BERT for Aviation Text Classification

Xiao Jing*, Akul Chennakesavan ‡, Chetan Chandra *, Mayank V. Bendarkar §, Michelle Kirby §, Dimitri N. Mavris ¶

Aerospace Systems Design Laboratory, School of Aerospace Engineering
Georgia Institute of Technology, Atlanta, Georgia, 30332

The advent of transformer-based models pre-trained on large-scale text corpora has revolutionized Natural Language Processing (NLP) in recent years. Models such as BERT (Bidirectional Encoder Representations from Transformers) offer powerful tools for understanding contextual information and have achieved impressive results in numerous language understanding tasks. However, their application in the aviation domain remains relatively unexplored. This study discusses the challenges of applying multi-label classification problems on aviation text data. A custom aviation domain specific BERT model (Aviation-BERT) is compared against BERT-base-uncased for anomaly event classification in the Aviation Safety Reporting System (ASRS) data. Aviation-BERT is shown to have superior performance based on multiple metrics. By focusing on the potential of NLP in advancing complex aviation safety report analysis, the present work offers a comprehensive evaluation of BERT on aviation domain datasets and discusses its strengths and weaknesses. This research highlights the significance of domain-specific NLP models in improving the accuracy and efficiency of safety report classification and analysis in the aviation industry.

I. Introduction and Background

Natural Language Processing (NLP) is a prominent branch of artificial intelligence (AI), which is widely used to help computers understand or process text and human languages in the same way that humans do. NLP involves analysis of text or speech to automatically perform tasks such as text classification, information retrieval, sentiment analysis, document summarization, and machine translation, finally advancing on language understanding and natural language generation [1]. Today with limitless amount of text data being generated, one of the biggest challenge is to extract knowledge embedded in a flood of information for the applications in various areas. Utilization of text mining techniques are becoming increasingly popular among researchers, with exponentially growing applications in numerous domains. Text data mining techniques are specialized in information extraction from unstructured text data, and NLP can be seen as an powerful tool to improve information extraction techniques [2]. With the huge volumes of text data increasing, NLP has recently made great pace in various fields, one of important area is aviation [3].

There are three main application areas of NLP research in Aviation domain: Safety reports, aviation maintenance, and traffic control [1]. The data in aviation safety report is exceptionally invaluable for learning lessons from past incidents and accidents, and therefore for spotting new safety threats and bringing ways to avoid them. As in any complex system, the source of these threats can be technical, organizational, environmental or human, or a combination of these factors and so on. Manual analysis of these safety is complicated and time consuming for substantial resources. For this reason, NLP technique is beneficial for national and international regulatory authorities as well as transport company to analyze a large collection of reports [4]. Aircraft maintenance, repair, and overhaul (MRO) operations are among the most important in the aviation industry due to their importance to aviation safety and aircraft performance [1]. One of the primary uses NLP techniques have established in aircraft maintenance is assisting in the transition to predict maintenance to prevent unexpected equipment failures by continuously observing the condition of the equipment and activating advance warnings [5]. With the growth of air traffic over the past decade, air traffic control (ATC) systems are facing modernization challenges. NLP is a tool that can be used to digitize and analyze heritage Air Traffic Management (ATM) documents to plan and optimize airspace operations [6].

The present work will focus on the study of NLP applications in the aviation safety report. Compelling research has been conducted on the data in aviation safety reports. Some studies have focused in the directions of topic modelling

*Graduate Research Assistant, AIAA Student Member
†Undergraduate Student, AIAA Student Member
‡Research Engineer II, AIAA Member
§Senior Research Engineer, ASDL, School of Aerospace Engineering, Georgia Tech, AIAA Senior Member
¶S.P. Langley Distinguished Regents Professor and Director of ASDL, AIAA Fellow

Copyright © 2023 by Xiao Jing, Akul Chennakesavan, Chetan Chandra, Mayank V. Bendarkar, Michelle Kirby, Dimitri N. Mavris. Published by the American Institute of Aeronautics and Astronautics, Inc., with permission.
Addressing this gap is crucial to understand the benefits of leveraging domain-specific NLP models in the aviation industry, which could potentially lead to improved accuracy and efficiency in safety report classification and analysis. One such application is in the classification of aerospace requirements. Ray et al. proposed the aeroBERT-Classifier, a fine-tuned BERT model specifically designed to classify aerospace requirements into design, functional, or performance requirements.

As stated above, while transformer-based models such as BERT have made significant advancements in various domains, the application of such models for multi-label classification tasks in aviation safety report analysis has not been thoroughly investigated. The research gap lies in the lack of a comprehensive study comparing the performance of generic and domain-specific BERT models for multi-label classification of anomaly events in aviation safety data. Addressing this gap is crucial to understand the benefits of leveraging domain-specific NLP models in the aviation industry, which could potentially lead to improved accuracy and efficiency in safety report classification and analysis.

In light of the transformer’s potential in analyzing aviation text data, this study aims to investigate and compare the performance of BERT-base-uncased and a custom aviation-specific pretrained BERT model (Aviation-BERT) in multi-label classification tasks applied to aviation safety reports from the ASRS dataset. The goal is to identify the most effective model for accurately classifying and predicting anomaly events in the aviation domain. By determining the most accurate and efficient model, this research contributes to improving the understanding and prediction of anomaly events in the aviation industry, potentially leading to enhanced safety measures and procedures. The long term goal of this effort is to develop an NLP pipeline that can help quantify the safety of the national airspace system (NAS) using the Integrated Safety Assessment Model (ISAM). It can also be leveraged to quantify system failure rates for use in safety methods developed in prior work by the authors.

II. Methodology

The methodology for developing the aviation text classification task using BERT is illustrated in Fig.1. The following sections describe each step in detail.
A. Data Sources

With the advent of modern sensors and computing, there exists a variety of sources to analyze flight data. Data-driven frameworks can be utilized to promote flight safety, which prove to be efficient in discovering patterns and anomalies. The Aviation Safety Reporting System (ASRS)\(^\ast\) and National Transportation Safety Board (NTSB)\(^\dagger\) are just a few examples of databases that contain an extensive number of aviation accident and incident reports.

The ASRS program has been collecting reports of potentially unsafe incidents that have occurred in the aviation industry, with the goal of finding areas of improvement in the national aviation industry. Reports come from varied individuals, including pilots, maintenance workers, air traffic controllers, flight attendants, and more. Since the 1980s, over 1.78 million reports have been processed by the ASRS program\(^{[22]}\). The database allows for the extraction of flight narratives, as well as numerous metadata filters. The most important aspects of these narratives that are used in our text analysis are the “Narrative” and “Synopsis” metadata columns. Respectively, they contain a first-person narrative of the incident and a briefly summarized version. While the present work is focused solely on the ASRS datasets, additional aviation text data sources are listed next and will be explored in the future.

The NTSB Aviation Accident Database mainly includes information about civil aviation accidents, located within the United States and its territories. The NTSB reports are written by air accident investigators, and include probable cause, factual and analysis narratives. Machine learning techniques have been applied to the NTSB database in multiple works, including to derive insights about helicopter accident rates by categories\(^{[23]}\), applying LSTM networks for safety prognosis\(^{[24]}\), or for question answering using a hybrid of knowledge graphs and deep learning\(^{[25]}\), among numerous others. The Aircraft Communications, Addressing and Reporting System (ACARS)\(^{[26]}\) consists of a digital link system for transmitting messages between an aircraft and ground stations. This database is proprietary to airlines themselves. The Aviation Safety Action Program (ASAP)\(^{[27]}\) was initiated by the FAA to provide a way for pilots, maintenance technicians, and aircraft operators to submit safety reports, which would then be reviewed by a committee to implement corrective actions. The World Aircraft Accident Summary (WAAS) Subset provides brief details of all known major operational accidents to jet/turboprop engines as well as helicopters around the world.

B. Data Preprocessing

1. Introduction to ASRS datasets

ASRS is a semi-structured data set which includes textual data in the form of narratives and synopses. The narratives depict the events from the perspective of reporters. For each narrative, the synopsis is generated is generated by ASRS staff member to summarize the narrative from a safety point of view. In addition, the ASRS data also involves the metadata which is either created by reporter or staff of ASRS. Reporter-generated metadata includes structured information about the context of the event (e.g. issues relating to aircraft, airports, airspace structure, ATC equipment/navigation facilities, tooling, or incorrect/not installed/unavailable aircraft parts)\(^{[22]}\). In this study, we utilize narrative and anomaly event data from the ASRS dataset. Each narrative in the ASRS data is associated with multiple corresponding anomaly events, which are delineated by semicolons. For a visual representation of the structure of the ASRS dataset, please refer to the Fig. 2.

\(^{\ast}\)NASA ASRS Program: https://asrs.arc.nasa.gov/  
\(^{\dagger}\)https://www.ntsb.gov/
2. ASRS Multi-label Encoding

In this study, we combined ASRS datasets spanning from January 1988 to December 2021. To efficiently manage the ASRS anomaly events data, we automated the multi-label encoding process. From the dataset, we extracted multi-label data comprising 218,286 narrative reports and over 100 unique labels. To render these labels compatible with BERT models, we need to convert them into a binary format and allocate a unique position for each label within the multi-label vector. In this representation, a "1" at a specific position signifies the presence of a label, while a "0" indicates its absence. This binary encoding scheme enabled the application of BERT models for multi-label classification tasks, ensuring the models’ ability to effectively discern the intricate relationships among various anomaly events in the aviation safety reports. The multi-label encoding process is illustrated in the Fig 3. However, multiple of the over 100 anomaly labels can apply to each narrative resulting in a large multi-label problem with a limited training dataset. To solve this issue, ASRS anomaly events are mapped to ICAO occurrence categories as is explained next.

3. Mapping ASRS Labels to ICAO Labels

The most important aspects of the ASRS narratives that are used in our text analyses are the “Narrative” and “Anomaly” metadata columns. Respectively, they contain a first-person narrative of the incident and a small number of phrases to summarize the type of incident that had occurred. These phrases serve as individual labels that can be utilized to categorize the different narratives by incident type. Since the individual labels are nonexclusive in multilabel classification, each report may contain no anomaly, one anomaly, or a set of anomalies associated with the report.

To assist with classification, the ASRS anomalies were mapped to ICAO Occurrence Categories [28]. The International Civil Aviation Organization (ICAO) has developed common taxonomies and definitions to improve the aviation community’s ability to address safety concerns in the industry. To reduce the number of unique labels, the occurrence categories were used as a basis instead of the ones provided by the ASRS reports. The ASRS anomalies were mapped to their corresponding ICAO occurrence category, consolidating them into 24 labels and an additional “OTHER” label as a catch-all. Reports that did not properly map to a valid occurrence category would be placed into the “OTHER” label. These along with any reports that did not have an anomaly associated at all would be discarded during text preprocessing. Table 1 shows a subset of the mapping done from various ASRS labels to ICAO occurrence categories. Note that some of the categories have many ASRS labels corresponding to them, such as “NAV”.

There were a few benefits to using the ICAO occurrence categories for this study. The main reason was that consolidating the ASRS narratives into a smaller number of labels would help reduce the complexity associated with utilizing many unique labels, making sure that related anomalies would be grouped together. This would lead to an improvement in the model’s performance and accuracy. In addition, using a set of standard aviation categories in this study would help enable consistency when needed to compare to other aviation text datasets or models. The partial
Mapping ASRS to ICAO is shown in the Table 1.

<table>
<thead>
<tr>
<th>ASRS Labels</th>
<th>ICAO Occurrence Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>'Ground Event / Encounter Person / Animal / Bird'</td>
<td>ANIM</td>
</tr>
<tr>
<td>'Inflight Event / Encounter Bird / Animal'</td>
<td></td>
</tr>
<tr>
<td>'Airspace Violation All Types'</td>
<td>ATM</td>
</tr>
<tr>
<td>'ATC Issue All Types'</td>
<td></td>
</tr>
<tr>
<td>'Other Airspace Violation Entry or Exit'</td>
<td></td>
</tr>
<tr>
<td>'Flight Deck / Cabin / Aircraft Event Passenger Misconduct'</td>
<td>CABI</td>
</tr>
<tr>
<td>'Flight Deck / Cabin / Aircraft Event Passenger Electronic Device'</td>
<td>CTOL</td>
</tr>
<tr>
<td>'Inflight Event / Encounter Object'</td>
<td></td>
</tr>
<tr>
<td>'Ground Event / Encounter Fuel Issue'</td>
<td>FUEL</td>
</tr>
<tr>
<td>'Inflight Event / Encounter Fuel Issue'</td>
<td></td>
</tr>
<tr>
<td>'Weather Elements / Visibility Hail'</td>
<td>ICE</td>
</tr>
<tr>
<td>'Deviation / Discrepancy - Procedural Maintenance'</td>
<td>MAINT</td>
</tr>
<tr>
<td>'Flight Deck / Cabin / Aircraft Event Illness / Injury'</td>
<td>MED</td>
</tr>
<tr>
<td>'ATC Issues ATC Issues'</td>
<td>NAV</td>
</tr>
<tr>
<td>'Deviation / Discrepancy - Procedural Clearance'</td>
<td></td>
</tr>
<tr>
<td>'Deviation / Discrepancy - Procedural Security'</td>
<td>SEC</td>
</tr>
<tr>
<td>'Inflight Event / Encounter Laser'</td>
<td></td>
</tr>
</tbody>
</table>

4. Imbalanced Multi-label Datasets

We conducted an in-depth analysis of our datasets, as depicted in the Fig 4. On the left side, the figure displays the frequency of all labels, while the right side presents a heatmap illustrating label correlations. Upon examining the dataset, we observed a significant label imbalance, with certain labels having considerably higher frequencies than others. This uneven distribution of label frequencies can potentially impact the model’s performance, as training machine learning models on imbalanced datasets can cause classifiers to be biased towards the majority class, resulting in suboptimal performance and poor predictive accuracy in data-driven models [29].

Moreover, we identified weak correlations between the labels, as evidenced by the low values in the correlation heatmap. This observation suggests that the multi-label classification task is inherently complex, given that the weak correlations between labels imply that the relationships among the anomaly events are intricate and could pose challenges for the model to accurately capture [30].

5. Strategies to Address Imbalanced Multi-label Classification

In this paper, our primary focus is to compare the performance of the bert-base-uncased model and our custom aviation pretrained model (Aviation-BERT) [16] for multi-label classification tasks applied to aviation safety reports. To ensure a fair comparison, we address the issue of imbalanced multi-label datasets by employing appropriate techniques to mitigate the potential impact of label imbalance on model performance.

Sophisticated methods such as deep learning techniques [31] might be required to deal with weak correlation between labels in the present data. This could complicate the comparison between the two models. Therefore, in this study, we focus on addressing the label imbalance issue. Future work could explore more advanced methods to handle weak
correlations between labels and investigate their impact on the performance of both the bert-base-uncased model and the custom aviation pre-trained model.

Two strategies to address the challenges posed by imbalanced multi-label classification were evaluated: custom oversampling and Focal Loss. Both methods aim to handle class imbalance but approach the problem differently, and their effectiveness may vary depending on the specific characteristics of the dataset and the task. An initial experiment was conducted with oversampling, a widely used technique for tackling imbalanced classes. We developed a custom oversampling function that iterates over each label to detect imbalances between majority and minority classes, replicating instances of the minority class to achieve a balanced distribution for training. However, this strategy resulted in marginal improvements in the multi-label classification F1 score, with some labels still showing an F1 score of 0 despite high accuracy, indicating a continued imbalance. This suggests that the oversampling approach may not adequately address the intricacies of the multi-label classification task in our specific context. Consequently, Focal Loss was utilized, which is a cost-sensitive learning approach specifically designed to tackle class imbalance by concentrating on difficult examples and down-weighting simpler ones during loss computation [32]. This method entails implementing a custom Focal Loss function, modifying the model, and training with Focal Loss, with detailed implementation discussed in the subsequent fine-tuning section. By employing Focal Loss, we aim to capitalize on its ability to more effectively address the inherent challenges present in our dataset, particularly by focusing on difficult examples that were not sufficiently handled using the oversampling strategy.

C. BERT

Bidirectional Encoder Representations from Transformers (BERT) was developed by Google in 2018 [33]. BERT architecture consists of several Transformer encoders stacked together. Each Transformer encoder encapsulates two sub-layers: a self-attention layer and a feed-forward layer. In this study, there are two steps in our framework: pre-training and fine-tuning. During pre-training, the model is trained on unlabeled data over different pre-training tasks. A pre-trained version of BERT on the english wikipedia and bookcorpus (3.3 Billion words) is freely available‡. During fine-tuning for the supervised task, the BERT model is first initialized with the pre-trained parameters, and all of the parameters are fine-tuned using labeled data from the downstream tasks.

1. BERT Input and Output

BERT model expects a sequence of tokens (words) as an input. In each sequence of tokens, there are two special tokens that BERT would expect as an input:

- \([CLS]\): This is the first token of every sequence, which stands for classification token.

‡ https://github.com/google-research/bert, accessed 31 Oct 2022
• \([SEP]\): This is the token that makes BERT know which token belongs to which sequence. This special token is mainly important for a next sentence prediction task or question-answering task. If we only have one sequence, then this token will be appended to the end of the sequence.

The maximum number of tokens that can be fed into BERT model is 512. If the tokens in a sequence are less than 512, we can use padding to fill the unused token slots with \([PAD]\) token. If the tokens in a sequence are longer than 512, then we need to do a truncation. BERT model then will output an embedding vector of size 768 in each of the tokens. We can use these vectors as an input for different kinds of NLP applications, whether it is text classification, next sentence prediction, Named-Entity-Recognition (NER), or question-answering [34].

2. \textbf{BertTokenizer and Encoding the Data}

We first transform our text into the format that BERT expects by adding \([CLS]\) and \([SEP]\) tokens. We can do this with 
\textit{BertTokenizer} class with \textit{Transformers} library from Hugging Face - open source library [35]. The \textit{BertTokenizer} takes care of all of the necessary transformations of the input text such that it’s ready to be used as an input for our BERT model. It adds \([CLS]\), \([SEP]\), and \([PAD]\) tokens automatically. Here are some sample for the \textit{BertTokenizer} parameters setting:

• \textit{Padding} : to pad each sequence to the maximum length that you specify.
• \textit{Max\_Length} : the maximum length of each sequence. In this study, the maximum sequence length for each input is set to 256, which aligns with the configuration used during the training process for our custom pre-trained model, Aviation-BERT.
• \textit{Truncation} : if True, then the tokens in each sequence that exceed the maximum length will be truncated.

3. \textbf{Aviation-BERT: An aviation specific pre-trained model}

Aviation-BERT is a preliminary aviation domain-specific BERT model created using mixed domain pretraining in a parallel effort [16]. A BERT-Base-Uncased model pre-trained on English Wikipedia and BookCorpus was accessed through the Hugging Face Transformers library [16]. This was then further pre-trained on over 500,000 NTSB and ASRS text narratives. The methodology for the creation of Aviation-BERT involves pre-training using Masked Language Modelling (MLM) involving whole word masking and updating the BERT-base vocabulary to incorporate aviation-specific terms among other considerations. Interested readers are directed to the work of Chandra et al. [16] for details.

D. \textbf{Fine-tuning BERT}

The two models considered in the present work, Bert-base-uncased and Aviation-BERT’s performance in the multi-label classification task applied to ASRS dataset are compared after fine-tuning. The methodology for fine-tuning BERT is illustrated in Fig [5] the following sections describe each step in detail.

1. \textbf{Training Loop}

PyTorch’s machine learning framework [36] streamlines the training loop process, which involves training the model for a predetermined number of epochs using the provided DataLoader, optimizer, and scheduler. For the fine-tuning process, datasets are combined with a sampler, resulting in an iterable over the specified dataset. The training and validation datasets are generated by dividing the balanced training dataset into an 80% training set and a 20% validation set, ensuring a representative sample for model training and evaluation. Our training leverages the ‘BertForSequenceClassificationWithFocalLoss’ model, which is a modified version of the ‘BertForSequenceClassification’ class that incorporates the focal loss function to tackle imbalanced datasets, as elaborated in Sec. I.B.5. The ‘BertForSequenceClassification’ class is a part of the Hugging Face Transformers library [37] and is designed for sequence classification tasks utilizing the BERT model. The two models were fine-tuned for a total of 5 epochs, with early stopping implemented when the validation loss increased to prevent overfitting and enhance model generalization. A batch size of 16 was employed during the training process. Furthermore, a custom ASRSDataset class is established to manage input data for training and evaluation.

\footnotesize{https://huggingface.co/bert-base-uncased
https://github.com/huggingface/transformers}
2. Loss Function

In this study, as discussed in Sec. II.B.5, Focal Loss is employed to address the class imbalance problem by down-weighting easy examples and focusing on hard examples during training [32]. The Focal Loss function is characterized by two main parameters: alpha ($\alpha$) and gamma ($\gamma$). Alpha ($\alpha$) is a scalar that balances the importance of positive and negative examples, while gamma ($\gamma$) is a focusing parameter that controls the rate at which easy examples are down-weighted. Focal loss is calculated in the forward method of the class. It is a modified version of the cross-entropy loss [38], which introduces a weighting factor to concentrate on hard-to-classify examples. Given input logits, the binary cross-entropy with logits (BCE_loss) is computed first. The probability of each example belonging to the target class ($p_t$) is then calculated by taking the exponent of the negative BCE_loss. The Focal Loss ($FL$) in multi-label classification problem is computed based on the following equation [32]:

$$FL (p_t) = -\alpha_t (1 - p_t)^\gamma \cdot \text{BCE} \_\text{loss}$$

3. [CLS] Token for Classification

In the BERT model, the [CLS] token plays a crucial role in classification tasks. As illustrated in Fig 5, a single input vector $C$ captures the overall meaning and is used for classification. The 'BertForSequenceClassification' class mentioned in the above section, relies on the output corresponding to the [CLS] token for classification by default. To enhance the classification process, a fully-connected layer is incorporated into the model. In contrast to multiclass or single-label classification, a sigmoid activation function replaces the original softmax function for the final output. The sigmoid function is preferred in this scenario as it allows for independent probabilities in a multi-label classification setting, where each class is considered separately [39]. This enables the model to produce a more accurate representation of the relationships between different classes.
4. Optimizer & Scheduler

In our study, we employed the AdamW optimizer, a variant of the widely-used Adam optimizer with incorporated weight decay, to facilitate superior convergence and generalization performance compared to the conventional Adam optimizer [40]. The scheduler strategy encompasses a warm-up phase [41], which dynamically adjusts the learning rate throughout the training process. Initially, the learning rate escalates linearly from 0 to a predefined initial learning rate during the warm-up phase, followed by a linear decrease until it ultimately reaches 0. This learning rate scheduler effectively enhances model convergence and mitigates overfitting.

5. Learning Rate

To identify the optimal learning rate, we initially employed the LRFinder, a technique designed to estimate the learning rate that maximizes model performance by analyzing the model’s loss curve with respect to different learning rates [42]. However, as the estimated learning rate may not be ideal for our specific case, we also manually adjusted and fine-tuned the learning rate to ensure its appropriateness for our task. Following this process, we determined that a learning rate of $2e^{-5}$ was optimal for both the Bert-Base-Uncased model and the Aviation-BERT model, facilitating a fair comparison between the two models.

6. Threshold

In a multi-label classification task, determining a suitable threshold is crucial for transforming the model’s output probabilities into binary predictions. This process aids in managing imbalanced datasets and optimizing the performance metric [30]. In our study, a range of threshold values from 0 to 1 is defined. These thresholds are used to convert the model’s output probabilities into binary predictions. The optimal threshold value can be selected based on a performance metric like F1 score. Different threshold values are useful for dealing with imbalanced data, as they allow for better control over the trade-off between precision and recall.

7. Performance Metrics

In this study, we have evaluated the performance of the Aviation-BERT and Bert-base-uncased models using multiple evaluation metrics to ensure a comprehensive understanding of their effectiveness in multi-label classification tasks applied to aviation safety reports. The evaluation metrics utilized in our analysis include:

- **Per-label F1 Score**: This metric computes the F1 Score [43] for each label separately, allowing us to analyze the models’ performance on a label-by-label basis and identify areas of strength and weakness.
- **Micro F1 Score**: This metric calculates the F1 Score by considering the total true positives, false positives, and false negatives across all labels, providing an overall performance measurement for the models.
- **Macro F1 Score**: This metric computes the F1 Score for each label individually and then takes the average, giving equal weight to each label regardless of their frequency. This score offers a balanced perspective on the models’ performance across all labels, even those that are less frequent.
- **ROC-AUC Score**: This metric computes the area under the Receiver Operating Characteristic (ROC) curve for each label [44], which is a graphical representation of the trade-off between sensitivity (true positive rate) and specificity (false positive rate) at various threshold settings. The ROC-AUC score is a useful metric for evaluating the models’ ability to distinguish between the positive and negative classes for each label. The ROC-AUC score is particularly useful for evaluating classifiers in multi-label settings or when dealing with imbalanced datasets because it is less sensitive to class imbalance than other metrics. By providing both F1 scores and ROC-AUC scores, we offer a more comprehensive evaluation of two BERT models’ performance in multi-label classification tasks applied to ASRS datasets.
- **Hamming Loss**: This metric calculates the fraction of misclassified instances (labels) relative to the total number of instances [45], providing an insight into the overall error rate of the models. A lower Hamming loss indicates better performance of the classifier, as it means that fewer labels have been misclassified.

By employing these evaluation metrics, we aim to provide a robust and in-depth comparison of the Aviation Pretrained and Bert-base-uncased models’ performance in the complex multi-label classification task applied to aviation safety reports from the ASRS dataset.
8. Compute Resources

The multi-label fine tuning exercise on BERT-base-uncased and Aviation-BERT was run on a node with 12 CPUs, one NVidia Tesla V-100 GPU, 32 GB RAM using the PACE supercomputing cluster at Georgia Tech and took about 8 hours to run [46]. Optimizing the code for parallel computing to make it more efficient will be explored in future work.

III. Results and Discussions

In our study, we compared the performance of models on both the full set of labels and the top 10 most frequent labels for obtaining a comprehensive understanding of the models’ capabilities.

A. Full set of Labels

![Fig. 6 F1 score of Bert-base-uncased and Aviation-BERT models for full set of labels](image)

Our analysis of the performance of the Aviation-BERT [10] and the Bert-Base-Uncased model on multi-label classification tasks revealed differences in their effectiveness. In terms of F1 scores for each label in Fig 6, Aviation-BERT outperformed the Bert-Base-Uncased model in the majority of cases, indicating better overall performance. For instance, the Aviation Pretrained model demonstrated superior F1 scores in the 'ATM', 'CABIN', 'CTOL', 'FIRE', 'LOC-G', 'MAC', 'NAV', among others.

When comparing the ROC-AUC scores shown in Fig 7, Aviation-BERT consistently exhibited higher values than the Bert-Base-Uncased model, suggesting that the Aviation-BERT model is better at distinguishing between true positive and false positive instances. This is particularly evident in the 'ARC', 'ATM', 'CABIN', 'CFIT', 'CTOL', 'FUEL', 'LOC-G', 'LOC-I', among others.

<table>
<thead>
<tr>
<th>Models</th>
<th>Micro F1 Score</th>
<th>Macro F1 Score</th>
<th>Hamming Loss</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bert-base-uncased</td>
<td>0.7654</td>
<td>0.6562</td>
<td>0.0447</td>
</tr>
<tr>
<td>Aviation-BERT</td>
<td>0.7748</td>
<td>0.6738</td>
<td>0.0417</td>
</tr>
</tbody>
</table>
An evaluation of the micro F1 score, macro F1 score, and Hamming loss shown in Table 2 further supports the superiority of the Aviation-BERT model. Aviation-BERT achieved a micro F1 score of 0.7748, a macro F1 score of 0.6738, and a Hamming loss of 0.0417, compared to the Bert-Base-Uncased model’s scores of 0.7654, 0.6562, and 0.0447, respectively. These results indicate that Aviation-BERT offers improved overall classification performance and lower misclassification rates on the ASRS dataset.

B. Top-10 Frequency Labels

The results obtained after focusing on the top 10 high-frequency labels demonstrate improved performance for both
the Aviation-BERT and Bert-base-uncased models. By removing the low-frequency labels, we can partially address issue of class imbalance that affects the model’s performance. This approach allows us to concentrate on the most common labels and provides a clearer comparison of the models’ capabilities for these specific labels. In an evaluation of the performance depicted in Figure 8 and Figure 9, it can be observed that Aviation-BERT consistently achieves higher F1 scores and ROC-AUC scores for each label when compared to the BERT-base-uncased model.

The Aviation-BERT model outperforms the Bert-base-uncased model in terms of both Micro F1 Score (0.7858 vs. 0.7772) and Macro F1 Score (0.7453 vs. 0.7350) shown in Table 3, suggesting that Aviation-BERT has better overall performance for multi-label classification tasks in the aviation domain. The Hamming Loss for Aviation Pretrained model (0.0833) is also lower than that for the Bert-base-uncased model (0.0864), indicating fewer misclassifications.

Table 3 Overall performance of Bert-base-uncased and Aviation-BERT models for Top-10 Frequency labels

<table>
<thead>
<tr>
<th>Models</th>
<th>Micro F1 Score</th>
<th>Macro F1 Score</th>
<th>Hamming Loss</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bert-base-uncased</td>
<td>0.7772</td>
<td>0.7350</td>
<td>0.0864</td>
</tr>
<tr>
<td>Aviation-BERT</td>
<td>0.7858</td>
<td>0.7453</td>
<td>0.0833</td>
</tr>
</tbody>
</table>

When comparing these results to the ones obtained using the full set of labels, we can observe that the Micro F1 Score, Macro F1 Score, and Hamming Loss have all improved for both models. This suggests that focusing on high-frequency labels allows the models to perform better, as they are less affected by the challenges posed by an imbalanced dataset. Consequently, this approach facilitates a more accurate comparison between the two models for the most relevant labels in the ASRS datasets.

In summary, our analysis demonstrates that Aviation-BERT exhibits superior performance in multi-label classification tasks applied to aviation safety reports when compared to the Bert-base-uncased model, making it a more suitable choice for accurately classifying and predicting anomaly events in the aviation domain.

Although the Aviation-BERT’s micro F1 score and macro F1 score may not seem exceptionally high at first glance, it is essential to recognize the complexity and challenges associated with multi-label classification tasks. In this context, the achieved scores represent a considerable accomplishment. Multi-label classification involves predicting multiple labels for each input instance, which can be difficult due to the intricate relationships between labels and the imbalanced distribution of labels within the datasets. Therefore, these scores should be interpreted within the context of the task’s inherent complexity and challenges.

In addition, the ROC-AUC score provides a complementary perspective on a model’s classification performance. While the F1 score takes into account both precision and recall, the ROC-AUC score evaluates a model’s ability to discriminate between true positive and false positive instances across a range of threshold values. We observe that the majority of the ROC-AUC scores for Aviation-BERT labels fall within a good range, indicating the model is able to
assign the correct labels to each instance with a high degree of confidence. By presenting both F1 scores and ROC-AUC scores, we furnish a comprehensive assessment of the Aviation-BERT model’s performance in multi-label classification tasks applied to ASRS datasets, showcasing its efficacy in spite of the intrinsic complexity and challenges associated with the task.

IV. Conclusions and Future Works

The present work demonstrated the effectiveness of Aviation-BERT in comparison to the Bert-base-uncased model for multi-label classification tasks applied to aviation safety domain. By addressing the challenges posed by imbalanced datasets, we have shown that the Aviation-BERT model outperforms the Bert-base-uncased model across a range of evaluation metrics, including Micro F1 Score, Macro F1 Score, Hamming Loss, per-label F1 Score, and ROC-AUC Score. Aviation-BERT’s superior performance suggests that leveraging domain knowledge and fine-tuning BERT models for specific applications can lead to improved outcomes in real-world scenarios. Such fine-tuned models that leverage aviation domain specific understanding can make an immense impact in processing millions of text documents to distill and extract important knowledge, thus making subject matter experts more productive.

Although some performance metrics, such as F1 score, might not be remarkably impressive, it is important to consider that Aviation-BERT has been trained on only about 500,000 text narratives, a number that can be increased to improve its aviation language understanding. Future work will involve further pre-training Aviation-BERT using additional aviation specific text data, potentially from multiple publicly available sources. Further improvements in the fine-tuning pipeline will be explored, like more advanced methods to handle weak correlations between labels to improve performance. As is commonly said, what distinguishes a good model from bad one is often just the volume of data available to train it on. In that vein, additional datasets like NTSB for multi-label classification tasks involving ICAO occurrence categories will be incorporated. By carefully curating and refining the data, we can further improve the performance of our models and potentially uncover new insights or relationships between labels. This may lead to a more comprehensive understanding of the factors affecting multi-label classification performance in the aviation domain. There is also immense scope to fine-tune Aviation-BERT on a variety of other downstream tasks like question answering that will be explored.

Acknowledgments

This research was supported in part through research cyberinfrastructure resources and services provided by the Partnership for an Advanced Computing Environment (PACE) at the Georgia Institute of Technology, Atlanta, Georgia, USA. This work was funded by the US Federal Aviation Administration (FAA) as a part of the Top-down Safety Risk Modeling For Facilitating Bottom-up Safety Risk Assessment project under FAA Award Number: 692M15-21-F-00225.

Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the FAA. The authors would like to thank Huasheng Li, Shane Bertish, Tom Tessitore, Anis Ommezine, Dereck Wilson, Poornima Shankar, Alan Durston, and Chris Andrzejczak for their guidance and support.

References


