INFORMS Journal on Data Science

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Hierarchical Multilabel Classification for Fine-Level Event Extraction from Aviation Accident Reports

Xinyu Zhao; , Hao Yan; , Yongming Liu

To cite this article:
Xinyu Zhao; , Hao Yan; , Yongming Liu (2025) Hierarchical Multilabel Classification for Fine-Level Event Extraction from Aviation Accident Reports. INFORMS Journal on Data Science 4(1):51-66. https://doi.org/10.1287/ijds.2022.0032

https://pubsonline.informs.org/journal/ijds

INFORMS JOURNAL ON DATA SCIENCE
Vol. 4, No. 1, January–March 2025, pp. 51–66
ISSN 2694-4022 (print), ISSN 2694-4030 (online)

Hierarchical Multilabel Classification for Fine-Level Event Extraction from Aviation Accident Reports

Xinyu Zhao,[a] Hao Yan,[a,*] Yongming Liu[b]

[a] School of Computing and Augmented Intelligence, Arizona State University, Tempe, Arizona 85287; [b] School for Engineering of Matter, Transport and Energy, Arizona State University, Tempe, Arizona 85287

*Corresponding author

Contact: xzhao119@asu.edu, https://orcid.org/0000-0002-3296-9495 (XZ); haoyan@asu.edu, https://orcid.org/0000-0002-4322-7323 (HY); Yongming.Liu@asu.edu (YL)

Received: November 4, 2022
Revised: May 31, 2023
Accepted: March 7, 2024
Published Online in Articles in Advance: October 17, 2024

https://doi.org/10.1287/ijds.2022.0032

Abstract. Large numbers of accident reports are recorded in the aviation domain, which greatly values improving aviation safety. To better use those reports, we must understand the most important events or impact factors according to the accident reports. However, the increasing number of accident reports requires large efforts from domain experts to label those reports. To make the labeling process more efficient, many researchers have started developing algorithms to automatically identify the underlying events from accident reports. This article argues that we can identify the events more accurately by leveraging the event taxonomy. More specifically, we consider the problem to be a hierarchical classification task, where we first identify the coarse-level information and then predict the fine-level information. We achieve this hierarchical classification process by incorporating a novel hierarchical attention module into the bidirectional encoder representations from transformers model. To further utilize the information from event taxonomy, we regularize the proposed model according to the relationship and distribution among labels. The effectiveness of our framework is evaluated using data collected by the National Transportation Safety Board. It has been shown that fine-level prediction accuracy is highly improved and that the regularization term can be beneficial to the rare event identification problem.

History: Kwok-Leung Tsui served as the senior editor for this article.
Funding: The research reported in this paper was supported by funds from NASA University Leadership Initiative program (Contract No. NNX17AJ86A, Project Officer: Dr. Anupa Bajwa, Principal Investigator: Dr. Yongming Liu) and NSF DMS 1830363.
Data Ethics & Reproducibility Note: The code capsule is available on Code Ocean at https://codeocean.com/capsule/9128124/tree/v1 and in the e-Companion to this article (available at https://doi.org/10.1287/ijds.2022.0032).

Keywords: hierarchical classification • accident reports analysis • BERT • rare event identification

1. Introduction

In air traffic management, identifying safety risk and supporting safety improvements to mitigate risk are of great importance for the next generation of national air transportation systems. The National Transportation Safety Board (NTSB) has investigated and collected more than 60,000 aviation accident and incident reports with labeled sequences of accident events. It is of great importance to improve aviation safety by preventing repeat accidents. The engineers at NTSB have spent a lot of time recording past aviation accidents and understanding the crucial causes of past accidents. Because the accident is usually recorded in raw text format, the root cause analysis requires experts with domain understanding. However, as more and more accident data are collected, such an analysis strategy becomes very inefficient.

With the development of natural language processing (NLP) techniques, text mining methods have become popular in traffic management. For example, Yao and Qian (2021) proposes to analyze Twitter messages using text mining to predict the traffic on the next day. Rath and Chow (2022) proposes to utilize the state-of-the-art NLP model to use Wikipedia data to predict the typology of city transport. More specifically, people started to design more efficient text mining tools to automatically identify all accident events from the accident report using historical accident report data with identified critical events. When new accident reports are available, the underlying events can be automatically identified using powerful supervised learning methods. There are many existing works on the analysis of incident reports. For example, Pereira et al. (2013) proposes using traffic incident record messages to predict the duration of the incident. The knowledge graph is also used to reveal the relationship among critical components in ship collision accidents

(Gan et al. 2023). Many researchers have started bringing NLP into the aviation domain. Yelundur et al. (2016) proposed to apply Bayesian logistic regression to classify four major events, namely loss of separation, deviation and Air Traffic Control anomalies, ground and landing events, and loss of control. A question and answer system is developed for the aviation domain with graph knowledge in Agarwal et al. (2022). Abedin et al. (2010) applies weakly supervised semantic lexicon construction to identify causes from aviation incident reports. Similarly, Robinson (2018) focuses on cause identification task as well from a multilabel classification perspective. However, because of the limited capacity of these traditional machine learning models, they were unable to utilize a large number of accident reports, and they were unable to model the contextual relationship between words.

In recent years, deep learning models have been proposed to analyze the data from the aviation accident report. Dong et al. (2021) proposed attention-based long short-term memory (LSTM) to identify the six most frequent impact factors from incident reports recorded by the Aviation Safety Reporting System (ASRS). Another work based on ASRS data proposed identifying the 14 most important cause types, such as physical factors, physical environment, proficiency, etc. These works have achieved satisfactory results on a limited number of factors (Abedin et al. 2010). Zhao et al. (2022) has proposed another method to identify 58 main causes in the NTSB accident report using the LSTM model. Recently, the bidirectional encoder representations from transformers (BERT) model has been proposed in the natural language processing domain. The BERT model has also been applied to accident report modeling. For example, a BERT-based question-answering system has been proposed based on the ASRS data set (Kierszbaum and Lapasset 2020). However, the major limitation of existing NLP models on the aviation accident report data is that we can only identify coarse categories from reports. Existing works still require experts' effort to specify the details within those categories. For example, it is very challenging for algorithms to tell what the exact physical factors of accidents are, which is an important task in aviation accident modeling. Zhang and Mahadevan (2020) applied the Bayesian network to the NTSB data set to study the causal and dependent relationships among a wide variety of detailed contributory factors. Rao and Marais (2020) mentioned that there exists inconsistent coding and short chain lengths associated with general aviation accidents in NTSB: to be specific, a great variety of chains (152 different occurrence chains), with more than 82% of the accidents having an average chain length of two or less. By consistently identifying fine categories from reports, we will compensate for the missing logic in an accident event chain. Finally, Tanguy et al. (2016) also provides evidence that the existing taxonomy categories are generally too broad to identify a specific characteristic of an event in ASRS.

There are four major challenges for identifying the fine categories from the accident reports. We will use accident reports from the National Transportation Safety Board as an example to illustrate the challenges as follows.

1. Multiple accident events for each accident report. In general, there can be multiple accident events corresponding to each accident report. The number of accident events in each report is also typically unknown.

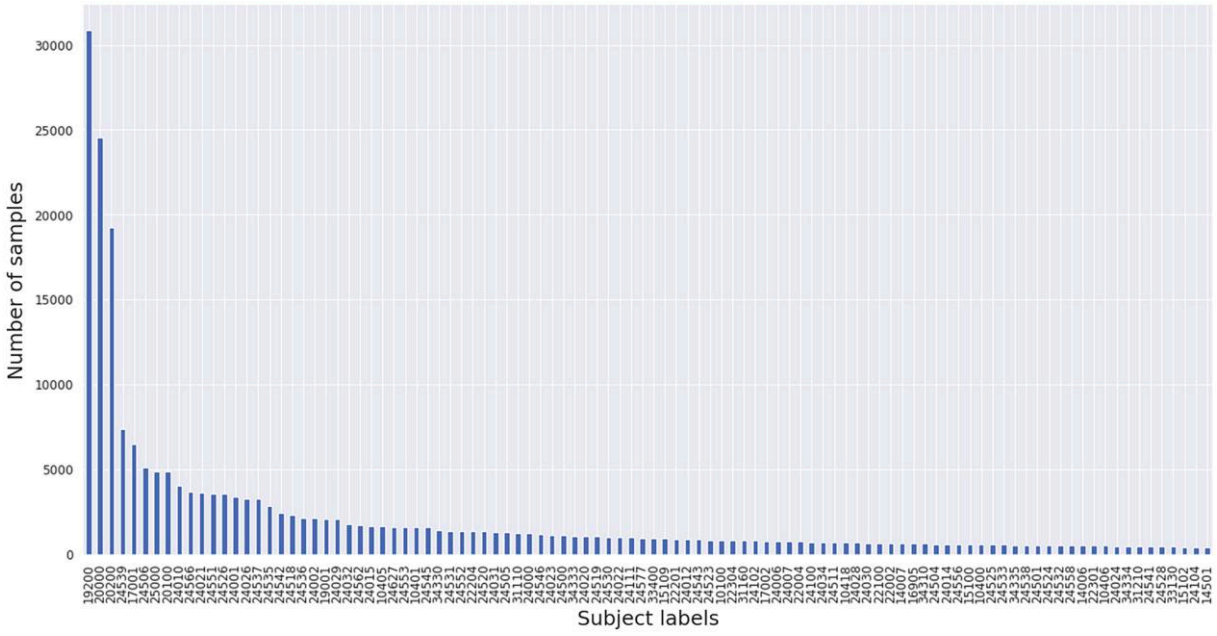
2. Complex correspondent. Many occurrence codes or even subject codes correspond to multiple sentences from the original aviation accident report. For example, the occurrence code "Airplane/component/system failure/malfunction" corresponds to multiple sentences in the original aviation accident report, including "loss of the alternator, "loss of the cockpit lighting," "loss of compass," etc. These keywords are never directly mentioned in the sentence, but rather, they are inferred from several sentences or the entire manuscript.

3. Lack of data for some fine-level categories. When we are trying to build a supervised learning model, we must ensure enough samples for all categories. For example, according to the event taxonomy defined by NTSB, there are 58 occurrence codes in the coarse-level labels and 1,432 subject codes in the fine-level labels. In this case, because the coarse category is always a general abstract of many possible causes, it often accumulates enough samples for us to build a supervised learning model. However, as a subset of coarse categories, many fine categories might suffer a few-shot learning problem (Wang et al. 2020). For example, there are 1,191 fine categories that have fewer than 100 accident reports, and there are 602 fine categories have fewer than 10 samples in the NTSB data.

4. Data imbalance issue. As the number of fine categories is much larger than the number of coarse categories, it resembles the natural nonuniform distribution in real-world scenarios. The fine categories appear to follow a long-tail distribution pattern, as shown in Figure 1. This phenomenon makes the model tend to be overconfident in the frequent categories and makes the model have a hard time recognizing the rare categories. The two challenges mentioned here are commonly seen when dealing with a data set with a large label size. In this paper, we propose to deal with the problems by utilizing the category taxonomy and viewing this problem as a hierarchical classification task.

In the NTSB accident report, the accident events may consist of a hierarchical structure. Here, we define a more general definition of label hierarchy using a directed acyclic graph, where each node may have more than one parent node. The parent nodes represent a high-level abstract of all its children. Figure 2 presents an example of label taxonomy in the aviation domain defined by NTSB. Figure 2 shows that flight control

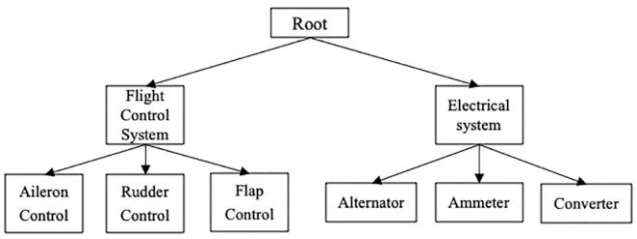
Figure 1. (Color online) Label Statistics for the 100 Most Frequent Fine-Level Labels in NTSB Data



systems and electrical systems are coarse categories. They provide general information about what went wrong during the accident. To classify the fine-level category, such as the flap control system or the rudder control system, because the sample size for coarse-level categories is much larger than that for fine-level categories, it is much easier to discriminate between the flight control system and the electrical system. On the other hand, it is challenging for the model to directly recognize aileron control or rudder control because of the limited sample size. To address this challenge, we propose to bring the hierarchical information from the labels into the model instead of treating each category level independently.

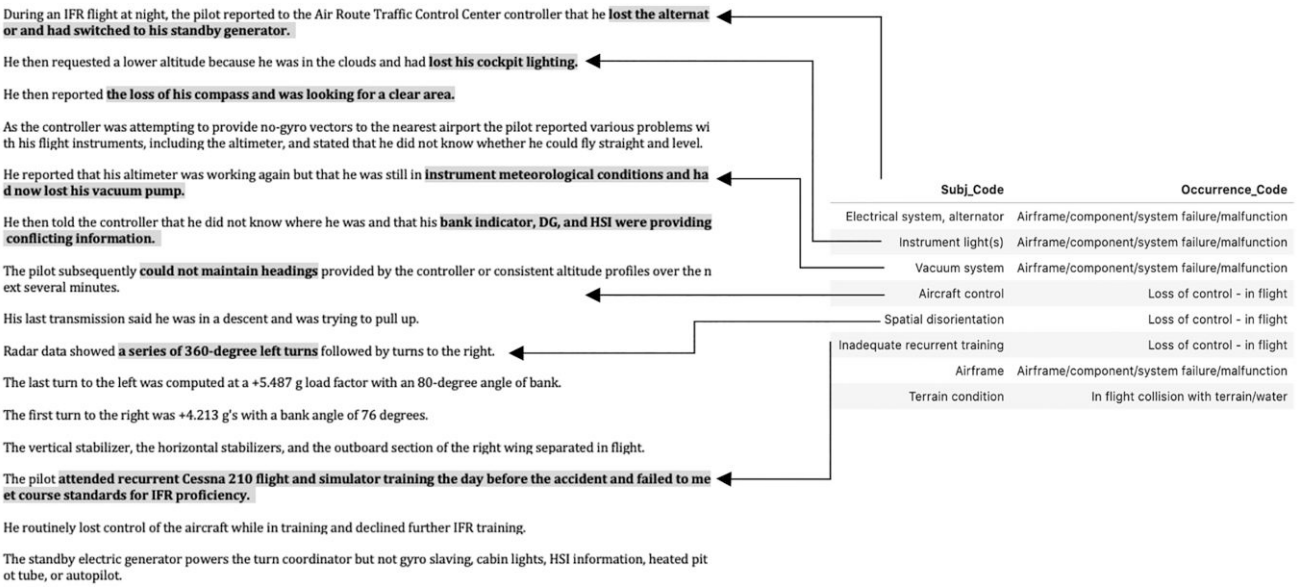
In the literature, bringing the label hierarchy to the classification problem is often known as hierarchical classification, which has shown the capacity to improve fine-level prediction accuracy in the case of unbalanced data and small sample sizes for some categories. The hierarchical classification approaches are typically grouped into two directions. (1) Local approaches train multiple classifiers according to the label taxonomy, and the prediction follows a top-down manner. (2) Global approaches introduce a single classifier that can handle different levels of categories together (Silla and Freitas 2011). The local approaches train category-wise models or level-wise models. Level-wise models follow a top-down manner where coarse categories are predicted first, and the fine level is predicted using the inferred coarse-level information as additional predictors. It is much easier to capture local information from the data. However, such an approach often suffers from the error propagation problem as fine-level prediction requires very accurate coarse-wise models. Global approaches treat the problem as a whole, which might not capture the local information very well. However, it usually utilizes the hierarchical information much better by simultaneously optimizing for both coarse categories and fine categories together. For more detailed literature on the general hierarchical classification, please refer to Section 2.

Figure 2. An Example of the Label Taxonomy Defined by NTSB



In conclusion, we propose to combine the hierarchical multilabel classification approaches with a state-of-the-art BERT model to identify multiple events with fine-level categories from aviation accident reports. Our experiments are conducted based on the reports collected by NTSB. As shown in Figure 3, the accident report is associated with multiple event labels. The occurrence code represents a coarse-level description of the accident, and the subject code represents a fine-level description. Given that the size of subject labels is too large, the traditional multilabel classification algorithm cannot achieve satisfying results because of data sparsity and data imbalance. This work aims to improve the fine-level classification by incorporating

Figure 3. Aviation Accident Report to Event Labels: Report 20001208X07734 from NTSB



Notes. The left side is the raw accident report. On the right side is the event sequence labeled by NTSB. We highlight the keywords in the narrative reports and plot their relationship with the corresponding event labels. IFR, Instrument Flight Rules are a set of regulations that govern aircraft operations when visual reference is not safe; DG, directional gyro; HSI, horizontal situation indicator.

the event taxonomy into the hierarchical classification. The proposed method takes advantage of the state-of-the-art BERT model to handle the raw text data with the complex contextual word dependency. Furthermore, related to the hierarchical multilabel classification, the proposed method takes advantage of both the local and global approaches for hierarchical classification by integrating hierarchical regularization, hierarchical attention, and hierarchical label distribution penalty to extract multiple events simultaneously from the accident report.

In summary, the major contributions of this paper are listed as follows.

1. We formulate the cause identification of aviation accident reports by combining the hierarchical classification and the state-of-the-art BERT model. By comparing with the BERT-based flat classification model, we demonstrate that the hierarchical information in the event taxonomy can benefit the fine-level classification accuracy.

2. We propose a novel hierarchical classification framework that benefits from both the local and global approaches. More specifically, our proposed hierarchical classification approach consists of three major components.

   a. Recursive regularization. Recursive regularization is added to encourage the model parameter to be similar for the sibling nodes to the parent nodes.

   b. Hierarchical attention. Hierarchical attention is added to guide the attention module of the fine-level model using the coarse-level information.

   c. Hierarchical label distribution penalty. We propose a novel hierarchical label distribution penalty term. The component can further improve the classification accuracy on rare labels by dealing with the overconfidence issue on frequent labels.

3. We have applied the proposed method to the NTSB accident report data and have demonstrated the improvement of the multilabel classification accuracy, especially for rare events.

The paper is organized as follows. We first review existing works on texting mining on narrative reports and state-of-the-art hierarchical classification methods in Section 2. We further introduce some basic notations and the proposed methods in Section 3. The experiment results will be discussed in Section 4. Moreover, we finally conclude the paper in Section 5.

2. Literature Review

We would like to review the related literature in the areas of both accident report analysis and hierarchical classification. More specifically, from the application perspective, we will first give a brief overview of existing works on automating the information extraction for narrative reports in Section 2.1. From a methodology perspective, we will review the recent progress on hierarchical classification in Section 2.2.

2.1. Information Retrieval for Narrative Reports

It is necessary to provide an efficient prevention strategy when an accident happens by analyzing past accidents.

Therefore, more researchers started developing tools to automatically extract information from narrative reports. The fundamental of the algorithms is the meaningful event taxonomy or impact factors defined by domain experts. There are various types of taxonomic structures defined early in the aviation domain. Three codifications are designed specifically for ASRS, which are the ASRS codification, the X-Form, and the Cinq-Demi (Ferryman et al. 2006). The ASRS codification is currently used to describe the accident in the ASRS database, which presents an abstract of the accident through the "chains of events" and "human performance consideration." There are nine coarse-level categories from the ASRS codification: aircraft, events, maintenance factors, etc. Under each course category, multiple fine categories are included. For example, around 180 subcategories are under the aircraft category, and around 130 subcategories are under the event category. Another widely used accident reporting system is NTSB. It designed three-level categories to describe the accident: phase, occurrence, and subject. There are around 56 phase categories, 58 occurrence categories, and 1,432 subject categories. Based on those well-defined taxonomies, many researchers have started designing efficient information extraction frameworks to automate the accident reporting system in the aviation domain. Huang et al. (2019) developed algorithms to extract the information from the ASRS data set according to the X-Form taxonomy. They proposed a template-based framework to identify the related keywords in the reports, and 20 ASRS reports were selected for validation purposes. The potential issue with this method is to develop an exhaustive set of keywords and expressions. Abedin et al. (2010) proposed an automatic keyword extraction strategy via semisupervised semantic lexicon learning and validated the efficiency of the model with all 140,599 reports from 1998 to 2007. To further improve the classification accuracy, many advanced machine learning methods are applied, such as naive Bayes (Shi et al. 2017), Bayesian logistic regression (Yelundur et al. 2016), and latent semantic analysis (Robinson 2018). It is worth noting that all the previous works are only conducted on small label sizes. As mentioned by Tanguy et al. (2016), fine-grained investigation of sources of danger is very challenging. It is very difficult for the experts to tell the difference between two closely related values at a detailed level. Existing works can only be implemented over the coarse-level identification. In this work, we propose a framework that tries to identify the fine-level categories by using the category taxonomy.

2.2. Hierarchical Classification

In many real-world applications, we can see the hierarchical category structure. For example, in the e-commerce system, products are classified into categories and subcategories (Karamanolakis et al. 2020). In the healthcare system, the taxonomy of patients' symptoms helps the doctors diagnose much easier (Ruggero et al. 2019). Motivated by those hierarchical representations, researchers start incorporating knowledge from the hierarchical taxonomy into traditional classification methods. Most existing hierarchical classification methods fall into two categories, which are local approaches and global approaches (Silla and Freitas 2011). Local approaches focus on building independent classifiers for each category or level from the taxonomy. They aim at extracting the most comprehensive local information from the samples. They usually follow a top-down prediction strategy according to the taxonomy (Koller and Sahami 1997, Kiritchenko et al. 2004, Costa et al. 2007). One major problem of the local approaches is the error propagation and the expensive computational cost. Because multiple classifiers are developed independently, the model parameters increase rapidly, and the bias from different classifiers will accumulate. Different from local approaches, global approaches treat the problem as a whole. They build a single model and predict categories from different levels together (Cai and Hofmann 2004, Cerri et al. 2012, Gopal and Yang 2013). However, global approaches might not capture the local information very well. They require fewer training resources and are more sensitive about the relationship across different levels among all categories. Considering all the pros and cons of the two methods, recent works have started building hybrid methods to take advantage of both approaches. Huang et al. (2019) proposed a hybrid approach through a recurrent neural network. The output of each layer corresponds to a level from the category taxonomy. By adding a global output layer as the final layer of the network, the proposed method optimizes local loss and global loss simultaneously. To further enhance the relationship between the input and labels, a hierarchical attention-based approach is proposed, which also falls into the category of a hybrid approach. By leveraging the hierarchical structure, the methods propagate the attention weight top down. Inspired by the aforementioned methods, we proposed a novel hierarchical attention-based framework.

Unlike previous work, because our task suffers a long-tail distribution problem, we need to find an efficient solution to deal with the data sparsity and imbalance challenge. Recent studies have shown that we can leverage the hierarchical structure to deal with those problems. A new sampling strategy is proposed to deal with the imbalanced hierarchical data set (Pereira et al. 2021). Wu et al. (2020) proposed a rejection strategy based on the label taxonomy to identify uncertain examples. Other works focusing on the few-shot learning settings have also proved the value of label hierarchy. Considering the relationship among siblings belonging to the same level of the category, Xu and Geng (2019) introduced label distribution learning to

solve the small training set issue. Li et al. (2019) proposed a novel hierarchical feature extractor that achieves state-of-the-art results for a few-shot learning task. Another two-stage few-shot learning approach is proposed in Liu et al. (2022). It argued that as the size of fine-level labels grows dramatically, the limited sample size for each category can hardly provide enough information for existing models to distinguish those. However, the sample size from the coarse-level categories is usually sufficient for a reliable supervised model. The paper develops a memory-augmented hierarchical classification network based on those observations. Multilayer perceptron and *K*-nearest neighbor are developed for coarse prediction and fine prediction, respectively, because of their sample size. Following this line of research, we also design a two-stage approach and feed the reliable coarse-level predictions into our fine-level model for fine-tuning purposes. We guide the model training process with a recursive regularization term and a label distribution penalty to improve the prediction accuracy on few-shot categories.

3. Methodology

In this section, we will introduce the proposed model. More specifically, we will first review the related methods in Section 3.1. We will then introduce the proposed hierarchical multilabel classification BERT model in Section 3.2.

3.1. Review of Related Methodology

In this subsection, we will review the three related methods in handling the hierarchical labels, including the hierarchical attention model in Section 3.1.1, recursive regularization in Section 3.1.2, and label distribution learning in Section 3.1.3.

3.1.1. Review of Hierarchical Attention. The attention mechanism is widely used in recent deep learning approaches. It helps us build the connection between labels and input. By adding the attention module, we can extract the most relevant component of the input to the label (Vaswani et al. 2017). To leverage the label hierarchy information through the attention module, hierarchical attention is proposed by Huang et al. (2019). With the assumption that coarse-level information can help us narrow down the relevant component for fine-level prediction, we can propagate coarse-level attention to fine-level attention by building better semantic representation. Let $S^h \in \mathbb{R}^{|C^h| \times d_a}$ denote the embedding vector for the hth level of the category. Let $|C^h|$ represents the label size, and d_a is a hyperparameter for the embedding dimension. The attention weight for the hth category level can be calculated as

$$W_{att}^h = \text{Softmax}(S^h \cdot O_h), \quad O_h = \tanh(W_s^h \cdot V_h^T), \tag{1}$$

where $V_h \in \mathbb{R}^{N \times 2u}$ is the semantic representation of the input and N is the length of the input. We further send V_h through a fully connected layer with weight matrix $W_s^h \in \mathbb{R}^{d_a \times 2u}$ to get $O_h \in \mathbb{R}^{d_a \times N}$. After getting the attention weight, we will further calculate the weighted text-category attention matrix $K^h \in \mathbb{R}^{|C^h| \times N}$ with the local prediction on the hth level of the category P_L^h:

$$K^h = \text{Broadcast}(P_L^h) \otimes W_{att}^h. \tag{2}$$

Here, the Broadcast operation is to make the shape of P_L^h compatible with attention matrix W_{att}^h so that we can conduct the element-wise multiplication $\otimes$. K_h is further averaged along the category dimension, and then, we can get the vector $\tilde{K}^h \in \mathbb{R}^N$ that shows the most relevant component from our input to the hth category level. Finally, the updated semantic representation can be calculated with the weight attention vector $\tilde{K}^h$:

$$V_{h+1} = \omega^h \otimes \mathbf{V}, \quad \omega^h = \text{Broadcast}(\tilde{K}^h). \tag{3}$$

It is worth noting that the attention weight is propagating to the lower level through constructing a new semantic representation. In our framework, we simplify the calculation and let coarse-level attention directly influence the attention weight on the fine level.

3.1.2. Review of Recursive Regularization. Recursive regularization was first proposed by Gopal and Yang (2013) as a global strategy to deal with hierarchical classification. It encourages the model parameters among sibling nodes in the label hierarchy to be similar. Later work extends this method to a deep learning setting (Peng et al. 2018). Given the observation that coarse-level categories usually have more training samples than fine-level categories, it is easier to get the optimal parameters for the coarse category. Thus, regularizing the children's nodes to have similar parameters to their parents will help improve the fine-level classification, even though we have fewer training samples on fine categories. To be formal, let w_i be the parameters in the final fully connected layer for all categories. Let l_i^j represents the children of l_i. Then, the recursive regularization term can be added to the loss function as shown in Equation (4):

$$\lambda(\mathbf{W}) = \sum_{l_i \in L} \sum_{l_i^{(j)}} \frac{1}{2} \| w_{l_i} - w_{l_i^j} \|^2. \tag{4}$$

Here, the final loss function will be $H + C\lambda(\mathbf{W})$, where H is the crossentropy loss. Inspired by this formulation, our model also implements a similar regularization idea. However, because we mainly express the hierarchical information through the attention module, we will add this penalty by embedding vectors from different levels of categories.

3.1.3. Review of Label Distribution Learning. Another algorithm that can deal with the small training set issue through the label hierarchy is called label distribution

learning (Geng 2016, Xu and Geng 2019). It is argued in the paper that the fine classifiers tend to overfit because of the small training size. We can leverage the relationship among siblings in the label hierarchy to get additional supervision information. As mentioned in the paper, label distribution learning is a more general learning framework for single-label learning or multilabel learning. Under the label distribution learning setting, each instance x_i is associated with a label distribution $D_i = \{d_{x_i}^{y_1}, d_{x_i}^{y_w}, \ldots, d_{x_i}^{y_n}\}$, where y_j is the jth label. Let $d_{x_i}^{y_j} \in [0,1]$ represents the degree of y_j to instance x_i, where $\sum_j d_{x_i}^{y_j} = 1$. $d_{x_i}^{y_j}$ can also be represented as the form of conditional probability, where $d_{x_i}^{y_j} = P(y_j|x_i)$.

To be more formal, let $\mathcal{X} \in \mathbb{R}^q$ denote the input space, and let $\mathcal{Y} = \{y_1, y_2, \ldots, y_c\}$ denote the complete set of labels. Given a training set $S = \{(x_1, D_1), (x_2, D_2), \ldots, (x_n, D_n)\}$, then the goal of label distribution learning becomes to learn a conditional probability mass function $p(y|x)$ from S, where $x \in \mathcal{X}$ and $y \in \mathcal{Y}$. Let the parameter θ control the conditional probability $p(y|x,\theta)$. Then, we need to find the optimal parameter set to generate a similar distribution to D_i given the instance x. Multiple criteria can be used to evaluate the difference. Kullback–Leibler (KL) divergence is a frequently used measurement to describe the similarity between two distributions. Moreover, the optimization objective can be formulated as follows:

$$\theta^* = \arg\min_{\theta} \sum_i \sum_j \left(d_{x_i}^{y_j} \ln \frac{d_{x_i}^{y_j}}{p(y_j|x_i,\theta)} \right). \tag{5}$$

Under the hierarchical classification setting, the true label distribution D is usually not achievable. Moreover, there are multiple ways to construct meaningful label distribution according to the label hierarchy. One possible way to get the label distribution, which was mentioned by Xu and Geng (2019), is through leveraging the knowledge of the number of common nodes with the true path. We propose a novel way to achieve label distribution in our work. We propose to use the Bayes rule to get the fine-level distribution through our prediction of the coarse-level distribution.

3.2. Proposed Model for Hierarchical Multilabel Classification

In this section, we will introduce the proposed model for the hierarchical multilabel classification model and combine it with the BERT model to analyze the accident report data. Section 3.2.1 presents the overall problem definition. Section 3.2.2 presents the overall model architecture. Section 3.2.3 presents the BERT-based feature extractor. Section 3.2.4 presents the architecture of the coarse-level model. The three major components of the fine-level model are introduced in Sections 3.2.5, 3.2.6, and 3.2.7, respectively. Finally, the loss function that integrates all the components together is introduced in Section 3.2.8.

3.2.1. Problem Definition.

This paper aims to design an information extraction framework for aviation accident reports. As shown in Figure 3, there is a two-level description in the reports, and these two levels are called occurrence and subject. Occurrence is a coarse-level event description containing 56 categories, whereas subject is a fine-level description with 1,432 categories. Motivated by the label hierarchy, we want to build a coarse-to-fine hierarchical classification model to address the major challenges of the lack of enough samples in many fine-level categories.

Under the supervised learning setting, our problem can be formulated as follows. Given a set of training samples, $\mathcal{D} = \{(\boldsymbol{x}_n, \boldsymbol{y}_n, \boldsymbol{z}_n)\}_{n=1}^N \subseteq \mathcal{X} \times \mathcal{Y} \times \mathcal{Z}$. Because our input is accident reports, each $\boldsymbol{x}_n$ represents the input documents, and $\mathcal{X}$ denotes the space of the natural language text, where $\mathcal{X} = \{\omega_1, \omega_2, \ldots, \omega_M\}$ can be viewed as a sequence of words. $\boldsymbol{y}_n = [y_1, y_2, \ldots, y_{L_1}] \in \mathcal{Y} = \{0,1\}^{L_1}$ denotes L_1 number of coarse-level categories. $\boldsymbol{z}_n = [z_1, z_2, \ldots, z_{L_2}] \in \mathcal{Z} = \{0,1\}^{L_2}$ denotes L_2 number of fine-level categories. Our goal is to get an accurate fine-level prediction based on our observations:

$$\min_{\theta} \mathbb{E}_{\{\boldsymbol{x},\boldsymbol{y},\boldsymbol{z}\}\sim\mathcal{D}}(-\log \mathbf{Pr}(\boldsymbol{z}|\boldsymbol{x};\theta)), \tag{6}$$

where θ denotes the model parameters. Based on the observation that coarse-level categories usually have enough samples for the training purpose, in order to further improve the fine-level prediction, the objective function can be broken into two components as follows:

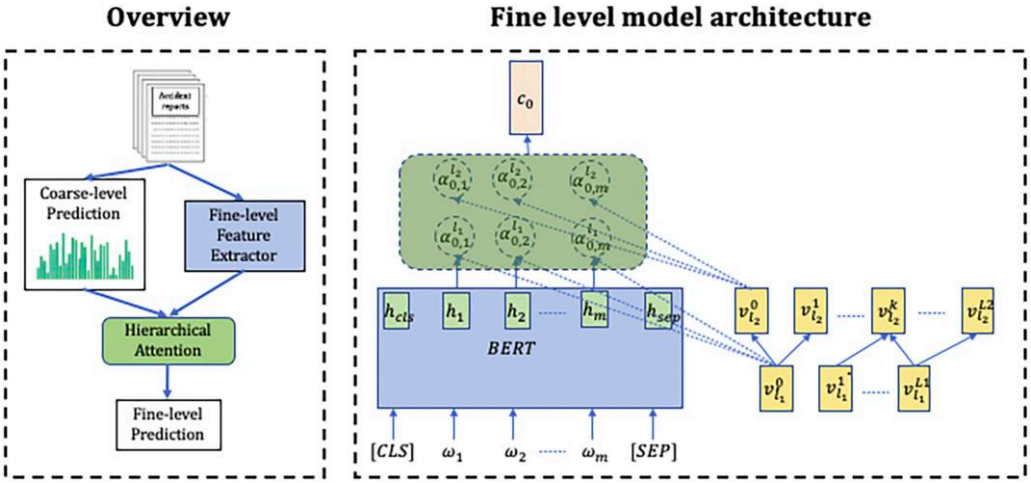
$$\mathbf{Pr}(\boldsymbol{z}|\boldsymbol{x};\theta) = \sum_{\boldsymbol{y}\in\mathcal{Y}} \mathbf{Pr}(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{y};\theta_f)\mathbf{Pr}(\boldsymbol{y}|\boldsymbol{x};\theta_c), \tag{7}$$

where θ_f and θ_c represent the fine-level model parameters and coarse-level model parameters, respectively. Thus, the problem can be transformed into a two-stage decision framework. First, we need to design a coarse-level classifier. Second, we need to design a fine-level classifier that can utilize the information predicted from the coarse model. It is worth noting that the proposed procedure is similar to the top-down prediction framework of the local approaches in hierarchical classification. To mitigate the error propagation issue, the information from the coarse models is not added to the fine models directly as in traditional local hierarchical classification approaches (Koller and Sahami 1997). We build a hierarchical attention module to get better fine-level feature representation with the guidance of coarse-level information. We further optimized embedding vectors for coarse categories and fine categories together following a global approach framework.

3.2.2. Overview of the Model Architecture.

Figure 4 presents an overview of the proposed framework. There are three major components of the framework: a coarse-level classifier, a fine-level feature extractor, and

Figure 4. (Color online) Overview of the Two-Stage Hierarchical Attention



Note. CLS, classification; SEP, seperation.

a hierarchical attention module. Our methods provide coarse-level prediction through fine-tuning on a pretrained BERT model (Devlin et al. 2019b). Bidirectional encoder representation from transformers is one of the most powerful pretrained deep language models, and it has achieved state-of-the-art results in many text mining tasks. A BERT-based classifier can provide satisfying results with a sufficient amount of coarse data. Therefore, it can further guide our fine classifiers. We first construct the deep representation for the accident reports with BERT for fine-level prediction. Then, the text features will feed into the hierarchical attention module and the coarse category distribution, and they will make the fine-level prediction. Finally, we develop two penalty terms in our loss function according to the label distribution and label hierarchy to deal with the long-tail distribution over fine categories. In the following sections, we will mainly introduce the major components of the fine-level classifier.

3.2.3. BERT-Based Feature Extractor. In order to get meaningful text representation, we apply the BERT model to encode text into embedding space to extract the text embedding for future use. As shown in Figure 4, the input document is a sequence of M words $\boldsymbol{x} = [\omega_1, \omega_2, \ldots, \omega_M]$. We apply the BERT model to get the semantic representation of the text data. The discrete word ω_i is first mapped into key $\boldsymbol{k}_i$, value $\boldsymbol{v}_i$, and query $\boldsymbol{q}_i$ according to the self-attention mechanism. The contextual representation of the word $\boldsymbol{z}_i$ is computed as follows:

$$z_i = \text{Softmax}\left(\frac{\boldsymbol{q}_i \boldsymbol{k}}{\sqrt{d_k}}\right)\boldsymbol{v}. \tag{8}$$

The BERT model is built upon multiple self-attention layers and feed-forward connections. In the last layer, we can get the semantic representation of each word based on its correlation to all other words in the document. Here, we will use $\boldsymbol{H} = [\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_M] \in \mathbb{R}^{M \times u}$ to represent the final embedding vector calculated by BERT, where u is the embedding dimension. To further regularize the overfitting issue, a dropout layer is added to the final layer of the BERT model.

3.2.4. Coarse-Level Model. We will then train a coarse-level multilabel classification model based on the extracted features from the BERT-based feature extractor to be used as the local approach to guide the fine-level multilabel classification. The coarse-level prediction model consists of the following components: a BERT-based feature extractor, an attention module for identifying the coarse-level relevance score, and the final Binary Cross Entropy loss function for multilabel classification. Figure 5 illustrates the architecture of the coarse-level model. Let $V_{l_1} = [\boldsymbol{v}_{l_1}^1, \boldsymbol{v}_{l_1}^2, \ldots, \boldsymbol{v}_{l_1}^{L_1}] \in \mathbb{R}^{L_1 \times u}$ denote the coarse label embedding. We first calculate the relevance score between documents and labels as follows:

$$e_{ij} = \boldsymbol{v}_{l_1}^i \cdot \boldsymbol{h}^j, \tag{9}$$

where e_{ij} is the relevance score between the ith label and the jth word in the document. The attention weight can be further calculated through a Softmax function:

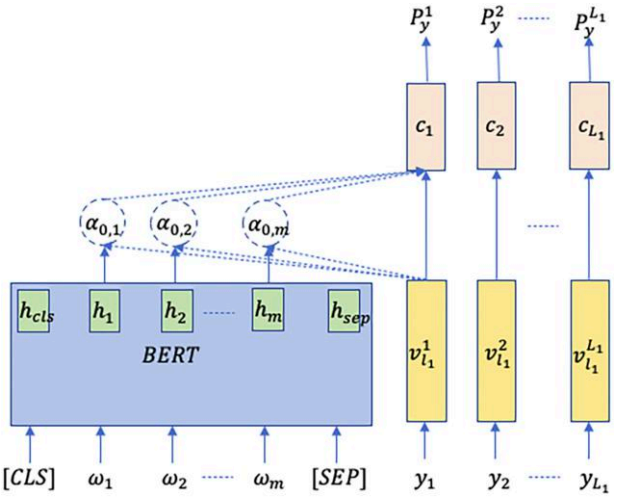
$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{i=1}^{m} \exp(e_{ij})}. \tag{10}$$

With the attention weight, the contextual vector $\boldsymbol{c}_i \in \mathbb{R}^u$ for the ith label can be calculated with the hidden vector as follows:

$$\boldsymbol{c}_i = \sum_{j=1}^{m} \alpha_{ij} \boldsymbol{h}_j. \tag{11}$$

Let $P_y = [P_y^1, P_y^2, \ldots, P_y^{L_1}] \in \mathbb{R}^{L_1}$ denote the predicted probability for coarse-level labels, which can be

Figure 5. (Color online) Coarse-Level Model Architecture



Note. CLS, classification; SEP, seperation.

calculated through a fully connected layer and the sigmoid function as follows:

$$P_y^i = \frac{1}{1+\exp(\boldsymbol{w}_f \cdot \boldsymbol{c}_i + \boldsymbol{b}_f)}, \tag{12}$$

where $\boldsymbol{w}_f \in \mathbb{R}^u$ and $\boldsymbol{b}_f \in \mathbb{R}^u$ are the weight and the bias vector for the final fully connected layer, respectively. Finally, we optimize the binary crossentropy loss between the true coarse label and the predicted probability as follows:

$$\mathcal{L}_{BCE}(\boldsymbol{y}, P_{l_1}) = -\sum_{i=1}^{L_1}[y^i \log(P_{l_1}^i) + (1-y^i)\log(1-P_{l_1}^i)]. \tag{13}$$

3.2.5. Hierarchical Attention. After getting the embedding representation of the accident reports $\boldsymbol{H}$, we will make the fine-level prediction through the hierarchical attention module. We first calculate the coarse-level relevance score similar to coarse-level attention as follows:

$$K_{l_1} = V_{l_1} \cdot \boldsymbol{H}^T, \tag{14}$$

where $K_{l_1} \in \mathbb{R}^{L_1 \times M}$ is the relevance score matrix between the context and coarse labels under a fine-level model. We further collect coarse category distribution $P_y \in \mathbb{R}^{L_1}$ from coarse classifier. The coarse category distribution helps us understand the importance of the attention weight from different categories. By applying an element-wise multiplication, we can get the weighted coarse-level relevance score matrix $O_{l_1} \in \mathbb{R}^{L_1 \times M}$ as follows:

$$O_{l_1} = \text{Broadcast}(P_y) \otimes K_{l_1}, \tag{15}$$

where Broadcast() is an operation to expand P_y so that it has the same dimension as O_{l_1}. Our weighted attention matrix $O_{l_1} = (O_{l_1}^1, O_{l_1}^2, \ldots, O_{l_1}^{L_1}) \in \mathbb{R}^{L_1 \times M}$ comprises L_1 vectors, where each of them presents the category-wise word relevance within the input documents. In order to get a global attention weight score for each word, we further take a summation over the category dimension and apply the Softmax function to ensure that all of the computed weights sum to one as follows:

$$W_{l_1} = \text{Softmax}\left(\sum_{i=1}^{L_1} O_{l_1}^i\right), \tag{16}$$

where $W_{l_1} \in \mathbb{R}^M$ represents the global word attention weight for each input instance.

After calculating the coarse-level attention, we can further develop the fine-level attention weight. Let us first define fine category embedding $V_{l_2} \in \mathbb{R}^{L_2 \times u}$, and the attention weight matrix can be calculated as follows:

$$W_{l_2} = \text{Softmax}(V_{l_2} \cdot H^T), \tag{17}$$

where $W_{l_2} \in \mathbb{R}^{L_2 \times M}$ shows the attention weight for each fine-level category. To further incorporate the coarse-level information, we build a linear combination between coarse-level attention and fine-level attention as follows:

$$W'_{l_2} = W_{l_2} + \lambda \cdot \text{Broadcast}(W_{l_1}), \tag{18}$$

where λ controls how much we want to leverage the coarse-level guidance. When $\lambda = 0$, the formulation becomes a flat classification framework. As we increase λ, the attention weight will rely more on the coarse-level information.

After getting the updated attention weights, we can further calculate the category-wise contextual representation as follows:

$$C = W'_{l_2} \cdot H, \tag{19}$$

where $C = (C^1, C^2, \ldots, C^{L_2}) \in \mathbb{R}^{L_2 \times u}$ represents L_2 contextual representation for each category according its relationship to the input document.

Then, the fine-level prediction can be achieved through fully connected layers and an activation function as follows:

$$P_{l_2} = \sigma(\phi([C \oplus V_{l_2}] \cdot W + \boldsymbol{b})), \tag{20}$$

where $W \in \mathbb{R}^{2u \times 1}$ and $\boldsymbol{b} \in \mathbb{R}^{L_2}$ represent the parameters that belong to the final fully connected layer. $\oplus$ denotes the concatenation operation to concatenate C and V_{l_2} together. ϕ is the nonlinear activation function, where we apply rectified linear unit in our context.

Finally, the proposed model will optimize the binary crossentropy loss between the true label distribution and the predicted distribution as follows:

$$\mathcal{L}_{BCE}(\boldsymbol{z}, P_{l_2}) = -\sum_{j=1}^{L_2}[z^j \log(P_{l_2}^j) + (1-z^j)\log(1-P_{l_2}^j)]. \tag{21}$$

3.2.6. Recursive Regularization. The fine-level prediction can be highly improved through the top-down prediction method given coarse-level information. However, as mentioned in Section 3.1.2, we still do not

have enough training samples for many fine categories. That means that we cannot get optimal parameters for the corresponding embedding representation V_{l_2}. On the other hand, training samples for the coarse level are usually sufficient. Based on the assumption that the parent category should share similarities with the children category, we can optimize the fine-level embedding by letting it close to its connected coarse-level embedding. We incorporate it into our loss function as a similarity constraint. Let $v^i_{l_1} \in \mathbb{R}^u$, $\forall i = 1, \ldots, L_1$ represent the embedding vectors for coarse categories. For the ith coarse category, the connected fine-level category embedding is $v^j_{l_2}$, $\forall j \in \pi(i)$. Inspired by the recursive regularization framework in Equation (4), the constraint can be expressed as follows:

$$\mathcal{L}_{similarity} = \sum_{i=1}^{L_1} \sum_{j \in \pi(i)} \frac{1}{2} \| v^i_{l_1} - v^j_{l_2} \|. \tag{22}$$

3.2.7. Label Distribution Penalty. The recursive regularization term helps us build more connections between parent labels and children labels across the label hierarchy. To further improve the rare label identification, we can also use the sibling labels of the rare label. As introduced in Section 3.1.3, a proper label distribution can add additional supervision information to the training process. For those rare labels, we can enrich the label space by leveraging the frequent labels that share similar information to the rare labels. In this work, we propose to use the coarse label distribution to help build fine label distribution according to the strength of the connection between parents and children. Let $P^i_{l_1}$, $\forall i = 1, \ldots, L_1$ be the coarse-level probability calculated by the coarse classifiers. For the jth fine category, we use w^i_j, $\forall i = 1, \ldots, L_1$, $\forall j = 1, \ldots, L_2$ to denote its connection with the ith coarse category. The strength of connection w^i_j is calculated as follows:

$$w^i_j = \frac{\#(y_i, z_j)}{\#(y_i)}, \tag{23}$$

where $\#(y_i, z_j)$ represents the number of co-occurrences of the coarse category y_i and the fine category z_j and $\#(y_i)$ represents the number of occurrences of y_i. The probability of the jth fine category can be calculated as

$$d^j = \frac{\sum_{i=1}^{L_1} P^i_{l_1} w^i_j}{\sum_{i=1}^{L_1} \sum_{j=1}^{L_2} P^i_{l_1} w^i_j}. \tag{24}$$

We further use KL divergence to measure the distance between the predicted fine distribution and the expected distribution calculated through coarse distribution as follows:

$$\mathcal{L}_{distribution} = \sum_j \left(d^j \ln \frac{d^j}{P^j_{l_2}} \right). \tag{25}$$

3.2.8. Loss Function. After adding the distribution penalty, our final objective becomes

$$\mathcal{L} = \mathcal{L}_{BCE} + \lambda_1 \mathcal{L}_{similarity} + \lambda_2 \mathcal{L}_{distribution}, \tag{26}$$

where $\mathcal{L}_{BCE}$ is the multilabel classification loss defined in Equation (21); $\mathcal{L}_{similarity}$ is the recursive regularization defined in Equation (22); $\mathcal{L}_{distribution}$ is the hierarchical label distribution loss defined in Equation (25); and λ_1 and λ_2 control the impact of recursive regularization and label distribution penalty, respectively. The best combination of those parameters is found through experiments. When $\lambda_1 = 0$, the framework becomes a local hierarchical classification approach, and the fine-level embedding vectors are optimized, relying on local information. As λ_1 increases, the framework becomes a global approach, where the fine-level embedding is regularized by the coarse-level embedding. When $\lambda_2 = 0$, the label space is under a multilabel learning setting. As λ_2 increases, the framework becomes a label distribution learning setting. It helps enrich the label space and provides additional information for the rare labels.

4. Experiment Setup

In this section, we will introduce more details about the experiment design. An overview of the NTSB data set will be given in Section 4.1. In Section 4.2, we will briefly talk about the implementing platform. We will further introduce the models for comparison and the comparison methods in Sections 4.3 and 4.4, respectively. After introducing the experiment setup, the major results are discussed in Sections 4.5 and 4.6.

4.1. Data Description

In this project, we mainly work with the aviation accident reports collected by NTSB. The NTSB database and its coding system are considered to be the state of the art for aviation safety analysis given that the NTSB database is the largest and most comprehensive repository of aviation accident data in the United States. This distinction stems not only from its extensive historical coverage, dating back to 1962, but also from its meticulous structuring and coding of a wide array of data points. The database's depth and breadth provide great insights into aviation safety, encompassing various aspects, such as aircraft types, operational contexts, environmental conditions, and the sequence of events leading up to accidents. Moreover, the coding system enables detailed and nuanced analysis. It facilitates the identification of trends, patterns, and correlations that might otherwise remain obscured in less comprehensive data sets. This level of detail is crucial for developing targeted safety recommendations, shaping policy, and guiding industry practices. The database's influence extends beyond the United States, often serving as

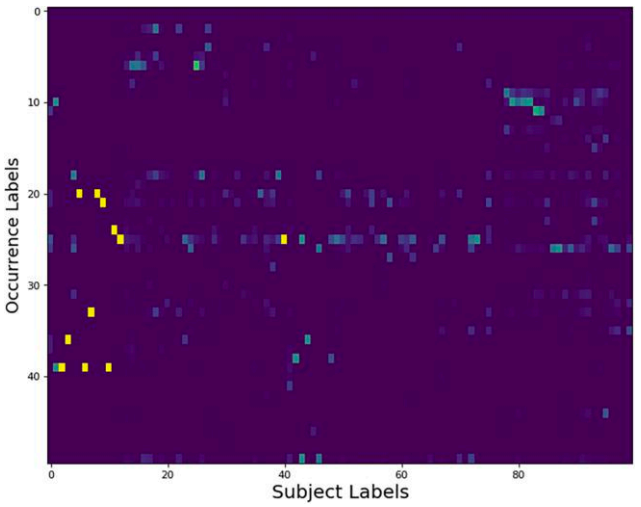
a benchmark for aviation safety standards and practices globally as well (Srinivasan et al. 2019, Fuller and Hook 2020, Zhang and Mahadevan 2020, Zhang et al. 2021, Zhao et al. 2022).

From 1982 to 2008, NTSB collected around 62,569 accident reports following the same hierarchical labeling strategy. There are 56 phase codes, 58 occurrence codes, and 1,432 subject codes (National Transportation Safety Board 2008). The data are publicly accessible and stored in Microsoft Access database format. Figure 1 presents the frequency of different labels in the data set, which suffers a long-tail distribution. As shown in Figure 1, the most frequent subject labels are 19200, 20000, and 20200, with 30,886, 24,565, and 19,231 samples, respectively. Those labels represent the most frequent critical events in the aviation accidents, which are *Terrain Condition*, *Weather Condition*, *Object Condition*, respectively. Two hundred and fifty subject labels have more than 100 samples, 397 labels have more than 50 samples, and 602 labels have fewer than 10 samples. We select around 90% of the accident reports for training, and we use the rest of those for testing during the modeling process. We further calculate the strength of the connection for the occurrence and subject labels with the training data as mentioned in Section 3.2.7. Figure 6 gives a brief overview of the connection matrix with the most frequent 100 subject labels. The x axis in Figure 6 shows the subject labels, and the y axis in Figure 6 shows the occurrence labels. The color of each entry in Figure 6 represents the frequency of labels from two levels occurring together. See also Figure 7.

4.2. Implementation Details

We apply the BERT model to extract the input documents' semantic representation. The hidden size of the embedding vector is set to 768. To mitigate the overfitting issue, the dropout rate is set to 0.2. We further design the embedding dimension for the hierarchical attention the same as in the BERT model, considering the consistency. In order to optimize the parameter, we first limit the range of potential parameters based on previous research from Peng et al. (2018) and Xu and Geng (2019). We test different λ's to control the impact from coarse-level attention by splitting the data set. Additionally, we get the best result when $\lambda = 2$. For different combinations of the three-loss terms $\mathcal{L}_{BCE}$, $\mathcal{L}_{similarity}$, and $\mathcal{L}_{distribution}$, we finally set the corresponding weights to one, $\lambda_1 = 0.01$, and $\lambda_2 = 0.001$, respectively. The experiments were all conducted on Pytorch 1.4.0 on a workstation with an Intel Core i7-5930K Central Processing Unit (CPU) @3.50-GHz CPU and accelerated with a single NVIDIA GTX 1080 Ti Graphics processing unit.

Figure 6. (Color online) Connection Matrix Between Occurrence Labels and Subject Labels



4.3. Evaluation Metrics

In the hierarchical classification task, we apply the most common measurements used: precision, recall, and F_1. First, we will present a global score through microaveraging to compare the performance of different models. As shown in the following equations, because the micro-measurements did not calculate label-wise accuracy, it illustrates the overall accuracy for all samples:

$$P_{micro} = \frac{\sum_{j=1}^{L} tp_j}{\sum_{j=1}^{L} tp_j + fp_j},$$

$$R_{micro} = \frac{\sum_{j=1}^{L} tp_j}{\sum_{j=1}^{L} tp_j + fn_j},$$

$$F_{1-micro} = \frac{\sum_{j=1}^{L} 2tp_j}{\sum_{j=1}^{L} 2tp_j + fp_j + fn_j}.$$

However, micro-F_1 tends to favor categories that have large sample sizes when we have an imbalanced data set. In our problem, the subject labels have a long-tail distribution, as shown in Figure 1. Supervised learning is more capable of learning a good classifier for the frequent labels, which will lead to a pretty high micro-F_1 score, even though all rare categories are not identified. Thus, we further evaluate the rare category identification using macroaveraging measurements. The macroaveraging calculates the label-wise accuracy and then takes the average of them as follows:

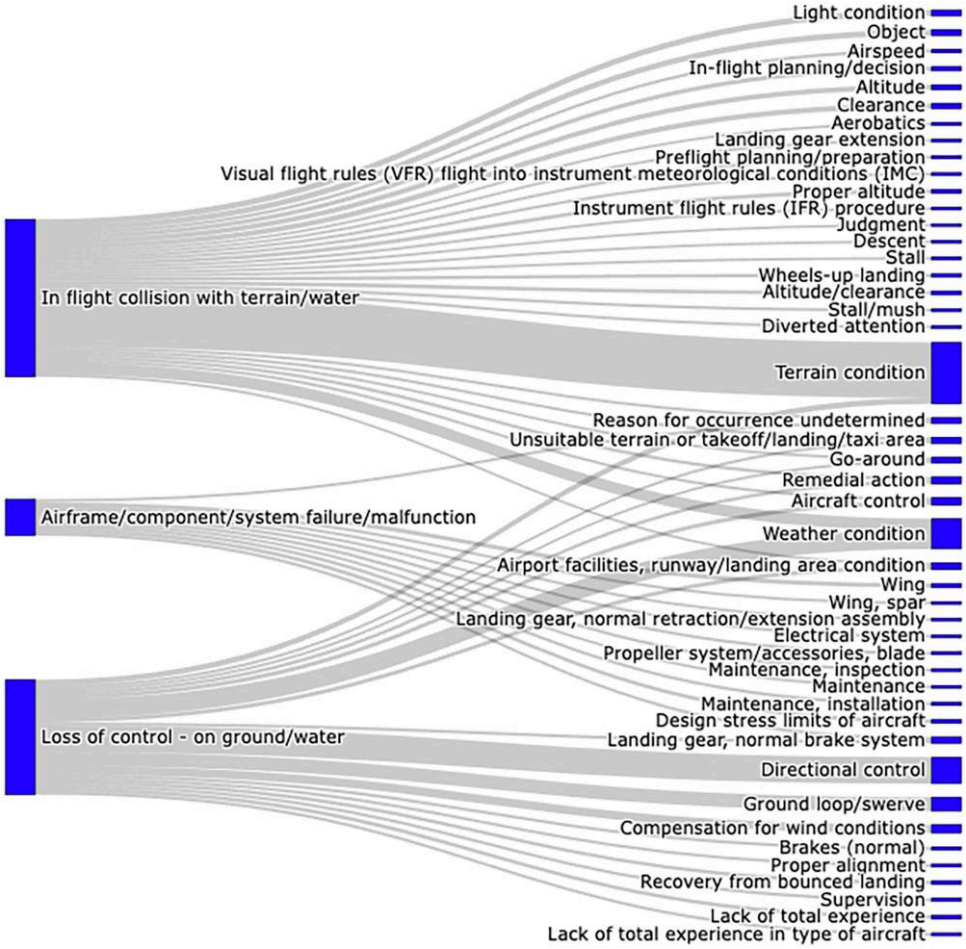
$$P_{macro} = \frac{1}{L}\sum_{j=1}^{L} \frac{tp_j}{tp_j + fp_j},$$

$$R_{macro} = \frac{1}{L}\sum_{j=1}^{L} \frac{tp_j}{tp_j + fn_j},$$

$$F_{1-macro} = \frac{1}{L}\sum_{j=1}^{L} \frac{2tp_j}{2tp_j + fp_j + fn_j}.$$

To present a fair comparison over frequent categories and rare categories, we further divide all labels into

Figure 7. (Color online) An Example of Label Taxonomy Defined by NTSB (Airframe Malfunction, Loss of Control, and in-Flight Collision)



different groups according to their sample size, and we calculate the group-wise macroaveraging.

4.4. Models for Comparison

The proposed model is compared with several state-of-the-art models as baselines, which include the following.

- Binary Relevance Support Vector Machine (BR-SVM) (Abedin et al. 2010). Under the binary relevance schema, the support vector machine can be treated as a flat classification method for the multilabel task.
- Sequence Generation Model (SGM) (Yang et al. 2018). The sequence generation model is another flat classification approach that treats the multiclassification problem as a sequence generation task.
- BERT (Devlin et al. 2019a). The BERT model provides us with a solid baseline to evaluate the effectiveness of the proposed modules.

The variants of the proposed models are listed.

- Hierarchical attention-based bidirectional encoder representations from transformers (HABERT). The HABERT model is the fundamental structure of the proposed model. It leverages the label hierarchy through the hierarchical attention module.
- HABERT-R. HABERT-R is a variant of HABERT considering the similarity between parent and children nodes through recursive regularization.
- HABERT-L. HABERT-L is a variant of HABERT considering the fine category distribution through the coarse category distribution.
- HABERT-RL. HABERT-RL is a variant of HABERT using both recursive regularization and the label distribution learning penalty.

4.5. Performance Evaluation

This section provides a global evaluation of the proposed methods through microaveraging. It tells us how different models perform over the whole data set, regardless of the frequency of the categories. Table 1 presents the results of the experiments. The results show that all the BERT-based models outperform the Support Vector Machine (SVM) and LSTM models in terms of the F_1 score. The original BERT model highly improved the recall rate because the pretrained model

Table 1. Global Comparison for Different Models with Microprecision, Microrecall, and Micro-F_1

	Metrics		
Models	Microprecision	Microrecall	Micro-F_1
BR-SVM	0.5356	0.3386	0.4608
SGM	0.5489	0.4179	0.4745
BERT	0.5408	0.4339	0.5069
HABERT	**0.6092**	0.5083	**0.5542**
HABERT-R	**0.6093**	0.5082	**0.5541**
HABERT-L	0.5743	**0.5188**	0.5452
HABERT-RL	0.5843	0.5084	0.5437

Notes. We conduct *t*-tests of the other benchmarks with the best performers in each category. We bold the model with the best performance (top performer) and any other models that are not significantly different from the top performer.

reduces the demand for training samples. We can also validate the efficiency of the hierarchical attention module where HABERT has 9% improvements on F_1 over BERT. It made a big improvement over both the precision rate and the recall rate. We further show the impact by adding the recursive regularization term. It has been shown that HABERT-R achieves the best precision score and the best F_1 score over all of the models. It proves the efficiency of adding additional supervision according to the label hierarchy. The F_1 score drops a little bit by adding the label distribution penalty. However, the recall rate for both HABERT-L and HABERT-RL increases over other models. The label distribution penalty can help the model identify more rare categories, sacrificing some precision accuracy. Overall, the proposed hierarchical attention and recursive regularization can help us get better fine-level classification. Moreover, label distribution learning can further improve rare category classification.

4.6. Rare Category Identification

We further provide more experiment results to discuss the influence of sample size on the model performance. To ensure that all categories have the same contribution, we will apply macroaveraging metrics in this section. We divide all data into six groups according to their sample size. After the processing, 46 categories have sample sizes from 1,000 to 60,000, and 43 categories have sample sizes from 500 to 1,000, which can provide sufficient samples for training. Sixty-nine categories have sample sizes within 200–500, 92 categories have sample sizes within 100–200, and 146 categories have sample sizes within 50–100. Those categories provide fewer training samples that belong to the few-shot learning settings. Other categories with fewer than 50 samples are usually very challenging for deep learning methods. Additionally, 399 categories have sample sizes within 10–50, and 602 categories have sample sizes less than 10.

Figure 8 presents macro-level comparison for different models under different sample sizes. We directly take the average of precision, recall, and F_1 for categories belonging to each group. The x axis in Figure 8 shows the upper bound for the corresponding group, and the y axis in Figure 8 shows the metric's value. Figure 8 shows that the performance from all methods decreases as the sample size decreases. The proposed HABERT-RL can improve the HABERT for all groups in terms of macro-F_1. We can get a more accurate evaluation through Tables 2–4. For frequent categories, HABERT-RL has made 1.8% improvement over HABERT when the sample size is larger than 1,000 in terms of macro-F_1. Larger improvements are made when the sample size decreases. A 16.2% improvement is made for groups of 500–1,000. A 24.8% improvement is made for groups of 200–500. A 32.3% improvement is made for groups of 100–200. For categories with fewer than 100 samples, none of the models can perform well, and the improvements are not very significant. A 10.3% improvement is made for groups of 50–100. No improvement is made for groups of 10–50. The most significant improvement that our model made is within the categories with a small sample size from 50 to 200. It proves the effectiveness of the label distribution penalty.

We further present the F_1 score for selected categories under a few-shot learning setting. In Table 5, the first column shows the names of each category, and the corresponding sample size for each category is included in the parentheses. These are the categories

Figure 8. Model Comparison Under Different Sample Sizes

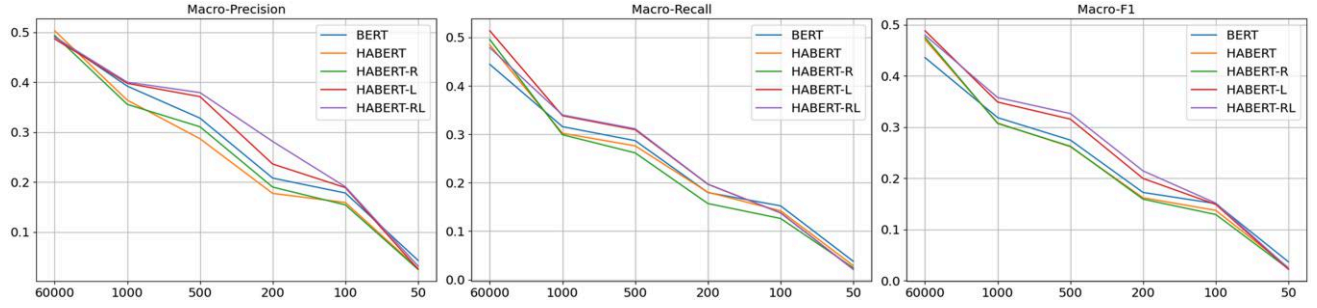


Table 2. Macro-F_1 for Different Models Under Different Sample Sizes

	Sample size					
Models	1,000–60,000	500–1,000	200–500	100–200	50–100	10–50
BERT	0.4356	0.3184	0.2746	0.1723	0.1506	0.0368
HABERT	0.4709	0.3077	0.2616	0.162	0.1373	0.0257
HABERT-R	0.4752	0.3069	0.2625	0.1592	0.1292	0.0233
HABERT-L	**0.4881**	0.3488	0.3154	0.1996	**0.149**	**0.0221**
HABERT-RL	0.4796	**0.3575**	**0.3264**	**0.2144**	**0.1515**	**0.0243**

Notes. We conduct *t*-tests of the other benchmarks with the best performers in each category. We bold the model with the best performance (top performer) and any other models that are not significantly different from the top performer.

Table 3. Macroprecision for Different Models Under Different Sample Sizes

	Sample size					
Models	1,000–60,000	500–1,000	200–500	100–200	50–100	10–50
BERT	0.4916	0.3923	0.3282	0.208	0.1782	0.0431
HABERT	**0.5034**	0.3642	0.2869	0.1774	0.1591	0.0269
HABERT-R	0.4942	0.3558	0.3108	0.19	0.1543	0.0251
HABERT-L	0.4869	**0.3979**	0.3713	0.2362	**0.1889**	0.0257
HABERT-RL	0.4905	**0.4002**	**0.3794**	**0.2815**	**0.1906**	**0.0317**

Notes. We conduct *t*-tests of the other benchmarks with the best performers in each category. We bold the model with the best performance (top performer) and any other models that are not significantly different from the top performer.

Table 4. Macrorecall for Different Models Under Different Sample Sizes

	Sample size					
Models	1,000–60,000	500–1,000	200–500	100–200	50–100	10–50
BERT	0.4443	0.3157	0.2868	0.1797	0.1522	0.0378
HABERT	0.4844	0.3021	0.2761	0.1803	0.142	0.0289
HABERT-R	0.4955	0.299	0.2616	0.157	0.1262	0.0246
HABERT-L	**0.5137**	**0.3382**	**0.3094**	**0.1969**	**0.1379**	**0.0211**
HABERT-RL	0.4787	**0.3398**	**0.3113**	**0.197**	**0.1376**	**0.0217**

Notes. We conduct *t*-tests of the other benchmarks with the best performers in each category. We bold the model with the best performance (top performer) and any other models that are not significantly different from the top performer.

Table 5. Category-Wise F_1 Score Comparison Under the Few-Shot Learning Setting

	Model				
Category	BERT	HABERT	HABERT-R	HABERT-L	HABERT-RL
IDENTIFICATION OF AIRCRAFT ON RADAR (28)	0	0	0	0	**0.381**
EXHAUST SYSTEM, CLAMP (39)	0.1053	0.1053	**0.1818**	0.1333	0
TRAFFIC ADVISORY (137)	0	0	0	0.0377	**0.2738**
WAKE TURBULENCE (99)	0.2513	0	**0.3288**	0.1311	0.3892
LUBRICATING SYSTEM, OIL FILTER/SCREEN (50)	**0.1778**	0.1176	0.069	0.1176	0
MAINTENANCE, LUBRICATION (70)	0.0408	0	0	0.069	**0.2069**
VACUUM SYSTEM (53)	0.2581	0.1053	0.1818	0.1818	**0.381**
ELEVATOR TRIM (72)	0.1488	0.1343	0.2687	0.2319	**0.2703**
LOSS OF TAIL ROTOR EFFECTIVENESS (85)	**0.252**	0.1356	0.1039	0.1356	0.16
MIXTURE CONTROL, CABLE (53)	0.2717	0.2078	0.289	0.3077	**0.3876**
ENG ASSEMBLY, BLOWER /IMPELLER/INTEGRAL (77)	0.1835	0.2509	0.3216	0.3368	**0.3821**
REMOVAL OF CONTROL/GUST LOCK (39)	0.2278	0.2278	**0.2535**	0.2069	0.2105
FLT CONTROL SYST, WING SPOILER SYSTEM (42)	0.2813	0.2951	**0.45**	0.3077	0.1429
ICE/FROST REMOVAL FROM AIRCRAFT (133)	0.2045	0.1778	0.1509	0.2046	**0.3273**
FUSELAGE, SEAT (86)	0.3173	0.2793	0.252	0.3404	**0.4673**
LANDING GEAR, STEERING SYSTEM (128)	0.1928	0.1667	0.0202	0.2792	**0.3077**

Note. We bold the model with the best performance (top performer) and any other models that are not significantly different from the top performer.

that have sample sizes between 10 and 200. By applying the proposed penalty terms, HABERT-RL can achieve significant improvements over other methods. For rare categories, such as IDENTIFICATION OF AIRCRAFT ON RADAR and TRAFFIC ADVISORY, only HABERT-RL is able to identify those from the reports. For categories that have slightly more samples, such as ICE/FROST REMOVAL FROM AIRCRAFT and LANDING GEAR STEERING SYSTEM, HABERT-RL can also make further improvements over other methods.

5. Conclusion

This work aims to discuss the feasibility of building an information extraction system for fine-level events in the aviation domain. Our work leverages the event taxonomy defined by the domain expert from NTSB and converts the problem into a hierarchical classification task. Because fine-level events have a very large label size, the long-tail distribution of the data leads to the biggest challenge for implementing an accurate algorithm. To tackle this challenge, we propose to extend the state-of-the-art BERT model with a novel multilabel hierarchical classification model. First, we develop a hierarchical attention module to introduce coarse-level information into the fine category classification process. Our experiment validates the efficiency of this module, where we achieve 20% improvements over the widely used BR-SVM in terms of micro-F_1. The second component is to provide additional supervision to the fine-level parameters through recursive regularization. The experiment shows that the proposed HABERT-R achieves the best performance among all models. Finally, we discuss how to improve the classification accuracy for categories with small training samples. We propose a label distribution penalty term in the model and evaluate the rare category identification through the macro-F_1 score. Our results show that the label distribution penalty can significantly improve the rare category classification accuracy. A 32.3% improvement is made when the sample size is between 100 and 200. In summary, we are the first work discussing the possibility of extracting fine-level information in the aviation domain for the NTSB data set.

Acknowledgments

The authors thank two experts in aviation accident report and safety analysis: Dan Larson, an aviation analyst from Metron Aviation, and Xue Ping, a computational linguist from Boeing (retired). They have worked with the team in supervising the research development and helping the team understand the current National Transportation Safety Board coding system.

References

Abedin M, Ng V, Khan L (2010) Cause identification from aviation safety incident reports via weakly supervised semantic lexicon construction. *J. Artificial Intelligence Res.* 38:569–631.

Agarwal A, Gite R, Laddha S, Bhattacharyya P, Kar S, Ekbal A, Thind P, Zele R, Shankar R (2022) Knowledge graph–deep learning: A case study in question answering in aviation safety domain. Preprint, submitted June 9, https://arxiv.org/pdf/2205.15952.

Cai L, Hofmann T (2004) Hierarchical document categorization with support vector machines. *Proc. Thirteenth ACM Internat. Conf. Inform. Knowledge Management* (Association for Computing Machinery, New York), 78–87.

Cerri R, Barros RC, de Carvalho AC (2012) A genetic algorithm for hierarchical multi-label classification. *Proc. 27th Annual ACM Sympos. Applied Comput.* (Association for Computing Machinery, New York), 250–255.

Costa EP, Lorena AC, Carvalho AC, Freitas AA, Holden N (2007) Comparing several approaches for hierarchical classification of proteins with decision trees. *Brazilian Sympos. Bioinformatics* (Springer, Berlin, Heidelberg), 126–137.

Devlin J, Chang MW, Lee K, Toutanova K (2019a) BERT: Pre-training of deep bidirectional transformers for language understanding. Preprint, submitted May 24, https://arxiv.org/pdf/1810.04805.

Devlin J, Chang MW, Lee K, Toutanova K (2019b) BERT: Pre-training of deep bidirectional transformers for language understanding. Burstein J, Doran C, Solorio T, eds. *Proc. 2019 Conf. North Amer. Chapter Assoc. Comput. Linguistics Human Language Tech. NAACL-HLT 2019*, vol. 1 (long and short papers) (Association for Computational Linguistics, Kerrville, TX), 4171–4186.

Dong T, Yang Q, Ebadi N, Luo XR, Rad P (2021) Identifying incident causal factors to improve aviation transportation safety: Proposing a deep learning approach. *J. Adv. Transportation* 2021:5540046.

Ferryman TA, Posse C, Rosenthal LJ, Srivastava AN, Statler IC (2006) What happened, and why: Toward an understanding of human error based on automated analyses of incident reports—vol. II. Technical Report No. NASA/TP–2006-213490, National Aeronautics and Space Administration, Washington, DC.

Fuller JG, Hook LR (2020) Understanding general aviation accidents in terms of safety systems. *2020 AIAA/IEEE 39th Digital Avionics Systems Conf. (DASC)* (IEEE, Piscataway, NJ), 1–9.

Gan L, Ye B, Huang Z, Xu Y, Chen Q, Shu Y (2023) Knowledge graph construction based on ship collision accident reports to improve maritime traffic safety. *Ocean Coastal Management* 240:106660.

Geng X (2016) Label distribution learning. *IEEE Trans. Knowledge Data Engrg.* 28(7):1734–1748.

Gopal S, Yang Y (2013) Recursive regularization for large-scale classification with hierarchical and graphical dependencies. *Proc. 19th ACM SIGKDD Internat. Conf. Knowledge Discovery Data Mining* (Association for Computing Machinery, New York), 257–265.

Huang W, Chen E, Liu Q, Chen Y, Huang Z, Liu Y, Zhao Z, Zhang D, Wang S (2019) Hierarchical multi-label text classification: An attention-based recurrent network approach. *Proc. 28th ACM Internat. Conf. Inform. Knowledge Management* (Association for Computing Machinery, New York), 1051–1060.

Karamanolakis G, Ma J, Dong XL (2020) Txtract: Taxonomy-aware knowledge extraction for thousands of product categories. Preprint, submitted May 1, https://arxiv.org/pdf/2004.13852.

Kierszbaum S, Lapasset L (2020) Applying distilled BERT for question answering on ASRS reports. *2020 New Trends Civil Aviation (NTCA)* (IEEE, Piscataway, NJ), 33–38.

Kiritchenko S, Matwin S, Famili AF (2004) Hierarchical text categorization as a tool of associating genes with gene ontology codes. *Eur. Workshop Data Mining Text Mining Bioinformatics (Pisa, Italy)*, 30–34.

Koller D, Sahami M (1997) Hierarchically classifying documents using very few words. Technical report, Stanford InfoLab, Stanford, CA.

Li A, Luo T, Lu Z, Xiang T, Wang L (2019) Large-scale few-shot learning: Knowledge transfer with class hierarchy. *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognition* (IEEE, Piscataway, NJ), 7212–7220.

Liu L, Zhou T, Long G, Jiang J, Zhang C (2022) Many-class few-shot learning on multi-granularity class hierarchy. *IEEE Trans. Knowledge Data Engrg.* 34(5):2293–2305.

National Transportation Safety Board (2008) Aviation coding manual. Accessed September 15, 2021, https://www.ntsb.gov/GILS/Documents/codman.pdf.

Peng H, Li J, He Y, Liu Y, Bao M, Wang L, Song Y, Yang Q (2018) Large-scale hierarchical text classification with recursively regularized deep graph-CNN. *Proc. 2018 World Wide Web Conf.* (International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE), 1063–1072.

Pereira RM, Costa YM, Silla CN Jr (2021) Toward hierarchical classification of imbalanced data using random resampling algorithms. *Inform. Sci.* 578:344–363.

Pereira FC, Rodrigues F, Ben-Akiva M (2013) Text analysis in incident duration prediction. *Transportation Res. Part C Emerging Tech.* 37:177–192.

Rao AH, Marais K (2020) A state-based approach to modeling general aviation accidents. *Reliability Engrg. System Safety* 193:106670.

Rath S, Chow JY (2022) Worldwide city transport typology prediction with sentence-BERT based supervised learning via Wikipedia. *Transportation Res. Part C Emerging Tech.* 139:103661.

Robinson SD (2018) Multi-label classification of contributing causal factors in self-reported safety narratives. *Safety* 4(3):30.

Ruggero CJ, Kotov R, Hopwood CJ, First M, Clark LA, Skodol AE, Mullins-Sweatt SN, et al. (2019) Integrating the hierarchical taxonomy of psychopathology (HiTOP) into clinical practice. *J. Consulting Clinical Psych.* 87(12):1069–1084.

Shi D, Guan J, Zurada J, Manikas A (2017) A data-mining approach to identification of risk factors in safety management systems. *J. Management Inform. Systems* 34(4):1054–1081.

Silla CN, Freitas AA (2011) A survey of hierarchical classification across different application domains. *Data Mining Knowledge Discovery* 22(1):31–72.

Srinivasan P, Nagarajan V, Mahadevan S (2019) Mining and classifying aviation accident reports. *AIAA Aviation 2019 Forum* (American Institute of Aeronautics and Astronautics, Reston, VA), 2938.

Tanguy L, Tulechki N, Urieli A, Hermann E, Raynal C (2016) Natural language processing for aviation safety reports: From classification to interactive analysis. *Comput. Indust.* 78:80–95.

Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I (2017) Attention is all you need. *Advances in Neural Information Processing Systems* (Curran Associates Inc., Red Hook, NY), 5998–6008.

Wang Y, Yao Q, Kwok JT, Ni LM (2020) Generalizing from a few examples: A survey on few-shot learning. *ACM Comput. Surveys* 53(3):1–34.

Wu TY, Morgado P, Wang P, Ho CH, Vasconcelos N (2020) Solving long-tailed recognition with deep realistic taxonomic classifier. *Eur. Conf. Comput. Vision* (Springer, Cham, Switzerland), 171–189.

Xu C, Geng X (2019) Hierarchical classification based on label distribution learning. *Proc. Conf. AAAI Artificial Intelligence* 33:5533–5540.

Yang P, Sun X, Li W, Ma S, Wu W, Wang H (2018) SGM: Sequence generation model for multi-label classification. Preprint, submitted June 15, https://arxiv.org/abs/1806.04822.

Yao W, Qian S (2021) From Twitter to traffic predictor: Next-day morning traffic prediction using social media data. *Transportation Res. Part C Emerging Tech.* 124:102938.

Yelundur A, Giannella C, Megerdoomian K, Pfeifer C (2016) Event classification in foreign language aviation reports. *Internat. J. Knowledge Engrg. Data Mining* 4(1):54–73.

Zhang X, Mahadevan S (2020) Bayesian neural networks for flight trajectory prediction and safety assessment. *Decision Support Systems* 131:113246.

Zhang X, Srinivasan P, Mahadevan S (2021) Sequential deep learning from NTSB reports for aviation safety prognosis. *Safety Sci.* 142:105390.

Zhao X, Yan H, Liu Y (2022) Event extraction for aviation accident reports through attention-based multi-label classification. *AIAA AVIATION 2022 Forum* (American Institute of Aeronautics and Astronautics, Reston, VA), 3831.