Original Article

Institution of MECHANICAL ENGINEERS

Journal of RISK AND RELIABILITY

Proc IMechE Part O:
J Risk and Reliability
1–18

DOI: 10.1177/1748006X241272831
journals.sagepub.com/home/pio
Sage

Integrating an incident dataset with a question and answering language model to assist hazard identification: Comparison of an extractive and generative model

Jon Ricketts, Weisi Guo, Jonathan Pelham and David Barry

Abstract

Robust hazard identification (HAZID) relies upon extensive knowledge of the system being analysed, the technical aspects, and how it will be used operationally. Typically, this knowledge is held by human participants who can draw out answers in natural language to hazard related questions based upon their own experience. However, several threats exist to this, such as high staff turnover, a poor learning from incidents capability or even insufficient Information Technology resources. Alternatively, incident databases hold vast amounts of hazard information that can be transformed into a source of knowledge. As mitigation to the aforementioned issues, this paper presents a Question and Answering (Q&A) Bidirectional Encoder Representations from Transformers (BERT) language model trained upon aviation incidents and a unique Q&A dataset. The model can extract answers to typical HAZID questions, based upon factual incident reports. Alongside this extractive approach, the paper also explores the use of a generative Large Language Model combined with an incident dataset. Both models proved a useful addition to HAZID activities based upon the Structured What If Technique (SWIFT), answering safety-themed questions based upon a retrieved context of incident reports that semantically matched the query. For the purposes of HAZID, it was suggested that the generative option is preferable based upon its ease of implementation, lower resource requirements and quality of responses. Additionally, it is shown that it is possible for organisations to train and create their own custom models for HAZID purposes. Future work may wish to consider the application of models that can hypothesize scenarios based upon incident reports, building further understanding to the relationships between causes, hazards and consequences.

Keywords

Natural language processing, hazard analysis, information retrieval, incident reporting, safety analysis

Date received: 15 October 2023; accepted: 13 July 2024

Introduction

Hazard identification (HAZID) and risk assessment techniques are a core component to the design, safety assessment and certification of safety critical systems. Many techniques such as Fault Tree Analysis (FTA), Functional Hazard Analysis (FHA) and Failure Mode Effect Analysis (FMEA) have remained unchanged while the systems they represent have become more advanced,[1] including additional complexity with human and system interaction. Furthermore, these techniques heavily rely on human knowledge to assess a given system for possible hazards. This knowledge may not always be available due to factors such as high staff turnover, poor Information Technology systems or the inability to learn from incidents.[2–4] However, the underlying information to support the knowledge can often be found in electronic databases such as incident reporting systems.

Incident reports form a source of important safety data, describing accidents and incidents and often the associated causes, consequences and hazards of which

School of Aerospace, Transport & Manufacturing, Cranfield University, Cranfield, UK

Corresponding author:
Jon Ricketts, School of Aerospace, Transport & Manufacturing, Cranfield University, College Road, Cranfield MK43 0AL, UK.
Email: j.ricketts@cranfield.ac.uk

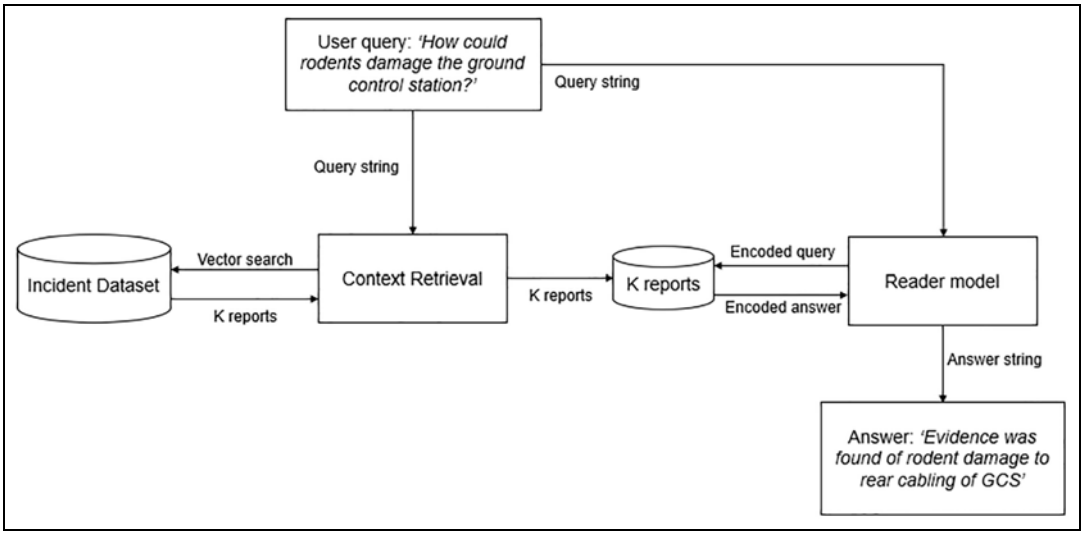

Figure 1. User query process utilising an incident dataset.

HAZID techniques should identify. A system that can draw out knowledge from a vast collection of incidents would enhance HAZID and could alleviate the aforementioned issues. The analysis of incident reports in this way also helps with understanding different mental models (described by Leveson[1]) adopted by the different communities involved in the design and operation of air systems. For example, improving the designers understanding of how systems are used, challenging design assumptions and decisions based upon the operator feedback.

This paper uses an incident dataset where incidents feature a free text narrative describing the events of the incident and any investigation. It is generally agreed that drawing upon knowledge from past incidents and accidents is required during HAZID.[3,5–7] Where previous research has covered deployment of generic Question and Answer (Q&A) models[8] or simply trialling ChatGPT for hazard analysis,[9] there is limited research around the use of developing bespoke safety-themed Q&A models from incident datasets, or utilising generative language models in conjunction with incident datasets to support HAZID. A gap in the knowledge addressed by this paper.

With the rapid progress of machine learning and availability of data, opportunities exist to apply machine learning techniques to achieve this aim. One such area within natural language processing (NLP) is the field of Q&A systems. Latest state of the art machine learning models such as Bidirectional Encoder Representations from Transformers (BERT) typically utilize a Q&A dataset (usually a set of Q&A pairs with a corresponding narrative) on which the model can be fine-tuned or trained. Alternatively, Generative Pre-trained Transformers (GPT) such as ChatGPT have been trained on vast amounts of text, followed by reinforcement learning to achieve Q&A ability – although the output is not always accurate, prioritising a human-like response over factual correctness.[10]

This paper describes the development of an extractive Q&A model and the augmentation of incident data with an openly available, generative Large Language Model (LLM). The general concept is displayed in Figure 1, where during a HAZID activity, the user can 'ask' the incident dataset questions. Similar incidents to the user query are returned by the retrieval method while the model attempts to extract (or generate) an answer from these. The answers can then inform the HAZID activity on historic, similar events leading to suggested mitigations for the design and operation of the given system. In effect, forming a tool to alleviate threats to learning from safety incidents such as the lack of focus on smaller incidents and the lack of an effective learning from incidents system.[2]

Remotely Piloted Air Systems (RPAS) was selected as the trial topic because it is an area of increasing popularity and hence represents a community where assistive technology such as this tool could play a beneficial role. It should be noted that the methods described in this paper are not limited to RPAS but could equally be applied to other aircraft types or safety critical systems.

This paper does not set out to examine and select the best language model for the task as this is covered in great depth within papers that focus on the topical computer science aspects.[11–14] Rather, the objective is to develop both an extractive model, and a pipeline featuring a generative model that can be used during a HAZID activity where the concept of using a language model in this setting can be trialled.

The paper is set out as follows; Section 2 of this paper covers the background elements to the work, building an argument as to why it would be useful. Section 3 details the development of the models and

evaluation criteria. Sections 4 and 5 present and discuss the results of the deployed models along with further development. Section 6 concludes.

Background

This research draws together several areas of research: hazard identification, knowledge within risk assessment and Q&A datasets/models. These are covered in the following sub-sections.

Q&A models and datasets

Q&A datasets and associated models are a subset of NLP, where the general aim is to accurately answer questions posed by human beings in natural language. Q&A frameworks are not a new advent within the field of NLP, having been in development since the 1960's. Simmons[15] neatly captures these early systems within a survey where the majority were rule based. The advancements of machine learning models have led to further improvements in Q&A framework functionality and adoption across multiple domains.

Latest state of art solutions includes BERT and GPT models. Both models were trained on large textual datasets, for example GPT-3 was trained on 300 billion tokens, while BERT was pre-trained on 3.3 billion tokens.[16,17] The larger dataset tends to give GPT-3 the advantage in translation or summarisation type tasks. Likewise, there is vast delta in model parameters with GPT-3 featuring 1.5 billion parameters and BERT-large 340 million parameters – it is suggested that once parameters exceed a certain level, the model gains special abilities such as context learning.[17] However, both models have similarities in the fact they utilise the transformer architecture and learn context from text using attention mechanisms in an unsupervised way. Overall, both models are strong contenders for many present-day NLP tasks, the selection is often based upon the intended output, available computing resources, data restriction requirements and whether the model is open-source. Many organisations do not have the sheer quantity of data available to train their own models, and in many cases not the knowledge to do so. An optimum solution is to use publicly available datasets and models while fine-tuning and adapting them for bespoke data, such as the process described in this paper.

The key component to many present-day Q&A models is the size and quality of the Q&A dataset for which the model is trained or fine-tuned upon. From reviewing existing datasets,[18–24] it can be seen that extensive resource is required for their creation, often utilising crowd workers to generate, in some cases 100,000 + questions. This presents a challenge for the task at hand, where resource such as crowd workers are unavailable, and the security aspects of the data prevent its wider distribution. To the best of the authors knowledge, no safety specific Q&A datasets currently exist.

A viable option is to fine-tune an existing model with a bespoke dataset, where the model's parameters are changed for the new task. This can be convenient as a general model can be adapted for multiple tasks. Alternatively, feature extraction can be applied, in this instance the model's weights are 'frozen' and the pre-trained representations are used in a downstream model.[25] However, pre-trained models may not have encountered aviation specific terminology and use of language resulting in unknown words (or tokens). To resolve this, it is beneficial to train a model from scratch on the aviation data before fine-tuning for a specific task.

Alternatively, a generative LLM such as ChatGPT can be deployed 'off the shelf' with a context retrieval process that supplies relevant incident data as a context to generate a response. This is known as a 'prompt' where the LLM is programmed through instructions provided in natural language.[26] In theory, this should leverage the generative capability of the model while grounding the response to previously recorded events.

Hazard identification: SWIFT

The basis for the Q&A dataset described later in this paper is the Structured What-If Technique (SWIFT). SWIFT has similarities to other HAZID techniques, therefore, it should be possible to use the described dataset with other techniques.

SWIFT was created within the chemical industry in the 1960's as a more convenient alternative to the commonly used HAZOP method. Rather than wildly asking 'what if' style questions the technique features a structured, system orientated approach containing the questions in the form of a checklist with guideword prompts to increase completeness of the given system under analysis. SWIFT has since spread from the chemical industry and is captured within ISO 31010 has a technique for identifying risk.[27]

The actual SWIFT process will vary between industries and organisations, however the principle remains the same. The team should ideally have varied backgrounds (e.g. maintainers, aircrew and designers) whilst having relevant experience for the scope of the analysis. All team members should participate, asking questions, expressing opinions and concerns. The 'what if' questions which should follow structured categories in order to maximise questioning and coverage of the system. The categories of the questions will vary between industries, an example for the process industry is covered by Crawley and Tyler.[28] The Acquisition Safety and Environmental Management System (ASEMS)[29] online resource covers general question categories.

The 'What-if' questions do not have to strictly begin with those terms but 'How could' and 'Is it possible' may be used. Ultimately, the intent is to ask questions that will ensure the group carefully considers potential scenarios and possible causes and consequences. Appropriate mitigations to prevent the consequences

being realised can then be recorded to inform the design or operation of the system, ensuring the identified mitigation activity is carried out.

Knowledge within risk assessment

Being that the aim of the models developed in this paper is to unlock knowledge associated with a given risk, it is important to understand risk, how it might be assessed and quantified. A simple definition of risk is 'the potential for undesirable consequences of an activity'.[30] For example, risk is present during an aircraft flight, an accident might occur (negative consequence), alternatively the flight may conclude in a successful landing (positive consequence). Overall, risk is about these positive and negative consequences as well as the uncertainties around these consequences.

Apostolakis[31] formed a model on how to treat these uncertainties when it comes to assessing risk. The model is expressed as $G(\varphi| M,H)$ featuring a function G with input parameters φ. M is the set of model assumptions that define the model, and H is the entire body of knowledge and beliefs of the modeller. Both the input parameters and model assumptions are subject to uncertainties, where Apostolakis suggests subjective probabilities are used to express these. This leads to a probability statement of $P(A| H)$ where the probability of event A occurring is based upon knowledge H. It is this knowledge that is formed from expert opinions – the difficulty often lies in retrieving and quantifying this knowledge.

Being that knowledge is often tacitly held by individuals within a risk assessment or HAZID scenario, it is difficult to assess the quality and strength of this knowledge. Aven[32] describes two methods to measure the strength of knowledge, one factor that indicates strong knowledge is having reliable data available. If accessible and usable data can be drawn from incident datasets, then this would go some way to providing reliable data, that is, data that has been curated by competent people in the given field, assessed and describing factual events.

Many safety practitioners will be familiar with assigning probabilistic values to safety events, however as highlighted by Kay[33] 'Thinking probabilities does not come easily to the human mind. Telling stories and narratives do'. Although Kay was considering financial markets, the same concept can be applied to system safety and the narratives within incident reports, where there could be an opportunity to link narratives to probabilistic events.

This knowledge element of risk assessment has been further explored by Aven[32] where knowledge is required to support and strengthen the given probability, while consideration should be given to surprises that can occur relative to the knowledge of those involved in the assessment process.

Having established that 'knowledge' is key to understanding and assessing risk, the next challenge is capturing this knowledge. For risk assessments, knowledge is typically acquired from the people involved in the process and their experiences, this in turn can be supported by information. In the absence of human knowledge, it is hoped that the dataset and model developed through this paper will augment typical HAZID processes.

Incident dataset

An incident dataset has been proposed as the source of textual data for both training the models and developing the Q&A dataset. Incident reports are a rich source of information for HAZID activities, being they describe incidents and accidents along with the associated investigation and findings. This information can then inform both current and future systems as to 'what works well' and ultimately reduce the likelihood of accidents.

The incident dataset used by this paper is from the Air Safety Information Management System (ASIMS) which was introduced in 2009, and is a web-based tool for the reporting, management and analysis of air safety incidents, investigations and recommendations within the UK military aviation domain.[34] ASIMS allows for the raw incident data to be downloaded and further analysed via common software applications such as Microsoft Excel, or in the case of this paper – Python libraries.

The scope of the study: RPAS

Remotely Pilot Air Systems (RPAS) have been selected as the subject on which to model and trial the solution. This ensured that Q&A dataset would be bound to a given theme rather than attempting to generate questions covering all aviation. RPAS is a rapidly evolving area both within civil and military applications, with many uses exhibiting extra safety concerns. For example, the delivery of life-saving drugs to remote areas or the carriage of munitions within a conflict zone.

RPAS are then typically split into categories dependent upon their weight and intended operation. The work undertaken within this paper is applicable to all categories where much of the operation and potential hazards are common. An overlap then features between the certified category and crewed aviation, for example hydraulic systems being a common system (Figure 2).

RPAS can be developed at a quicker pace than traditional air systems, within a regulation space that is currently catching up with technology. Comparatively, traditional air system development tends to progress slower, and the time and resource required to identify hazards or offer safety analysis is widely known and dictated within standards.

There are many smaller companies developing RPAS, hence a tool that offers knowledge gained from the wider domain could be of real significance for identifying unforeseen hazards. Overall, RPAS is a

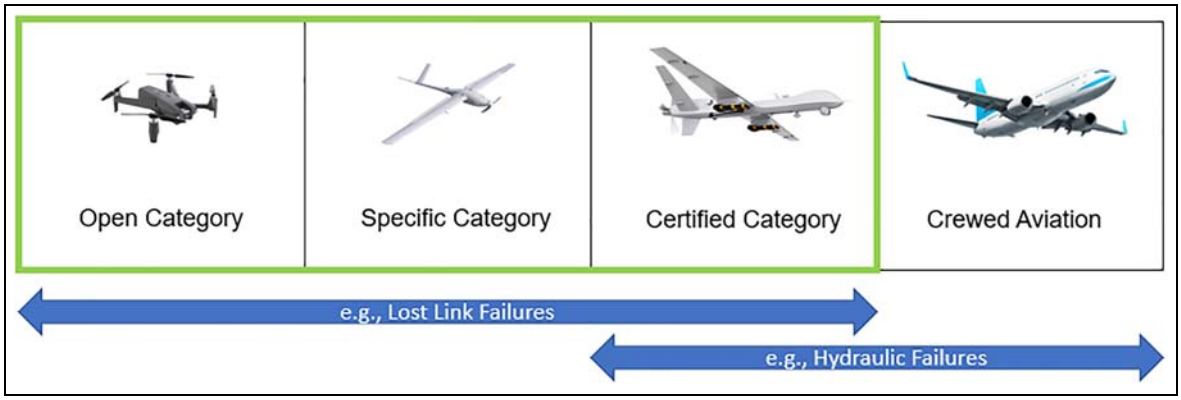

Figure 2. RPAS categories.

developing domain where a successful NLP tool for improving safety should offer a benefit moving forward.

Method

This paper adopts two methods, that are each described within this section; (1) to create an extractive Q&A model and (2), to augment a readily available generative model before the usage of both is compared.

Method 1: Extractive Q&A model development

The overall process for developing the extractive Q&A model is shown in Figure 3. The rationale for using and developing BERT and associated dataset is described in the following sections.

BERT. BERT was selected as the basis of the extractive method due to its previous success in Q&A and evidence retrieval methodologies as well as having a wide-range of supporting resources.[8,35–38] The structure of

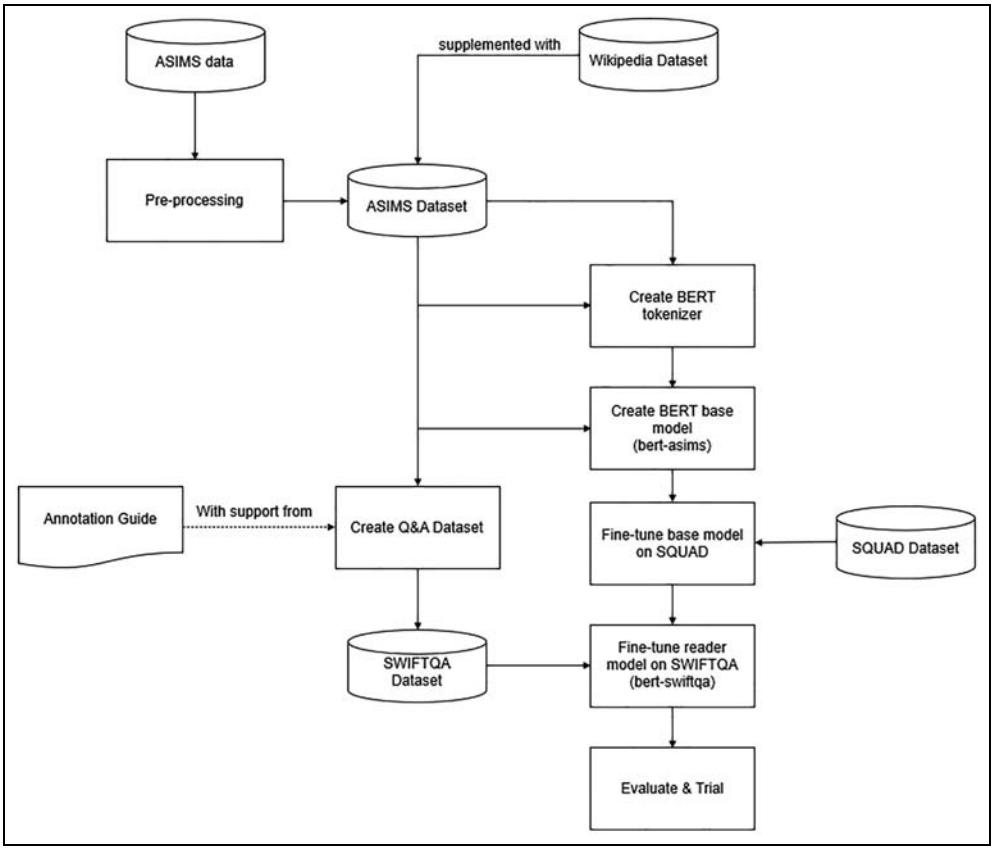

Figure 3. General process for creating Q&A dataset and extractive model in this paper.

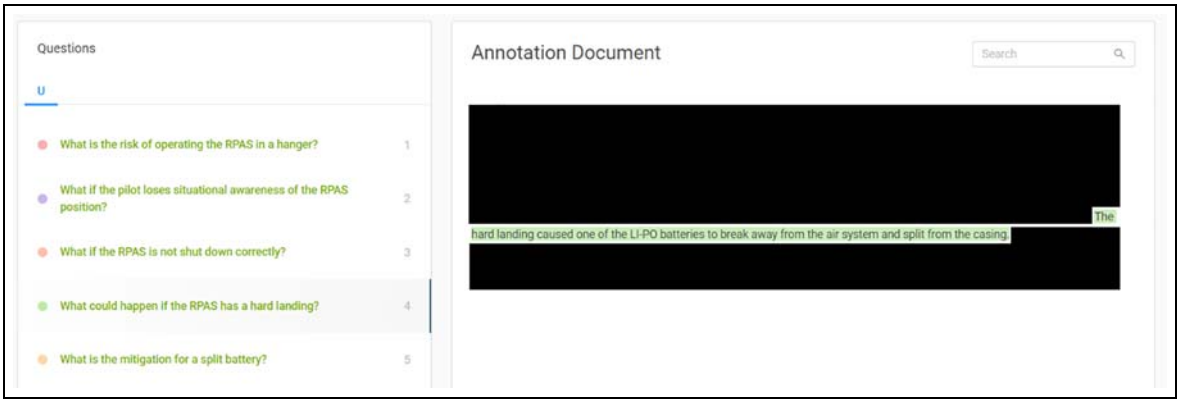

Figure 4. Screenshot from question/answer annotation (full incident data omitted).

the BERT language model consists of a stacked multi-layer bi-directional transformer along with a WordPiece tokenizer as the method to segment words.[39,40] BERT is a deep learning model where every output element is connected to every input element, and the weightings between them are dynamically calculated based upon their connection, a process called 'attention'. Where previous large language models such as Recurrent Neural Networks and Convolutional Neural Networks worked with sequenced data, BERT operates bidirectionally allowing training on larger amounts of data and resulting in improved performance such as disambiguation of polysemous words. This is an advantage when working with incident reports that are often polysemic in nature and present a difficulty for NLP.[38]

BERT is capable of being trained on further data or 'fine-tuned' for a specific task, such as Q&A that is described in this paper. BERT achieved high F1 scores when trained on the SQuAD.[39] Ultimately, BERT was selected for this work due to its good performance, open-source architecture, while having extensive research and guidance covering its abilities and further fine-tuning.

SQuAD. SQuAD is a curated dataset consisting of Q&A pairs associated with a given context, or passage of text. It was annotated by Amazon mechanical turk crowd workers to produce a dataset large enough to enhance the training of machine learning models, in this case the annotation of 23,215 passages of text resulting in 107,785 Q&A pairs.[18] The benefits of SQuAD not only include its huge size but it's challenging nature, where the answer is contained in a smaller passage of text rather than across multiple documents. SQuAD also requires more reasoning through complex answers.

The success of SQuAD has led to further research and development of several software tools that allow the curation of bespoke SQuAD-like datasets, although it is possible to create these with simple Python libraries. This allows the bespoke datasets to be either combined with the original SQuAD or used to fine-tune a model after training on SQuAD.

Ultimately, the use of both BERT and SQuAD should create a reliable model that can provide accurate answers. BERT trained upon SQuAD v1.1 question answering had an $F1$ score of 93.2, and on SQuAD v2.0 an $F1$ score of 83.1.

'SWIFTQA' dataset creation. This section describes the process for creating the bespoke Q&A dataset based on safety 'what if?' style questions, titled 'SWIFTQA'. The dataset format was based upon that of SQuAD, this ensured that the dataset could be processed with ease using existing python modules while also allowing datasets to be merged if required.

The ASIMS data was considered to be of good quality describing incidents and subsequent investigation in sufficient detail, having been written by competent personnel – that is people who have a good technical knowledge such as maintainers, engineers and aircrew. Finally, the text had essentially been peer-reviewed, that is the data had been checked and assured through the assessment and release of individual incident reports.

Incident data from the RPAS category was extracted before being processed to create a corpus that could be annotated to create a dataset saved as csv files where each row was an incident, and the columns represented the different fields. To be annotated to form a Q&A dataset, the incident data was pre-processed into a dataframe of two columns where the first column contained the incident reference (document_id). The second column contained free text containing a description of what happened, any investigation and lessons learnt merged into a single passage (document_text).

The primary aim of the model presented in this paper is to allow sematic searching of incident records and answer 'what if?' type questions to support a SWIFT (HAZID activity). To realise this aim, the dataset was created using the Haystack tool.[41] The pre-processed dataframe was loaded, allowing individual incidents to be explored and questions generated by the annotator(s) while corresponding answers were highlighted in the text (Figure 4).

Annotation was conducted using the Haystack guide[42] for best practice, in combination with hazard identification prompts shown in Table 1. Many of these prompts were derived from an earlier issue of Defence Standard 00-56,[43] and updated for RPAS operations, enabling the annotator(s) to frame questions from a hazard analysis perspective.

The final dataset consisted of 760 question and answer pairs and was saved as a .json file in the same format as SQuAD so that it could be utilized for model training.

Model development. The initial step was to train a BERT model from scratch on the incident data and additional English language resource. The incident dataset contained 117,694 incident reports (over the period 2010 – 2020) that were then processed so that each incident formed a section of continuous text containing data from the following columns:

- Description.
- Outcome Narrative.
- Cause Narrative.
- Causal Factor Narrative.
- Investigation and Rectification Work.

Presenting the data in this way allowed the incident to have a standard format explaining 'what happened?' that is, the consequences of the incident, followed by causes and mitigation activity. These combined narratives were then saved into .txt files where each new line was an incident.

The low quantity of incident data presents a challenge when the aforementioned language models have been trained upon Terabytes of data. To increase versatility, a Huggingface Wikipedia dataset was also included for training.[44] The Huggingface hub is a repository containing open-source pre-trained models and datasets that can be used in projects such as this.

Wikipedia data was also required as this is the 'language' the SQuAD is based upon. By mixing datasets, the tokenizer should be able to recognize not only English (Wikipedia) language but also the unique terminology used within incident reporting.

A BERT model was trained on the data via a Masked Language Modelling (MLM) process. This consists of masking a certain percentage of the tokens in each sentence, with the model trained to predict those masked words. At this stage the model architecture must be defined, for example, the number of layers, heads and dimensions. The parameters are defined below (any not stated were left as defaults):

- vocab_size – Vocabulary size of the BERT model. Defines the number of different tokens that can be represented by the inputs_ids passed when calling BertModel. Set to 30,522.
- num_hidden_layers – Number of hidden layers in the Transformer encoder, set to 12.
- num_attention_heads – Number of attention heads for each attention layer in the Transformer encoder. Set to 12.
- hidden_dropout_prob – The dropout probability for all fully connected layers in the embeddings, encoder and pooler. Set to 0.1.
- attention_probs_dropout_prob – The dropout ratio for the attention probabilities. Set to 0.1.
- max_position_embeddings – The maximum sequence length that this model can be used with. Set to 512.
- position_embedding_type – Type of position embedding. Set to 'absolute' for positional embeddings.[45]

The baseline BERT model could then be initially fine-tuned on the SQuAD followed by the SWIFTQA dataset to benefit from transfer learning effects. This created a separate 'reader' BERT model (titled 'bert-swiftqa') capable of extracting answers from a given context (provided via retrieval method). Rather than 'read' the context, the reader model reads integers representing words or sub-words as input IDs. This returns a span of input ID positions that are then translated into a human-readable answer.

The training parameters used for training the baseline BERT model on SQuAD and SWIFTQA were the same as those described by Moller et al.[46] and Zhang et al.[47]

Context retrieval method. The purpose of the context retrieval method is to return the reports that closest match the query so that the reader model does not have to search an entire dataset. Several solutions exist to achieve this; including a language model adapted for the task, a vector database or processes such as cosine similarity or Euclidean distance.[48–51]

As the incident dataset is not substantial in size, this method used cosine similarity for retrieval. This retrieval method ensures each incident is represented by a vector, where the coordinates correspond to individual words weighed by the frequency of their distribution within the incident and across all incidents, this is known as Term Frequency-Inverse Document Frequency (TF-IDF).[52] The user's inputted question vector can then be compared against incident vectors by measuring the distance (or similarity) between them (Figure 5).

Cosine values that are closer to '1' represent similar content, while those closer to '0' (raw frequency values are non-negative) represent dissimilar content. Cosine similarity provides a reliable method because it represents a measurement of orientation and not magnitude (Euclidean distance). To allow for a broad capture of incidents for the reader model, the 20 closest incidents were returned for each entered question.

Table 1. Prompts for what if style questions.

Top-level functional hazard category	Contributing factor prompts
Loss of control	- Reduced thrust (engine/motor failure/power system failure).
	- Malfunction of control system.
	- Datalink disruption.
	- Computer viruses.
	- System redundancy/diversity.
	- Hardware and software controls.
	- Inadvertent activation.
	- Insect, rodent or mould damage.
	- Foreign bodies and dust.
	- Electromagnetic interference.
	- Faults in supporting systems; for example, power supplies and hydraulic systems.
Loss of structural integrity	- Physical deformation.
	- Breakages.
	- Environment – weather conditions (gusts, etc).
	- Limitations of the platform.
Fire	- Electrical faults.
	- Battery faults.
	- Fuel system leaks.
	- Engine overheat.
	- Fire suppression system failure.
	- Ventilation.
	- Fire and explosion initiation and propagation.
	- Flammable substances; for example, solid, liquid or gaseous.
Controlled flight into terrain	- Reduced situational awareness.
	- Sensing systems failure.
	- Inaccurate mission profile.
	- Computer viruses.
	- Damage tolerance.
	- Hazardous materiel containment.
	- Effect of factors such as equipment layout, ergonomics and lighting.
	- Inadvertent activation.
Landing site incursion/excursion	- Method of launch/recovery.
	- Adequacy and effectiveness of instruction, training and rehearsal.
	- Effect of factors such as equipment layout, ergonomics and lighting.
Mid-air collision	- Congested airspace.
	- Part of a swarm.
	- Airprox with manned air systems.
	- Air traffic control organisation.
Payload – negative influences on system	- Electromagnetic Interference effects.
	- Weight and balance.
	- Release effects.
Ground handling	- Health and Safety hazards (trips, slips and cuts).
	- Hazardous materials.
	- Toxic substance control.
	- Personal protective equipment.
	- Noise or radiation barriers.
	- Alarms and warnings.
	- Electrical protection.
	- Training.
	- Lasers.
	- Explosives.
	- Asphyxiants, toxic or corrosive substances.
	- High temperature or cryogenic fluids.
	- Hazardous construction materials.
	- Pressure systems.
	- Electrical sources.
	- Ionising and non-ionising radiation sources.
	- Hydraulic arms or rotational machinery.
	- Other energy sources including those due to motion.
	- Exhaust gases.
	- Passive obstacles.
	- Hazardous surfaces.
	- Cut and puncture projections.

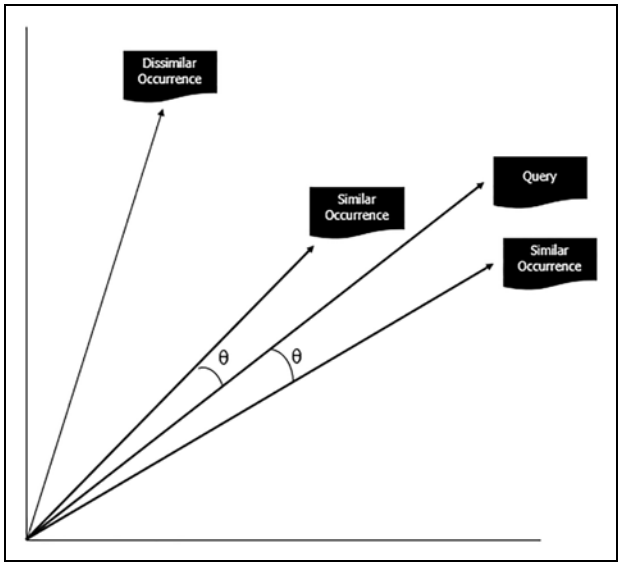

Figure 5. Cosine similarity method.

The model and cosine similarity retriever could be used within a pipeline (Figure 1) so that users could query the incident dataset to obtain extractive answers to questions. The results of which are described in Section 4.

Method 2: Utilising a generative large language model

Generative LLM. The comparative approach demonstrated in this paper utilises an LLM alongside the incident dataset to provide conversational responses to queries that are grounded by real-world reported events.

Rather than ask an LLM specific questions without context, a method called Retrieval Augmented Generation (RAG)[53] was applied to ensure the LLM could base answers upon incident data. This method takes the user's query, to search (and retrieve) relevant reports from the incident data, then provides these with the query to the LLM as a prompt; '*Answer the question based upon [inserted context]*'. The general process is shown in Figure 6.

ChatGPT Version 3.5 was used as the generative LLM for this study. ChatGPT in the context of this work consisted of two models that are collectively referred to as 'ChatGPT'; 'text-embedding-ada-002' was used to create embeddings, that is the encoder model, while 'text-davinci-003' was used for to generate an answer, that is the completion model. ChatGPT adapts a GPT for dialogue, providing a conversational ability with humans. GPTs are a type of LLM that consist of a neural network that uses self-attention to process sequential data. They can typically be fine-tuned for various natural language tasks such as translation or text classification. ChatGPT takes an existing GPT model and rather than training it on even more labelled data it is further trained using human feedback, through reinforcement learning. This helps reduce some of the undesirable outputs such as[54]:

- Hallucination – Where the model makes up responses.
- Bias – Output of toxic responses.

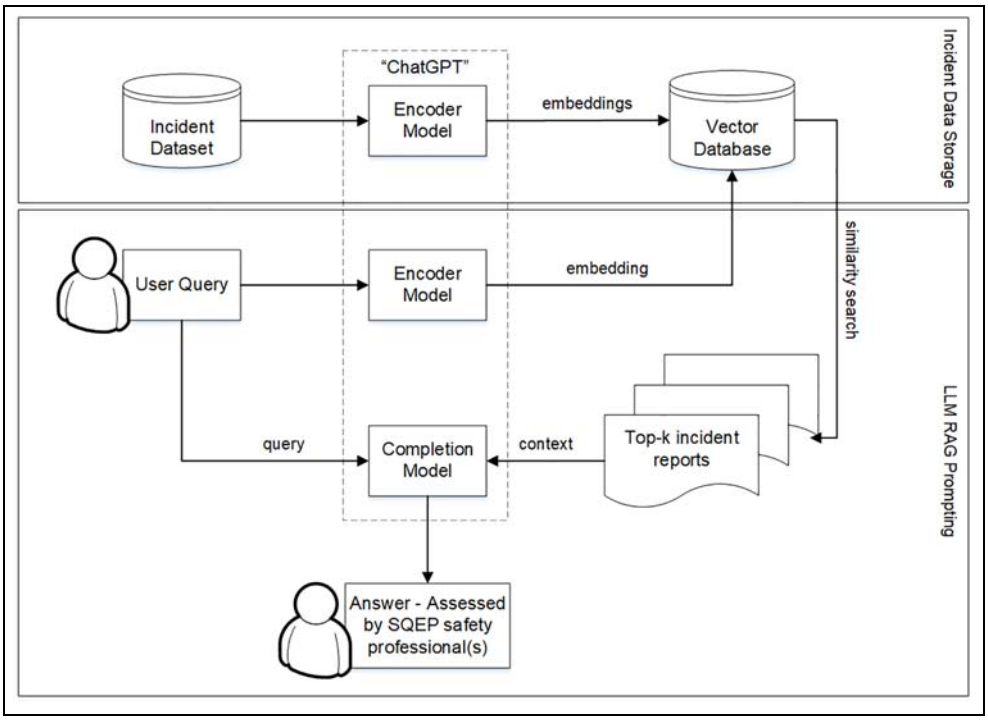

Figure 6. Retrieval Augmented Generation with an incident dataset.

- Poor interpretability – It is unclear how the model arrived at a given response.

Although the RAG process does not entirely mitigate the issues above, it should reduce them through provision of a context.

Vector database. The incident dataset was converted into a vector database to enable efficient RAG responses. A vector is an array of numbers, representing more complex objects such as incident reports in a continuous high dimensional space – forming embeddings that map the semantic features of the reports. Embeddings can then be used in machine learning applications such as searching relevant reports.

The vector database was hosted by Pinecone software (https://www.pinecone.io/) which indexes and stores the report embeddings for similarity searching and fast retrieval. Once each incident report is assigned a vector then nearest neighbours (similar reports) can be easily found against a user query, therefore allowing a semantic search capability revealing knowledge that would otherwise have been difficult to find through traditional lexical (key word) searches.

With the vector database constructed, a pipeline was constructed to take a user query, search the vector database, retrieving relevant reports before feeding these reports (as context) and the original query to the generative LLM to provide an answer.

Results

With the models prepared, it is possible to conduct various assessments to understand their effectiveness and useability.

Extractive Q&A model metric results

A simple method to validate the reader model would be to measure the exact matches for answer spans. However, this is not entirely appropriate as the model can select the 'correct' answer but may have a differing span of words to the recorded answer, in this case the exact match method would register an incorrect match.

An alternative solution is to use Recall-Oriented Understudy for Gisting Evaluation (ROUGE) metrics, or more specifically ROUGE-N which measures the number of matching n-grams between the predicted answer and the reference (true) answer.[55] The 'N' in ROUGE-N refers to the number of tokens/words within a single n-gram (i.e. a sequence of $\times$ words), therefore ROUGE-1 compares individual tokens (unigrams) while ROUGE-2 compares tokens in chunks of two (bigrams). The Rouge score can provide an indication of performance, with a focus on model recall, that is how often n-grams in the reference answers appear in the predicted model answers.

Table 2. Average ROUGE F1 scores on swiftqa dataset.

Model	ROUGE *F1* score
deepset/bert-base-cased-squad2	0.24
bert-swiftqa	0.29

Table 2 records the ROUGE results for the reader model developed in this paper and the more generic 'deepset/bert-base-cased-squad2' as a comparison. This comparator model was selected as it represents a good, generalised BERT reader model being pre-trained on the BookCorpus plus English Wikipedia; then fine-tuned on the SQuAD2.0 dataset.[56] This model achieves an F1 score of 74.67.[57]

The ROUGE scores indicate that the performance of both models is poor, with the bert-swiftqa model having better performance on the SWIFTQA dataset. However, these scores are not reflective of its usefulness – during basic trials the model was able to provide useful answers to questions, especially if it was set to provide the top 5 possible answers. This suggests the adoption of $F1$ score for such a model is too strict. A primary driver to the low scores was the model not predicting the full answer, which is a challenge for the SWIFTQA dataset that predominantly features long, whole sentence answers.

Similar word assessment of model vocabulary. An alternative, qualitative method to assess the model's language understanding is to search for the similar words of given terms. This assessment can be run on both bert-swiftqa and the more generalised deepset/bert-base-cased-squad2 for comparison.

It was expected that similar words of given terms from the vocabulary of bert-swiftqa would return terms related to the incident reporting language, while the generic BERT model would provide more generalised results. This would provide confidence that the developed model has learnt the 'new language' of incident reports.

The querying of similar words cannot be performed by BERT itself as unlike other algorithms such as Word2Vec, BERT does not consist of static vectors but produces them based upon the context. A method for comparing standalone context-independent terms was created, based upon that described by Arefeva et al.[58] Initially pairwise similarities were calculated between all the words in the BERT model vocabulary forming matrix, then a KDTree algorithm (from the python sklearn library) was used to create a search index over the matrix.

A total of 10 key terms were selected based upon their expected associations with other words typically found in incident reports. These terms and the returned 10 similar terms are shown in Tables 3 and 4.

Table 3. Similar words for given terms using deepset/bert-base-cased-squad2.

Term	Similar words
Aircraft	Damaged, propeller, gunfire, critics, survivors, intermediate, knife, catalina, investigation, critic
Damage	Amendments, review, wrecked, packet, fighter, battles, amendment, acts, studies, tire
Fault	Pre, responsible, homeland, review, speech, accordance, broadcasters, environment, candidates, κ
Loose	Ramirez, 212, rub, en, clean, boom, m, mi, carefully, calhoun
Maintenance	Operation, remotely, signalling, diverted, trusts, controllers, operational, mausoleum, shop, staffed
Air	Water, solid, penetrating, met, financing, fallen, reconstructed, choir, abandoned, echo
Pilot	Carleton, breathed, brown, explosion, joyce, sources, paul, professor, steps, screams
Control	Loving, kaufman, bliss, variable, orgasm, controls, fuck, ecstasy, reaper, appetite
Debris	Fruits, drying, fruit, surface, dismantled, disappears, runoff, video, discovery, oven
Ground	Podcast, there, 340, motion, ren, teresa, uncle, location, crow, caesar

Table 4. Similar words for given terms using bert-swiftqa.

Term	Similar words
Aircraft	Tractor, airspace, helicopters, cockpit, reservations, annex, nose, wheels, planes, kidnapping
Damage	Repair, structural, sustained, strike, inspection, paint, damaged, inadequate, compressor, alterations
Fault	Faults, suspect, invest, intermitt, failure, unrelated, problem, ongoing, ##rected, unable
Loose	Securing, sweep, sliding, locking, searches, unse, slight, screw, nose, missing
Maintenance	Servicing, removal, required, requirement, task, rect, procedure, undertaken, procedures, consultation
Air	Airways, aires, airbus, tart, ##aira, ##aires, ##airy, ##airn, ##air, ##aire
Pilot	Flight, abort, rocket, ##raul, raz, pilots, ##raulic, brake, daly, ##ateg
Control	Controlled, controls, restriction, collective, controlling, controller, imagery, activated, activation, licensing
Debris	Sample, evident, chip, damage, leaks, samples, compressor, 1710, detector, fou
Ground	Run, rund, trouble, downs, dirt, shutdown, brush, survival, down, rubin

Through comparison of the similar words returned by each model, it can be determined that the bert-swiftqa model has a better grasp on the language used within incidents. The term 'control' features a clear contrast where the bert-swiftqa vocabulary associates similar words such as 'restriction' and 'collective' which can be inferred in aviation, while the deepset/bert-base-cased-squad2 associates words which have a human-emotive meaning. Likewise, the term 'debris' is clearly associated with words used in the aviation field by bert-swiftqa (e.g. 1710 is a reference to the naval air squadron that perform engineering analysis for UK military aviation).

Evaluation in SWIFT (HAZID) activity

Alongside computational measures of performance, a specific aim of this study was to trial the models in an actual SWIFT activity. This would help establish how useful the model (and associated pipeline) is for safety professionals. It would also assist in understanding how important computational performance metrics are, for example, these could be poor but actual usage might be positive with new knowledge being readily revealed.[38] A qualitative assessment is also required to understand the performance of the generative model approach.

To mitigate the issue of poor metric performance of the extractive model, an element of post-processing code was applied. With the number of returned answers set to 5, the full sentence that contained each answer was returned to the user rather than the single answer itself. These sentences were presented in descending order, with the closest match at the start of the list. The generative model responses were left unaltered. Being that the generative model was controlled by a third-party, a subset of occurrences was used that were anonymized and de-sensitised prior to processing through the model framework.

For the purposes of this paper, the SWIFT analysis considered a fictional scenario, based upon using a specific category RPAS to conduct surveillance operations. This considered the RPAS and its usage covering launch, operation and recovery. A schematic of this scenario is shown in Figure 7 where a laptop ground control station controls the RPAS via a data link, while the RPAS gathers sensing data of the landscape. The functional threat categories and prompts in Table 1 were used to inspire 'what if' queries during the activity.

A sample of both model outputs in relation to user queries is shown in Table 5. While the generative model without the associated RAG process can produce detailed responses to SWIFT queries – this can be openly tried on the ChatGPT Application Programming Interface (API). These responses are generic and change over time. The RAG process steers or grounds the generative model's responses in the truth of previous reported events.

The models were since trialled in various other RPAS related HAZID activities, where the reception is discussed in the following section.

Table 5. Sample of query results.

User query	Extractive model answers	Generative model answer
How can rodents damage the GCS?	Answer: rodents chewing ACU pipes. Answer: ATOLS Fibre Optic cables have proved to be extremely susceptible to damage caused by rabbits and rodents as it lacks any form of protection. Answer: Thus GCS's can not be placed fully under the dutch barns. Answer: Upon going back to see whether there was an issue, someone had reported to them that we had hit the concrete pillar. Answer: It is undetermined as to how exactly the rat gained access into the GCS however measures have been taken to reduce the probability of it happening again.	Rodents can potentially damage a UAV Ground Control Station (GCS) in several ways, based on the context provided: Chewing on Cables and Wires: The reference to rodents chewing ACU pipes indicates that rodents in the area have a propensity for chewing on various materials. This behaviour could extend to cables and wires connected to the GCS. Rodents may chew through these cables, causing electrical shorts or other damage that could disrupt communication or control signals between the GCS and the UAV. Nesting in Equipment: Rodents may also seek shelter within the GCS itself, particularly if it is located in an environment where they are prevalent. Nesting within the GCS could lead to damage caused by nesting materials or by the rodents themselves as they navigate and explore the equipment. Contamination and Corrosion: Rodents can leave behind urine, faeces and other contaminants that may lead to corrosion or other forms of damage to sensitive electronic components within the GCS. Over time, this could degrade the performance and reliability of the equipment. Interference with Cooling Systems: If the GCS includes cooling systems or ventilation ducts, rodents may block or damage these systems by nesting or by chewing on components. This could lead to overheating or other issues that affect the operation of the GCS. In the context provided, while the specific damage observed on the arrestor cable may not definitively point to rodent activity, the high rodent numbers in the area suggest that rodents could be a potential cause. Given their propensity for chewing on materials and the absence of other obvious sources of damage, it's reasonable to consider rodents as a potential factor in the observed damage to the arrestor cable. Additionally, the mention of problems with rodents chewing on ACU pipes underscores the likelihood of rodent activity in the area.
What if the RPAS has a lost link?	Answer: The RPAS was launched and raised to approx 3 m, once in the hover it seemed to become unstable. Answer: At this point it was noted by not only the MO/VO but also the students on the course that the RPAS was losing altitude rather than maintaining altitude as commanded. Answer: The RPAS was recovered with no visible damage and found upright in the snow alongside it's wings and stabilator. Answer: This is also the same location and grid were the RPAS was recovered, 4.1 km from my location. Answer: - On two separate flights, the UAV-p1 forgot to change the uploaded Lost Link Route to the circuit Lost Link Route having joined the circuit after an unplanned (automatic) ATOL Abort.	In the scenario you provided, if an RPAS (Remotely Piloted Aircraft System) experiences a lost link, it can lead to a series of critical events and potentially hazardous situations. Let's break down the implications of a lost link in both instances described in your scenario: RPAS Incident #1: The lost link occurred during flight, leading to the RPAS losing altitude rapidly. Attempts were made to regain control, including switching to manual mode and attempting an auto-land maneuver, but they were unsuccessful due to the rapid loss of altitude and system delays. The RPAS ultimately conducted an uncontrolled landing, resulting in an unintended landing site. RPAS Incident #2: The lost link occurred while the RPAS was flying through broken cloud cover. The vehicle operator redirected the RPAS to a known waypoint to reestablish the link. Despite attempts to regain control and maintain altitude, the RPAS continued to descend and eventually conducted an uncontrolled landing. In both cases, the lost link significantly affected the ability to control the RPAS, leading to potential safety hazards such as uncontrolled descents and unintended landings. Additionally, in incident #2, flying through cloud cover

(continued)

Table 5. Continued

User query	Extractive model answers	Generative model answer
		compounded the issue, causing moisture to gather within the pitot tube system and further impacting the aircraft's performance. To address lost link incidents and mitigate their potential risks, it's crucial to: Ensure robust communication systems and protocols are in place to minimize the occurrence of lost links. Implement fail-safe mechanisms such as autonomous return-to-home procedures or emergency landing protocols to safely manage the RPAS in case of lost links. Enhance pilot training and awareness regarding lost link scenarios and their appropriate responses. Continuously monitor and evaluate environmental conditions such as cloud cover to anticipate potential challenges to communication and aircraft performance. These measures can help mitigate the impact of lost link incidents and enhance the safety and reliability of RPAS operations.
How can the RPAS cause injury to the operator at launch?	Answer: The individual injured within this DASOR was not trained on any form of RPAS and had only been given minimal instruction of how to launch the RPAS by and AVUAS MO/VO Operator. Answer: Flight through cloud caused moisture to gather within the pitot tube system. Answer: The RPAS incurred minimal damage. Answer: Event – During the attempted vehicle borne hand launch (VL) of a mini unmanned air system (MUAS), the launch operator fell from the rear compartment of the moving vehicle and sustained a major physical injury. Answer: injury sustain to left leading hand Index finger due to no PPE being warn on hand.	The RPAS (Remotely Piloted Aircraft System) can cause injury to the operator in several ways, as evidenced by the scenarios described: Propeller Strike: During launch or operation, if the operator inadvertently comes into contact with the rotating propellers, it can cause serious injury. In the given scenario, the operator's left index finger was struck by the propellers when attempting to catch the falling aircraft. This resulted in a portion of the skin being amputated. Lack of Proper Personal Protective Equipment (PPE): The operator's injury was exacerbated by the absence of proper PPE. In this case, the operator was not wearing a compliant glove, which could have mitigated or prevented the injury to some extent. Unforeseen Circumstances: Stumbling or unexpected movements during the launch process can lead to accidents. In the scenario provided, the operator stumbled just before releasing the RPAS, resulting in an injury to the index finger. Operating Environment Hazards: Presence of obstacles or vehicles in the launch area can pose a risk during operation. Although the operator ensured the launch area was clear before proceeding, the presence of a civilian engineer vehicle near the launch site could have contributed to the accident. To mitigate such risks and prevent injuries to operators, it's essential to: Ensure proper training for operators in handling and launching RPAS. Enforce the use of appropriate PPE, including gloves, to protect against propeller strikes and other hazards. Conduct thorough pre-flight checks to identify and mitigate potential risks in the operating environment. Maintain situational awareness and vigilance during operation to respond effectively to unforeseen circumstances. Establish clear procedures for communication and coordination among team members to ensure safe operation of RPAS. Continuously review and improve safety protocols based on lessons learned from incidents or accidents

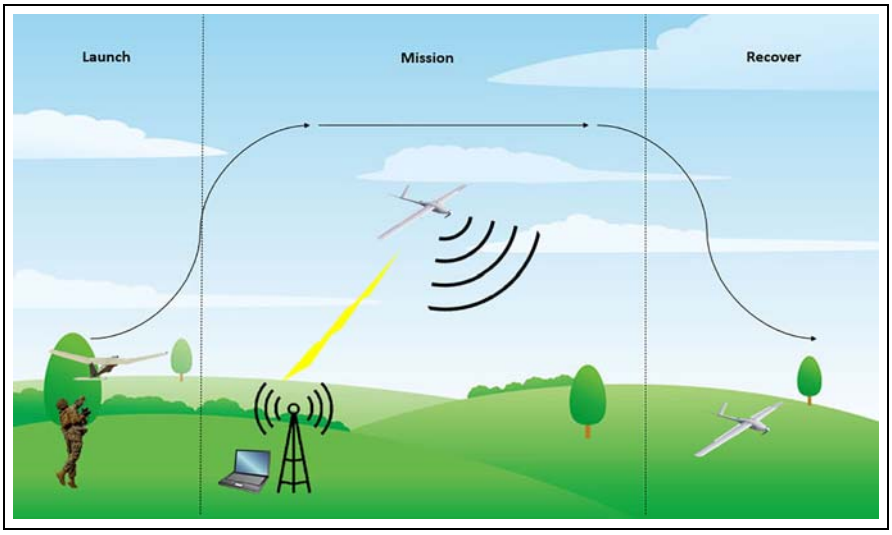

Figure 7. Fictional RPAS scenario for the SWIFT activity.

Discussion

Applicability for HAZID

The evaluation trial results show that either the extractive or generative model was able to return answers to queries and fulfil the aim set out by the paper. Although the rationale for the extractive model returning multiple answers was originally to alleviate issues with returning single incorrect answers, this ultimately proved more useful from a safety perspective as multiple causes, consequences and hazards could be revealed rather than just one extracted answer. This is an important aspect of HAZID where multiple options can be revealed and explored to reveal risks[28] and appeals to the brain-storming nature of SWIFT.

Queries entered into the generative model returned detailed responses that elaborated on the core answer, thus providing additional detail that could be useful in a HAZID. In some cases, even suggesting appropriate mitigation. Alternatively, the extractive method literally returned the closest matching sentence, and the associated incident report would need to be read if more understanding was required – a task that would add time to a HAZID activity, that can already be a time-consuming process. A benefit to the extractive model is its repeatability where the same sentence would be returned as the answer if the query was re-run, unlike the generative response. This can be preferable if seeking an auditable trail to decisions within the HAZID.

The overall concept perhaps works best for revealing nuances to posed questions. For example, the question asking how rodents can damage the ground control station revealed that fibre optic cables were particularly susceptible to rabbits and rodents. A feature that was not apparent to those conducting the exemplar HAZID described in Section 4.2. Therefore, the model goes someway to revealing rarer issues and 'black swan' events that would otherwise typically be hidden by generic incident classifications within a database.

Extractive versus generative: Which is better for HAZID?

A key aspect of this paper was to understand whether an extractive or generative model would be better for assisting a HAZID. The key advantages and disadvantages are shown in Table 6.

Both approaches are feasible, the selection in a HAZID mainly comes down to the interpretability and traceability of the required answers. The extractive results represented segments of actual incident reports and therefore represented events that had happened. However, from the trial several non-sensical answers were returned alongside good answers – therefore an element of human processing is required to filter out these answers.

Alternatively, the generative model provided a fluent response that was easy to read and could be elaborated upon further if treating the LLM as a chatbot. Where the extractive model required extensive effort to create – mainly in developing a bespoke dataset that required safety engineering knowledge. The RAG process using an LLM was far simpler to setup and deploy.

The method used in this paper to create the extractive model demonstrates that it is possible for organisations to create their own language models using open-source software and datasets, combined with their own data. Where this paper focussed upon Q&A, a bespoke BERT model could also be used for classification and entity recognition tasks, which both have roles within the processing of incident data with respect to managing safety.[38] It is also likely that a specialized BERT model would perform better in these areas in comparison to a generative LLM.

Overall, it is the authors opinion and those who participated in the evaluation that the generative method described within this paper is preferable for a HAZID activity. An exception would be if the incident data was highly nuanced (e.g. many bespoke terms).

Table 6. Advantages and disadvantages of extractive and generative models for HAZID.

Method	Advantages	Disadvantages
Method 1 – Extractive Q&A	BERT is trained on the incident corpus of text, which helps it understand the contextual relationship between words and sentences. It also considers the text bidirectionally (from both the left and right) which allows it to effectively capture the complex linguistic patterns.	BERT's performance heavily depends on the quality and representation of the training data (i.e. the incident dataset). It may struggle with out-of-domain or uncommon questions if not adequately trained on diverse datasets – this is where pre-processing of new data is paramount.
	BERT has demonstrated high performance on various natural language understanding tasks, including question answering.[8,18,46] Resources such as tutorials and support can be easily acquired online.	BERT is a feature-based model that extracts answers from a given text; it cannot generate new responses or provide explanations beyond what is contained in the input data. Hence, it can be difficult for HAZID participants to interpret some answers.
	BERT operates through masked language modelling and bidirectional representations, which allows for more interpretable analysis of how the model arrives at its predictions. This can be important in safety systems where traceability or rationale to decisions is required.	As shown in the results, non-sensical answers are returned alongside 'good' answers. These then require filtering out by safety professionals within the HAZID.
	Depending on the computational resources available, using a smaller model like BERT might be more feasible in terms of memory and computational requirements compared to generative LLMs which typically have more parameters.	
	Data remains secure as BERT can be trained, stored and run on a local system with no internet connection/third party resources.	
Method 2 – Generative Q&A	Generative models such as ChatGPT can generate responses to questions that may not be directly present in the incident dataset by leveraging its generative capabilities, allowing for more diverse and creative answers.	There is a potential for generating inaccurate or irrelevant responses, especially when the input question is complex or ambiguous. Therefore, care must be taken in crafting appropriate prompts for the LLM.
	By combining the RAG method with generation, ChatGPT can leverage the strengths of both approaches, ensuring accurate answers while also providing additional context or explanations when necessary.	If a generative LLM was to be stored and operated on a local system rather than leveraging an API then this would require more computational resources compared to using a pre-trained model like BERT.
	Through RAG, responses are generated based on its understanding of the input question and context, enabling it to provide explanations or reasoning for its answers, therefore enhancing interpretability and rationale to answers.	Being that many generative LLMs are third party owned and operated, the inputted data needs to be screened to de-sensitise information or legal agreements need to be in place.
	The RAG method does not require resource intensive training or fine-tuning of LLMs.	

Limitations

Pre-processing of incident data could improve answer accuracy, although BERT is capable of learning that an 'RPAS' is an 'air system', the limited amount of data may hinder this ability. One solution could be to perform some basic pre-processing of the incident text such as replacing a variety of terms that have the same meaning with one, common term. For example, 'air system', 'UA', 'UAV', 'UAS', 'unmanned air system', etc. could all be replaced with 'RPAS'. This pre-processing method has been previously undertaken by Ricketts et al.[59] where care must be taken to account for the surrounding context so that meaning is not lost. Likewise, the Q&A dataset would need the same processing to ensure it remains representative of the data.

It was suggested that use of the models could create a bias for reading the incident dataset rather than considering participants questions and 'blue-sky' thinking. To counter this, it may be prudent to use the models after participants have finished their own questioning for each category or only if there is a disagreement on whether particular scenarios are likely or not – hence the model augments the HAZID. Further opinions and suggested improvements could be gained from the continued use of the model in further HAZID activities.

A further limitation within this work was the small size of the developed SWIFTQA dataset, a feature due to the resource simply not being available to generate more question-answer pairs. Going forward it would be interesting to increase the size of SWIFTQA and understand if this improves the performance of the reader model. Alternatively, the creation of an artificial dataset could be explored to generate context and question-answer pairs quickly. However, it is the authors view that this would be just as labour intensive to validate such a dataset ensuring it is representative of incident reports and typical questions presented in HAZIDs.

It is the author's opinion that model metrics such as ROUGE scores were not particularly useful for the extractive model (bert-swiftqa) developed in this paper. The typically long 'true' answer is difficult for the model to predict and for many queries, the model will extract a legible extract regardless. This can then be easily interpreted as useful or not by the safety professionals in the HAZID activity. It is suggested that models should be assessed by human subject matter experts, firstly assessing the model vocabulary and then deploying the model with set scenarios to understand its performance and limitations. A further mitigation is the users of the model being suitable qualified and experienced, and therefore able to recognise nuisance answers while using the model as an aide.

A further finding was that the structure and content of the posed questions needed to be carefully considered when using the extractive model. If more key terms and descriptive terms could be included within a question, then this would generally result in better, more similar retrieved incident reports and therefore better answers. It was proposed that a standard question layout could be used to clearly bound the scope and assist the model in extracting or generating better answers.

Use of the generative model, although powerful, comes with some major limitations. Firstly, data privacy must be acknowledged where not all datasets can be readily processed through a third-party model (the subset used in this paper was completely anonymized and screened prior to use). This can only be overcome through non-disclosure agreements and transparency in how the data is processed. Secondly, the model is controlled by a third-party, hence can be modified and updated without knowledge. During production of this paper ChatGPT has progressed from Version 3.5 to 4 and 4o. This leads to new behaviour and a lack of repeatability or rationale for previous answers. To overcome this, an offline copy of the model would need to be stored by the HAZID organisation – potentially a costly requirement in both terms of finance and computing power.

Future work

An area of future work could consider a model that can interpret chains of events, for example, cause-hazard-consequence for greater integration to hazard analysis such has Bowties. This could be based upon the concept described by Tandon et al.[60] who developed a dataset of 'What if' questions over procedural text. For hazard identification purposes, incident reports (which contain greater detail and variance) may have to be processed into a particular format before such a dataset can be constructed.

The aim of the model developed in this paper was to extract knowledge – as identified in Section 2.3, a key element of risk analysis is probability. Future work would benefit from exploring how probability could be presented alongside extracted answers where this would benefit common risk analysis processes that are commonly concerned with severity, frequency matrices. It is likely that more, reliable data would be required for this. For example, the study of RPAS systems would benefit from the inclusion of flight hours, flight data and possibly maintenance records to reveal failure rates.

Where this paper focussed upon incident reports, future work could include further sources of information such as existing safety assessment reports, hazard analysis, maintenance data, etc which together, could build a more diverse knowledge base for a given system. Such a knowledge repository has the potential to save resources when it comes to unwittingly repeating hazard identification, for example, a separate project team may be required to conduct a HAZID very similar to a previous, separate one. Through using a model trained on wider data, it would be possible to find and query the previous HAZID.

Future iterations of generative LLMs should also seek to improve interpretability and reduce hallucinations (this is acknowledged to be a wider issue). If this can be achieved, then greater trust can be gained from users of the model while clear rationale to responses can be demonstrated – which is ideal for safety decision making.

Conclusion

This paper has brought together two distinct fields of research: that of NLP question and answering systems and hazard identification – a subset of safety engineering. The paper introduces a bespoke BERT model trained upon incident data and fine-tuned for unique extractive Q&A within HAZID alongside a generative LLM augmented by a RAG process. These proved to be a useful addition to the overall HAZID process, allowing participants to effectively 'Google' answers to safety-themed questions posed to the incident dataset. As far as the authors are aware, this has not been completed before.

It is suggested that utilising a generative LLM is preferable as the user can take advantage of the models expansive training and conversational responses while providing context from the incident dataset. The exception to this being incident datasets that are heavily nuanced and effectively form their 'own language' or contain restricted data.

A greater challenge beyond this paper is the ability for machine learning models to consider cause and effect in the context of a given system's operation. Future work should explore the wider progress on this topic and understand if further data sources can be incorporated such as previous hazard analysis reports, maintenance data and even technical reports.

Acknowledgements

J Ricketts thanks the contribution of the IMechE Whitworth Senior Scholarship Award, the Royal Air Force and BAE Systems in supporting this research.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article.

ORCID iD

Jon Ricketts https://orcid.org/0000-0001-9487-9092

References

1. Leveson N. A new accident model for engineering safer systems. *Saf Sci* 2004; 42: 237–270.
2. Stemn E, Bofinger C, Cliff D, et al. Failure to learn from safety incidents: status, challenges and opportunities. *Saf Sci* 2018; 101: 313–325.
3. Kletz TA. Searchlights from the past. *J Hazard Mater* 2008; 159: 130–134.
4. Drupsteen L and Hasle P. Why do organizations not learn from incidents? Bottlenecks, causes and conditions for a failure to effectively learn. *Accid Anal Prev* 2014; 72: 351–358.
5. Ericson C. *Hazard analysis techniques for system safety*. 2nd ed. Hoboken, NJ: Wiley, 2015.
6. Leveson N. *Engineering a safer world*. Cambridge, MA: The MIT Press, 2016.
7. Johnson CW. *A handbook of incident and accident reporting*. Glasgow: Glasgow University Press, 2003.
8. Kierszbaum S and Lapasset L. Applying distilled BERT for question answering on ASRS reports. In: *Proceedings 22nd International Conference on New Trends in Civil Aviation 2020*, NTCA2020, Prague, Czech Republic, 2020, pp.33–38.
9. Qi Y, Zhao X and Huang X. Safety analysis in the era of large language models: a case study of STPA using ChatGPT. arXiv, 2023; 1–26.
10. Chatterjee J and Dethlefs N. This new conversational AI model can be your friend, philosopher, and guide. and even your worst enemy. *Patterns* 2023; 4: 1–3.
11. Jidkov V, Abielmona R, Teske A, et al. Enabling maritime risk assessment using natural language processing-based deep learning techniques. In: *2020 IEEE symposium series on computational intelligence, SSCI 2020*, Canberra, ACT, Australia, 2020, pp.2469–2476.
12. Baker H, Hallowell MR and Tixier AJP. Automatically learning construction injury precursors from text. *Autom Constr* 2020; 118: 103145.
13. De Vries V. Classification of aviation safety reports using machine learning. In: *2020 International conference on artificial intelligence and data analytics for air transportation, AIDA-AT 2020*, Singapore, 3–4 February 2020. New York, NY: IEEE.
14. Zhang F, Fleyeh H, Wang X, et al. Construction site accident analysis using text mining and natural language processing techniques. *Autom Constr* 2019; 99: 238–248.
15. Simmons RF. Answering English questions by computer: a survey. *Commun ACM* 1965; 8: 53–70.
16. Clark K, Khandelwal U, Levy O, et al. What Does BERT Look at? An Analysis of BERT's Attention. In: *Proceedings of the second Blackbox NLP workshop on analyzing and interpreting neural networks for NLP*, Florence Italy, Association for Computational Linguistics, Kerrville, 2019, pp.276–286.
17. Zhao WX, Zhou K, Li J, et al. A survey of large language models. *arXiv Prepr* 2023; 1–51.
18. Rajpurkar P, Zhang J, Lopyrev K, et al. SQuad: 100,000 + questions for machine comprehension of text. In: *Proceedings of the conference on empirical methods natural language processing*, Austin Texas, Association for Computational Linguistics, Kerrville, 2016, pp.2383–2392.
19. Kwiatkowski T, Palomaki J, Redfield O, et al. Natural questions: a benchmark for question answering research. *Trans Assoc Comput Linguist* 2019; 7: 453–466.
20. Choi E, He H, Iyyer M, et al. QUAC: question answering in context. In: *Proceedings of the 2018 Conference on Empirical Methods Natural Language Processing*, Brussels Belgium, Association for Computational Linguistics, Kerrville, 2018, pp.2174–2184.
21. Reddy S, Chen D and Manning CD. CoQA: a conversational question answering challenge. *Trans Assoc Comput Linguist* 2019; 7: 249–266.
22. Fan A, Jernite Y, Perez E, et al. ELi5: long form question answering. In: *Proceedings of the 57th annual meeting of the assocication for computational linguistics*, Florence Italy, Association for Computational Linguistics, Kerrville, 2020, pp.3558–3567.
23. Nguyen T, Rosenberg M, Song X, et al. MS MARCO: a human generated MAchine reading COmprehension dataset. *CEUR Workshop Proc* 2016; 1773: 1–11.
24. Xiong W, Wu J, Wang H, et al. TweetQa: a social media focused question answering dataset. In: *Proceedings of the 57th annual meeting of the assocication for computational linguistics*, Florence Italy, Association for Computational Linguistics, Kerrville, 2020; pp.5020–5031.
25. Peters ME, Ruder S and Smith NA. To tune or not to tune? adapting pretrained representations to diverse tasks. In: *Proceedings of the 4th workshop on represent learning of NLP, RepL4NLP* 2019, Florence Italy, Association for Computational Linguistics, Kerrville, 2019, pp.7–14.
26. Liu P, Yuan W, Fu J, et al. Pre-train, prompt, and predict: a systematic survey of prompting methods in natural language processing. *ACM Comput Surv* 2023; 55: 1–46.
27. International Organization for Standardization. *ISO 31010 Risk management – Risk assessment techniques*, 2019.
28. Crawley F and Tyler B. *Hazard Identification methods*. Institution of Chemical Engineers, Rugby, Warwickshire, 2003.
29. Defence Equipment & Support. SWIFT. *Acquisition safety and environmental management system*, https://www.asems.mod.uk/toolkit/swift (2023, accessed 24 February 2023).
30. Aven T and Thekdi S. *Risk science an introduction*. New York: Routledge, 2022.

31. Apostolakis G. The concept of probability in safety assessments of technological systems. *Science* 1990; 250: 1359–1364.
32. Aven T. Practical implications of the new risk perspectives. *Reliab Eng Syst Saf* 2013; 115: 136–145.
33. Kay J. *The long and the short of it*. London: The Erasmus Press, 2009.
34. Military Aviation Authority. *ASIMS user manual*, 2020.
35. Feng X, Dai Y, Ji X, et al. Application of natural language processing in HAZOP reports. *Process Saf Environ Prot* 2021; 155: 41–48.
36. Jing X, Chennakesavan A, Chandra C, et al. BERT for aviation text classification. AIAA 2023-3438. AIAA AVIATION 2023 Forum, 2023; 1–15.
37. Soleimani A, Monz C and Worring M. BERT for evidence retrieval. In: *Advances in information retrieval. ECIR 2020*, pp.359–366. Cham: Springer International Publishing.
38. Ricketts J, Barry D, Guo W, et al. A scoping literature review of natural language processing application to safety occurrence reports. *Safety* 2023; 9: 22.
39. Devlin J, Chang MW, Lee K, et al. BERT: pre-training of deep bidirectional transformers for language understanding. In: *NAACL HLT 2019 - 2019 conference of the north american chapter of the association for computational linguistics: human language technologies - proceedings of the conference*, Minneapolis, MN, Association for Computational Linguistics, Kerrville, 2019, pp.4171–4186.
40. Wu Y, Schuster M, Chen Z, et al. Google's neural machine translation system: bridging the gap between human and machine translation. ArXiv abs/1609.08144, 2016; 1–23.
41. Pietsch M, Soni T, Chan B, et al. Haystack, https://github.com/deepset-ai/haystack/ (2020, accessed 2 October 2023).
42. Haystack. Annotation Tool, https://docs.haystack.deepset.ai/docs/annotation (2023, accessed 12 May 2023).
43. Ministry of Defence. *DEF STAN 00-56: PART 2 safety management requirements for defence systems part 2: guidance*, 1996.
44. Wikimedia Foundation. Wikimedia Downloads, https://dumps.wikimedia.org (2022, accessed 22 July 2023).
45. Huggingface. BertConfig, https://huggingface.co/docs/transformers/model_doc/bert#transformers.BertConfig (2023, accessed 27 July 2023).
46. Moller T, Risch J, Pietsch M, et al. Deepset/bert-base-uncased for QA, https://huggingface.co/deepset/bert-base-uncased-squad2 (2023, accessed 10 August 2023).
47. Zhang Y and Xu Z. BERT for Question Answering on SQuAD 2.0. *Stanf Univ Stud J* 2019; 9. https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1194/reports/default/15848021.pdf
48. Robinson SD. Visual representation of safety narratives. *Saf Sci* 2016; 88: 123–128.
49. Gammack J, Akay H, Ceylan C, et al. Semantic knowledge management system for design documentation with heterogeneous data using machine learning. *Procedia CIRP* 2022; 109: 95–100.
50. Fan H and Li H. Retrieving similar cases for alternative dispute resolution in construction accidents using text mining techniques. *Autom Constr* 2013; 34: 85–91.
51. Taipalus T. Vector database management systems: fundamental concepts, use-cases, and current challenges. Cogn Syst Res, 2024; 85: 101216, http://arxiv.org/abs/2309.11322 (2023).
52. Jones KS. A statistical interpretation of term specificity and its application in retrieval. *J Doc* 1972; 28: 11–21.
53. Lewis P, Perez E, Piktus A, et al. Retrieval-augmented generation for knowledge-intensive NLP tasks. *Adv Neural Inf Process Syst* 2020: 9459–9474.
54. OpenAI. ChatGPT: optimizing language models for dialogue, https://openai.com/blog/chatgpt/ (accessed 10 February 2023).
55. Lin C-Y. ROUGE: a package for automatic evaluation of summaries arteries. In: *Text summarization branches out*, Barcelona, Kerrville, 2004, pp.74–81. Association for Computational Linguistics.
56. Rajpurkar P, Jia R and Liang P. Know what you don't know: unanswerable questions for SQuAD. *Proceedings of the 56th annual meeting of the assocication for computational linguistics*, 2018, 2, pp.784–789.
57. Deepset. deepset/bert-base-cased-squad2, https://huggingface.co/deepset/bert-base-cased-squad2 (2022, accessed 2 October 2023).
58. Arefeva V and Egger R. When BERT started traveling: TourBERT—a natural language processing model for the travel industry. *Digital* 2022; 2: 546–559.
59. Ricketts J, Pelham J, Barry D, et al. An NLP framework for extracting causes, consequences, and hazards from occurrence reports to validate a HAZOP study. In: *2022 IEEE/AIAA 41st digital avionics systems conference (DASC)*, Portsmouth, VA, 2022, pp.1–8. New York, NY: IEEE.
60. Tandon N, Mishra BD, Sakaguchi K, et al. Wiqa: a dataset for 'What if...' reasoning over procedural text. In *Proceedings of the 9th conference on empirical methods natural language*, Hong Kong, China, Association for Computational Linguistics, 2019, pp.6076–6085.

Integrating an incident dataset with a question and answering language model to assist hazard identification: comparison of an extractive and generative model

Ricketts, Jon

2024

Ricketts J, Guo W, Pelham J, Barry D. (2024) Integrating an incident dataset with a question and answering language model to assist hazard identification: comparison of an extractive and generative model. Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability, Available online 29 September 2024

https://doi.org/10.1177/1748006x241272831