y.; Chen, Z.; Xie, Y.; Zhao, J.; Li, Q.; Chen, Y.; Zhang, L.; et al. Large Language Models Illuminate a Progressive Pathway to Artificial Intelligent Healthcare Assistant. *Med. Plus* **2024**, *1*, 100030. [CrossRef]
37. Claude 3.5 Sonnet. Available online: https://www.anthropic.com/news/claude-3-5-sonnet (accessed on 1 July 2024).
38. Gao, T.; Yao, X.; Chen, D. SimCSE: Simple Contrastive Learning of Sentence Embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*; Moens, M.-F., Huang, X., Specia, L., Yih, S.W., Eds.; Association for Computational Linguistics: Punta Cana, Dominican Republic, 2021; pp. 6894–6910. [CrossRef]
39. Maharjan, J.; Garikipati, A.; Singh, N.P.; Cyrus, L.; Sharma, M.; Ciobanu, M.; Barnes, G.; Thapa, R.; Mao, Q.; Das, R. OpenMedLM: Prompt Engineering Can out-Perform Fine-Tuning in Medical Question-Answering with Open-Source Large Language Models. *Sci. Rep.* **2024**, *14*, 14156. [CrossRef]
40. Pornprasit, C.; Tantithamthavorn, C. Fine-Tuning and Prompt Engineering for Large Language Models-Based Code Review Automation. *Inf. Softw. Technol.* **2024**, *175*, 107523. [CrossRef]
41. Chi, C.-F.; Sigmund, D.; Lin, Y.-C.; Drury, C.G. The Development of a Scenario-Based Human-Machine-Environment-Procedure (HMEP) Classification Scheme for the Root Cause Analysis of Helicopter Accidents. *Appl. Ergon.* **2022**, *103*, 103771. [CrossRef]
42. Cui, L.; Zhang, J.; Ren, B.; Chen, H. Research on a New Aviation Safety Index and Its Solution under Uncertainty Conditions. *Saf. Sci.* **2018**, *107*, 55–61. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.