Human Factors in Transportation, Vol. 95, 2023, 620–628
https://doi.org/10.54941/ahfe1003847

Project AVIAN-S: Development of a Natural Language Processing Model for Analyzing Aviation Safety Event Reports

R. Jordan Hinson, Edward T. Bynum, Amelia Kinsella, Katherine Berry, and Michael Sawyer

Fort Hill Group, Washington, DC, 20024, USA

ABSTRACT

Voluntary Safety Reporting Programs (VSRPs) allow civil aviation authorities, operators, and manufacturers to actively monitor and identify potential safety issues within their operations. These first-hand reports enable organizations to develop and implement safety and efficiency improvements based on front-line observations. The National Aeronautics and Space Administration (NASA) operates the Aviation Safety Reporting System (ASRS) to empower the aviation industry and its participants to report observed safety problems, discrepancies, or deficiencies. ASRS receives, processes, and publicly releases thousands of reports annually. For example, 6,428 ASRS reports are currently available detailing events that occurred in 2019; any interested party can download these ASRS reports and associated data. Often, researchers and analysts will then read and manually label factors of interest in each report to gain safety insights. This manual process can be labor-intensive and relies on the ongoing efforts of subject-matter experts. The full potential of various voluntary safety reporting data can be difficult to realize due to the limited resources available to analyze and summarize these data. New machine learning techniques involving natural language processing offer opportunities to assess and label factors of interest within safety reports more efficiently and effectively. A novel machine learning model has been developed and trained to identify human factors issues within aviation safety reports. The AVIAN-S model has been built and iteratively trained on over 50,000 rows of manually classified aviation safety reporting data. The model uses machine learning and natural language processing to automate the process of labeling aviation safety reporting data and codifying reporter narratives according to an established human factors taxonomy. This paper will describe lessons learned from the initial model development iterations and present interim results of the model as applied across a set of sample event reports. The paper will further discuss the challenges and implications of using natural language processing to identify human factors issues emerging from this or other large aviation safety reporting data sets.

Keywords: Aviation safety, Human factors, Artificial intelligence, Machine learning, Natural language processing

INTRODUCTION

Aviation is the safest form of transportation (ICAO, 1999). The probability of an accident, specifically one involving fatalities, is extremely low; therefore,

reactive analyses of aviation safety accidents only provide a partial picture of the aviation industry and aviation safety (Oster Jr. et al., 2013). When analyzing aviation safety data, it is important to incorporate other incidents and events that occur to create a more holistic view of aviation safety.

One way of doing this is by collecting and reviewing aviation safety event reports. The Federal Aviation Administration (FAA) established numerous aviation Voluntary Safety Reporting Programs (VSRPs) that allow individuals to file reports on specific aviation events that have occurred, as well as noting observed safety problems. One such program is Aviation Safety Reporting System (ASRS). This system established a partnership between the National Aeronautics and Space Administration (NASA) and the FAA and allows for confidential reporting from any National Airspace System (NAS) participant, including pilots, cabin crew, maintenance technicians, ground personnel, and air traffic controllers. These safety reports are analyzed by NASA personnel and are used to identify system-level safety risks. The ASRS program has been in operation since 1976 and, to date, over 1.7 million reports have been filed and analyzed (NASA, 2023).

While NASA conducts an initial analysis of these reports, often researchers and analysts will download a subset of reports to conduct their own analyses based on their needs. To do this, researchers will read and manually label factors of interest in each report to gain safety insights. This process can be labor-intensive and relies on the ongoing efforts of subject-matter experts (SMEs) to manually read and label those reports. The full potential of ASRS reports and other voluntary safety reporting data can be difficult to realize due to the limited resources available to analyze and summarize these data. New machine learning (ML) techniques involving natural language processing (NLP) offer opportunities to assess and label factors of interest within safety reports more efficiently and effectively.

Some aviation researchers have initially examined the application of NLP to aviation and specifically, ASRS. Kierszbaum and Lapasset (2020) used NLP to extract the event date from the free text portion of ASRS reports with relative success. Those same researchers have continued by highlighting the imporatance of using a pre-trained, aviation model in the ASRS application of NLP due to the unique language of aviation (Kierszbaum, Klein, & Lapasset, 2022). Other researchers have utilized NLP of aviation safety reports to examine flight delays in ASRS (Miyanmoto, Bendarkar, & Mavris, 2022) and probable cause in National Transportation Safety Board (NTSB) reports (Jonk et al., 2023). This research along with other NLP research emphasizes the potential application of NLP in aviation safety event reporting.

Our team has developed and trained a model utilizing VSRP data that was manually labeled by SMEs. This AVIation Analytic Neural network for Safety events (AVIAN-S) model incorporates ML and NLP to automate the identification and labeling of human factors (HF) taxonomy items within VSRP reports. This project is an independent self-funded research effort. Views and results are those of Fort Hill Group and do not represent opinions or views of the FAA or NASA.

SAFETY EVENT REPORTING ANALYSIS AND DEVELOPMENT OF MODEL TRAINING DATA

This paper explores the creation of a model aimed at codifying aviation safety report data as a means of gaining meaningful safety insights within the field of aviation. As such, a manual coding process for analyzing safety reports was developed. This process resulted in training data that were utilized as model inputs. The first step in this process was identifying an appropriate HF taxonomy to be applied to the safety event reports.

AirTracs Taxonomy

The AirTracs framework follows a tiered approach that promotes the identification of HF causal trends by allowing factors from the immediate operator context to agency-wide influences to be traced to individual events while still being able to identify HF patterns. AirTracs is a published and industry-applied aviation human factors taxonomy utilized for analyzing aviation data. Our goal was to apply the established AirTracs taxonomy to aviation safety event reports using the novel AVIAN-S AI model.

As depicted in Figure 1, the first tier is "Operator Acts"; the second tier is "Operating Context"; the third tier is "Facility Influences"; and the fourth tier is "Agency Influences." Operator Acts addresses those factors most closely linked to the actual safety event and describes the actions or inactions of

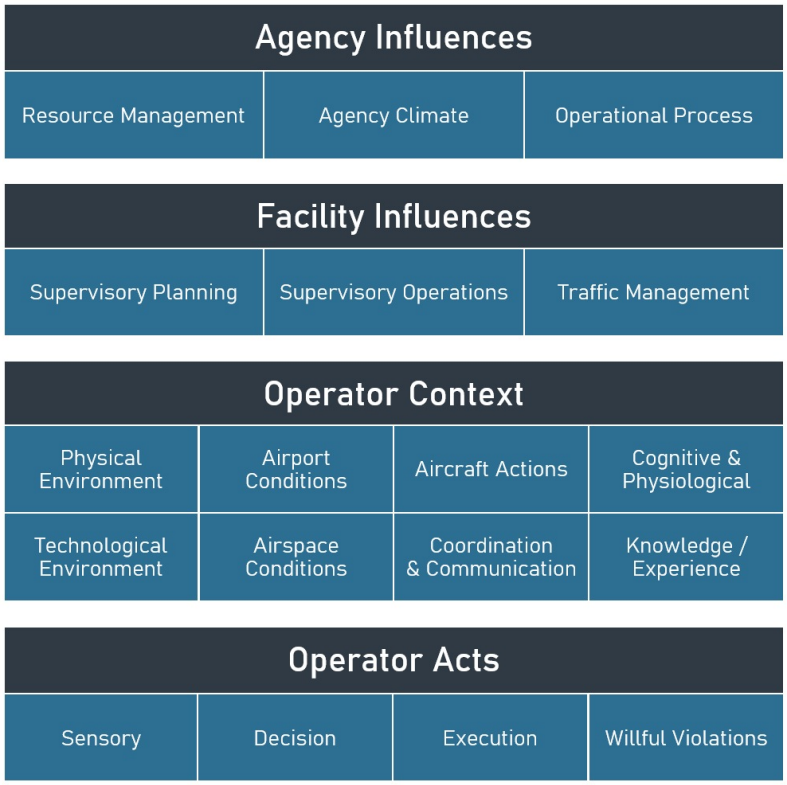

Figure 1: The AirTracs framework.

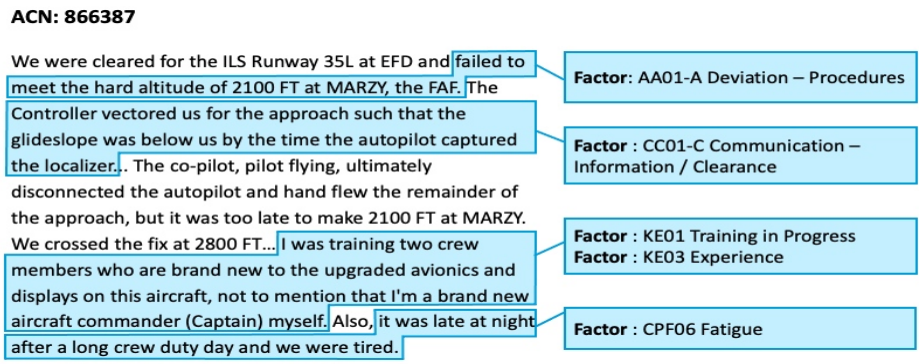

Figure 2: Example ASRS report with labeled AirTracs factors and rationales.

the operator. Operating Context factors detail the environment, interactions, and preconditions linked to the Operator Acts. The third tier, Facility Influences, identifies the factors related to the actions or inactions of individuals at a facility that can impact the entire facility or multiple individuals at that facility. The fourth tier, Agency Influence, examines those factors related to the actions or inactions of the Agency. The AirTracs factors are not mutually exclusive, and safety event classifications should include factors from all four tiers. For more information on AirTracs, refer to Berry, Sawyer, and Austrian (2012).

Training Dataset

A large set of manually labeled aviation safety reports is necessary to train an NLP model. This dataset was prepared by applying the AirTracs taxonomy to a significant volume of ASRS safety reports. The primary description of what happened in each safety event is recorded in a text field referred to as Combined Narratives. The Combined Narratives field consists of the first-person narrative description by the reporter(s), the self-reported roles of those reporters, and any callbacks, which are supplemental textual responses to an ASRS analyst's follow-up questions to the original reporter(s). A report may contain up to two narratives and as many as two callbacks.

The AirTracs taxonomy was applied to a selection of ASRS reports. The reports were analyzed by HF and aviation SMEs with the AirTracs taxonomy to identify a variety of applicable factors, along with a rationale (see Figure 2). The rationale identifies the relevant string of sub-text from the Combined Narratives and serves as a more focused explanation of the reason the factor was applied to the event. For more information on how to apply AirTracs and example results, please refer to Sawyer, Berry, and Austrian (2012). Each ASRS report has many additional fields beyond the narratives and callbacks (e.g., flight conditions, reporter function, etc.), and the application of AirTracs results in additional data fields (e.g., effect type). However, this initial application of NLP was scoped to address the Combined Narratives and the presence/absence of AirTracs factors.

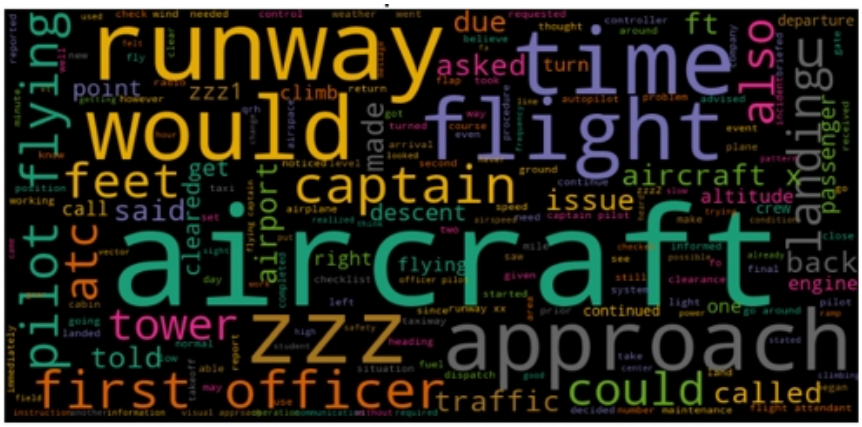

Figure 3: Word cloud output of 17,836 report analyzed with AirTracs.

The ASRS database contains over 100,000 reports filed between February 2002 and August 2022. Our team has analyzed 17,386 of those safety reports, identifying 61,680 total AirTracs factors (an average of 3.55 factors per report). For each identified factor, the team labeled the accompanying rationale within the narrative related to the identified factor. This initial data set of labeled reports represents a significant application and investment of resources.

Before beginning active model development, the data set was reviewed to confirm that it met expectations about language frequency, and to identify any notable or unusual features. The narratives contained 15,454 distinct words. Approximately 40% of the words do not appear in a standard-language dictionary. A brief review of the non-normal words revealed an unsurprising set of aviation-specific acronyms and other languages (e.g., go-around, ADSB), some timely language (e.g., COVID), and various misspellings. Figure 3 shows an example word cloud.

MODEL DEVELOPMENT

Several ML tools were explored to determine the best fit for this application, with the Python AI/ML tool chosen due to its flexibility, availability, and extensive library of support materials. After selecting Python AI/ML for our model development, the first step was to load the 61,680 AirTracs factors previously labeled by SMEs and the supporting ASRS data. This included the full Combined Narrative, the AirTracs factor code (e.g., EX-01B), and the rationale. AirTracs is a tiered taxonomy indicating that a factor at a lower tier (child) is also represented by the associated higher tier (parent). The model incorporated this multi-level classification scheme, with each rationale being labeled for every corresponding level of the hierarchy.

Early iterations of the model used the full narrative text for the analysis. Given that 1) ASRS output provides the full combined narratives, and 2) the AirTracs analysis identified the rationale sub-text, the goal was to have the ML model work directly from the source material. However, given that the

full narratives matched for multiple unrelated factors, it was decided that training the model using the rationale sub-text rather than the complete narrative would be more efficient. Later iterations of the model beyond the scope of this paper will explore the path to extracting and labeling rationales directly from the full narrative text. Table 1 describes these key model inputs.

Various Python AI/ML tools were then used to build, train, and test the model. Ninety percent of the data were used as training material for the model, with 10% retained for future testing purposes. The model was built using the Keras API (Chollet, 2015) for the TensorFlow library (Abadi et al., 2016). Most iterations of the model used components of the NLTK (Natural Language Toolkit) (Bird et al., 2009), as well as the scikit-learn and scikit-multi-learn classification libraries (Buitinck et al., 2013). With the source text selected, there were a number of pre-processing steps necessary to prepare the text for use by the model, e.g., removing punctuation, stop-words, non-English text, and numbers.

Because the language of machine learning algorithms is matrix math on numerical data, the text needs to be converted to a numerical vector, also called vectorization. The model progressed through several means of vectorization, beginning with one-hot encoding, and currently using the Term Frequency-Inverse Document Frequency (TF-IDF) Vectorizer.

The first layer in the model converts the vectorized text into an embedding layer, which positions related words in close proximity to each other in a multi-dimensional vector space. In the current iterations, the model is training its embedding layer as part of the neural network; future iterations are likely to use a pre-trained embedding layer such as GloVe (Global Vectors for Word Representation). This embedding layer is trained on a very large corpus of English language text, allowing the model to take advantage of information about the meaning of words and for synonyms to be seen as related, or sentiment (attitude of the writer) to be given weight. These are all factors that the SME analysts use in their process, and which are likely to be helpful to the model moving forward.

Table 1. Key model inputs.

Data Inputs	Example	Explanation
Tiered, multiple nested taxonomy codes versus unique individual code	**Tiered:** CC01-A, CC01, CC **Unique:** CC01-A	Single labels resulted in less information and very sparse labels. Tiered accounts for the multi-level nature of taxonomy.
Rationale sub-text versus complete Combined Narratives	**Sub-Text:** ~25-word text representing a factor **Full:** ~400-word text representing >3.5 factors on average	The rationale text is more closely describing the factor.

The remainder of model development involved building out the structure of the neural network, selecting layers, and modifying their hyperparameters. The model had a self-trained embedding layer with the training data affecting the selected size of the word vectors, sequence length, and vocabulary size. The current iterations of the model use a Long Short-Term Memory (LSTM) layer that looks for relationships between elements that appear in a sequence. Since the input data in this model are words, phrases or sequences of words can be connected, rather than being treated only as independent words. The model also has contained convolutional layers that work to extract useful information from the input data by making small changes and looking at the outcomes. Finally, a densely connected layer appears at the end of the chain of layers to map the output of all previous layers onto the multi-label classification output.

The resulting model uses a binary cross-entropy loss function to determine which changes in the network make improvements and which do not. This function compares the model's predicted output with the output from the SME analyst and gives a score that allows the model to test the quality of its changes. The model also uses the Adam optimizer to determine which changes to make as it updates the weights in the network (Kingma & Ba, 2014).

Finally, the metric used in the model to determine its accuracy is TopKCategoricalAccuracy. This accuracy measure uses the frequency that the targeted factors appear in the top [K] of the model's predictions. Because some rationales are used with different factors, it is possible that a correct model prediction for a given rationale would match many of those factors. The model currently uses a K of 9, with the intent of balancing the need for multiple correct matches with the possibility of overfitting with too many acceptable (but incorrect) matches.

PRELIMINARY RESULTS AND LESSONS LEARNED

After one training epoch, the current model has a training accuracy in the range of 94%-98%, with the validation accuracy dropping quickly and linearly with each additional epoch. Additional epochs show small improvements to measured accuracy, but the falling accuracy using the validation dataset indicates that the "improved" accuracy is largely due to the overfitting of the model. Running the model against the retained test data results in a real-world measured accuracy in the range of 89%-97%. These numbers vary depending on the random data split for the training and testing data.

Reviewing specific examples of model predictions helped to identify some of the sources of error as well as opportunities for improvements. Frequently, a rationale's predicted factors are correct but for a different labeled factor within the same event, i.e., the same rationale is labeled with two distinct factors. In the data preparation phase, those factors are then expanded to indicate matches for each tier of the taxonomy. However, when the model creates a prediction of factor "A" for the row of text that was marked for factor "B", it was marked as a failure. This is the kind of problem that the

TopK accuracy measure is intended to overcome, and as a result, the accuracy has improved.

NEXT STEPS

There are opportunities for improvements to the model within the current data inputs. In particular, using a pre-trained embedding layer (e.g., GloVe) to allow the model's treatment of the source text material to begin from a base level of English "understanding" is likely to improve results. Additionally, future iterations of the AVIAN-S model will be expanded to incorporate other portions of the ASRS data, such as the reporter function, and of the AirTracs data, such as the factor effect.

The current AVIAN-S model was trained using the rationale sub-texts that were extracted by the human labelers from the full combined narratives. Using the model to extract the rationale statements, in addition to labeling the associated factors, will be an important next step in the process of using the model to perform the initial task of labeling AirTracs factors from the combined narrative source material. There is also an opportunity to explore the validation and application of the AVIAN-S model to additional ASRS reports and even other safety event databases.

REFERENCES

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jozefowicz, R., Jia, Y., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Schuster, M., Monga, R., Moore, S., Murray, D., Olah, C., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., & Zheng, X. (2016). TensorFlow: Large-scale machine learning on heterogeneous systems. OSDI'16: Proceedings of the 12th USENIX conference on operating systems design and implementation. USENIX Association. https://www.usenix.org/conference/osdi16

Berry, K., Sawyer, M., & Austrian, E. (2012). AirTracs: the development and application of an air traffic safety taxonomy for trends analysis. In Proceedings of the 1st annual conference on interdisciplinary science for air traffic management, Daytona Beach, Florida.

Bird, S., Klein, E., & Loper, E. (2009). Natural language processing with Python: Analyzing text with the natural language toolkit. O'Reilly Media, Inc.

Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., Niculae, V., Prettenhofer, P., Gramfort, A., Grobler, J., Layton, R., Vanderplas, J., Joly, A., Holt, B., & Varoquaux, G. (2013). API design for machine learning software: Experiences from the scikit-learn project. ArXiv. https://doi.org/10.48550/arXiv.1309.0238

Chollet, F. (2015). Introduction to Keras for engineers. Keras. https://keras.io/getting_started/intro_to_keras_for_engineers/

ICAO. (1999). Report of Accident Investigation and Prevention (AIG) Divisional Meeting. https://www.document-center.com/standards/show/ICAO-9753

Jonk, P., de Vries, V., Wever, R., Sidiropoulos, G., & Kanoulas, E. (2023). Natural Language Processing of Aviation Occurrence Reports for Safety Management. *Proceedings of the 32nd European Safety and Reliability Conference*.

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

Kierszbaum, S., Klein, T., & Lapasset, L. (2022). ASRS-CMFS vs. RoBERTa: Comparing Two Pre-Trained Language Models to Predict Anomalies in Aviation Occurrence Reports with a Low Volume of In-Domain Data Available. *Aerospace*, 9(10), 591.

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. ArXiv. https://doi.org/10.48550/arXiv.1412.6980

Miyamoto, A., Bendarkar, M. V., & Mavris, D. N. (2022). Natural Language Processing of Aviation Safety Reports to Identify Inefficient Operational Patterns. *Aerospace*, 9(8), 450.

NASA. (2023). Aviation Safety Reporting System. https://asrs.arc.nasa.gov/

Oster Jr, C. V., Strong, J. S., & Zorn, C. K. (2013). Analyzing aviation safety: Problems, challenges, opportunities. Research in Transportation Economics, 43(1), 148–164. https://faculty.wcas.northwestern.edu/ipsavage/104-13.pdf

Sawyer, M., Berry, K., & Austrian, E. (2012). The use of odds ratios and relative risk to quantify systemic risk pathways in air traffic control. Proceedings of the International Society of Air Safety Investigators. https://www.isasi.org/Documents/library/technical-papers/2012/14-The-Use-of-Odds-Ratios-to-Quantify-the-Relationship-be.pdf