EXTRACTING LESSONS OF RESILIENCE USING MACHINE MINING OF THE ASRS DATABASE

Immanuel Barshi
Human Systems Integration Division
NASA Ames Research Center, California
Bryan Matthews
KBR Inc.
NASA Ames Research Center, California
Jolene Feldman
Human Systems Integration Division
NASA Ames Research Center, California

NASA's Aviation Safety Reporting System (ASRS) database is the world's largest repository of voluntary, confidential safety information provided by aviation's frontline personnel. The database contains close to 2 million narratives, many of which describe everyday situations in which people saved the day. In these situations, people's resilient behavior solved a problem, dealt with a malfunction, and maintained a safe operation despite a serious perturbation. To be able to extract lessons of such resilience from this large database, the use of machine learning algorithms is being explored. In this report, we describe a comparison between two such algorithms: BERT and Word2Vec. An identical search using both programs was done on a database containing approximately 270,000 ASRS reports. The comparison reveals some of the strength and weaknesses of each algorithm as well as the challenges inherent in using such algorithms to extract lessons of resilience from the ASRS database.

Aviation safety is often examined in terms of errors leading to incidents and accidents. There is much to be learned from such events, but these events represent an extremely small portion of flight operations. In the vast majority of commercial aviation operations, all goes well in spite of various perturbations. Moreover, in the vast majority of inflight malfunctions of any sort, the crew is able to solve the problem and complete the flight safely. Such resilience as demonstrated in everyday operations can also be a source of much learning.

Learning how to be resilient from what goes well has not been part of common flight training programs. As a result, there are no established methodologies to collect relevant data and to extract relevant lessons. Yet, the aviation industry collects vast amounts of data, especially with the advent of Safety Management Systems (SMS). Thus, it behooves us to make the most of existing sources of data for this purpose of learning resilience from what goes well. One such existing source of data is NASA's Aviation Safety Reporting System (ASRS).

The ASRS database is the world's largest repository of voluntary, confidential safety information provided by aviation's frontline personnel, including pilots, air traffic controllers, mechanics, flight attendants, dispatchers, and other members of the aviation community and the public. The database contains close to 2 million narratives, many of which describe everyday situations in which people saved the day. In these situations, people's resilient behavior solved a problem, dealt with a malfunction, and maintained a safe operation despite a serious perturbation. Hence, these narratives provide a rich source of potential lessons about being resilient in the face of adversity. But the database is very large and extracting such lessons can be challenging.

The online ASRS database has search tools built in. Airline-based Aviation Safety Action Partnership (ASAP) programs which are modeled after the ASRS also have such tools. These databases can be searched in a variety of ways, but depending on the size of the database and the search terms used, searches may yield a voluminous number of reports, well beyond the ability of a human analyst. Feldman et al. (2021) note that "key word searches can be informative, but can fail to detect resilient behaviors that

are not specifically named." Moreover, what would count as 'resilient behavior' is context-dependent; the very same action can be resilient in one situation and catastrophic in another. For instance, an old aviation adage says that the first thing to do in an emergency is to "wind the clock" (it goes back to the times when a mechanical clock requiring winding was installed in the cockpit) to allow the pilot a moment to pull back from the situation and think slowly to properly identify the problem. In many in-flight emergencies, taking a moment to think rather than react immediately can be life-saving. However, some situations such as a rejected takeoff in case of a power failure prior to V1 do require an immediate response and winding the clock at that moment can be life-ending.

Furthermore, resilience is often implicit in narrative reports and cannot be easily identified by keywords or key phrases. To go beyond keywords or phrases, the ASRS search tools allow the use of codes (e.g., ASRS coding taxonomy), and various filters. Thus, it is possible to limit the search to particular situations or events of interest (e.g., Chandra et al., 2020). Beyond such searches, advanced software tools are needed (Paradis et al., 2021).

The first such software tool, specifically designed to support searches of the ASRS database, was Perilog (McGreevy, 2005). Developed by Michael McGreevy at the NASA Ames Research Center, home of ASRS, Perilog is still one of the best text mining tools for studying the ASRS narratives. Functions such as "key word search," "phrase search," and "search by example" make Perilog an excellent search tool, whereas functions such as "review vocabulary," "review phrases," and "phrase generation" make Perilog an exciting discovery tool. One of the unique features of Perilog is the "search by example" function, in which an ASRS report, or any text of any size, can be used as the "search term."

Analysts and researchers may want to search the ASRS database to find information about a particular type of event (e.g., automation surprises or unstable approaches), a particular phase of flight (e.g., descent or approach), or a particular type of operations (e.g., general aviation of scheduled airlines). Operators may have different needs. An airline's safety officer may come across an ASAP report and want to know if there are similar reports in the database. The similarity might be in terms of the particular event at hand, a particular piece of equipment, or something about the circumstances leading to the reported event. Likewise, identifying a particular resilient strategy in a report can serve as the basis for finding the use of that strategy under different circumstances, or for finding different strategies that can be used under similar circumstances. Thus, being able to search the database using a report as an example can be very useful.

Below, we describe a comparison between two new algorithms: Word2Vec and Bidirectional Encoder Representations from Transformers (BERT). An identical search-by-example using both programs was done on a database containing approximately 270,000 ASRS reports submitted between 1988 and 2022. The comparison reveals some of the strength and weaknesses of each algorithm as well as the challenges inherent in using such algorithms to extract lessons of resilience from the ASRS database.

Method

Software Mining Tools

The Word2Vec algorithm (Mikolov et al., 2013) is a natural language processing (NLP) algorithm used to model term similarity between two words in a multi-dimensional embedding space. The algorithm accomplishes this by training a neural network to learn word associations. The algorithm is unsupervised, meaning that no labels are provided by a subject matter expert to train the model. There are two approaches to learning the word associations: 1) using a continuous bag of words (CBOW), and 2) skip-gram. CBOW uses the surrounding words to predict the probability of a target word in the middle of a window. Windows are typically 3 or 5 words in length. The skip-gram method is the inverse task, namely, learning to predict the probability of surrounding words from the target word. Neither method considers word ordering other than the target word being in the center of the window. Both methods use the same neural network architecture with a fully connected neural network layer of input size equal to the entire term corpus mapped to a 300-dimension hidden layer. The final layer maps the hidden layer to

the predicted output word space with a softmax activation function (Bridle, 1990) to convert the output to a classifier. A classifier model is used to learn word similarity becasue if the classifier can accurately predict the target word(s) with the contextual word(s), then the embedding space is presumed to be well organized by term similarity. Common NLP techniques such as removing stopwords and stemming are applied to all the ASRS reports before training. Stopwords are typically pronouns, articles, prepositions, and other words that do not add significant value to the text's meaning but often indicate grammatical relations. Stemming involves the use of stem-words for the different forms words can take such as using "friend" for friends, friendly, and friendship. For our work, we have been using Python's NLTK package's 'english' stopword list (Loper & Bird, 2002). Once the model is learned, the hidden layer can be leveraged to extract word associations. Each word is mapped into the embedding space and word similarities can be computed using the cosine similarity function between any two word vectors. Term Frequency-Inverse Document Frequency (TF-IDF) weighting, a commonly applied technique, is also applied to the embedding vector for each of the words. This weighting approach attempts to de-emphasize words that appear across a majority of the documents (and therefore their presence is less informative than infrequent terms) while boosting terms that occur frequently within a document. For example, if a term appears multiple times in a report and is very uncommon across the rest of the reports then its weighting is high. The inverse is true for common words that appear in both the report and the rest of the dataset. An entire report embedding can be represented by computing the average word embedding across the report with TD-IDF weighing. This vector representation allows comparisons among reports.

Bidirectional Encoder Represenations from Transformers (BERT) algorithm (Devlin et al., 2018) can also perform this task. Using the same concept as Word2Vec, the BERT algorithm maps a report into an embedding space. Similar to Word2Vec, the algorithm's architecture is based on a neural network; however, the BERT network is much deeper than Word2Vec with 12 fully connected multi-headed self-attention layers (Vaswani et al., 2017) with a hidden layer of 768 dimensions. The self-attention layers capture the bi-directional context of a word, using the words prior to as well as following a target word to predict the output sequence of words. Another difference from Word2Vec is that the embedding dimensions are applied to an entire sentence and not at the word level. The entire report embedding vector is obtained by calculating the average sentence embedding. Cosine similarity is also used as the similarity metric for ranking reports against the query. The pretrained Microsoft *mpnet* (Song et al., 2020) model with fine tuning on an additional 1.17B data tuples was used to perform the sentence embedding. This open-sourced model is publicly available (Espejel, 2021).

Runway Safety Narrative

Runway safety has been a high-priority safety concern in flight operations at an international level (ICAO, 2017). The Flight Safety Foundation launched a Global Action Plan for the Prevention of Runway Excursions (GAPPRE; FSF, 2021), and is currently engaged in launching a similar Global Action Plan for the Prevention of Runway Incursions (GAPPRI; FSF, personal communication). Given these efforts, our search of the ASRS database focused on issues related to runway incursions.

Rather than search the database for a report of a runway incursion, a narrative was drafted to be used in a search-by-example. Writing up such a narrative can be a very productive approach to mining the database. An airline's Safety Officer, or a safety researcher can imagine a situation of interest and write up a narrative as if experiencing the situation and writing an ASRS or an airline-internal ASAP report about it. Writing up such a narrative presents an opportunity to fashion the report along the specific aspects of interest. Moreover, different reporters often use different words and phrases to describe similar and even identical situations. Writing up a narrative allows the researcher to use multiple phrases and styles within a single report to increase the likelihood of finding relevant reports in the database.

Search-By-Example Process

For both Word2Vec and BERT any text/report can be used to query for similar reports. The Word2Vec process involves stemming and dropping stop words. Then each remaining word in the report is mapped to the model's 300-dimension embedding vector space and the TF-IDF weight for that word is applied to that vector. This process is repeated for all words in the report and the average embedding vector is calculated. Prior to the query, this process was applied across all reports in the sample ASRS database to calculate each report's average embedding vector. The cosine similarity function was used to compute the angle of similarity between the query report's average embedding vector and each of the ASRS report's average embedding vectors. This approach allows queries to be agnostic of report length and therefore a query can be a single term or an extensive report of any length. The process is similar for BERT, however stemming and stop word filters are not applied and instead of computing the average embedding vector across words, the average embedding vector is computed across sentences. The embedding vector for the large BERT model is 768-dimensions and the cosine similarity function is used to find the closest matches. With both algorithms, the top 10 most similar reports were analyzed.

Two search runs were employed. In the first run, the sample writeup was used to search for similar reports in the sample ASRS dataset. In the second run, the top most similar report from each search was used in a second round of searches. Thus, 38 reports in all were analyzed for similarity with the initial sample writeup.

Results

The top 10 most similar ASRS reports to the written-up runway incursion narrative produced by the Word2Vec algorithm were all related in some way to runway safety issues. Not all reports deemed similar were of the same type of operation; the written-up narrative described an airline operation and some of the reports found involved a general aviation operation. Furthermore, not all reports involved a runway incursion; some involved landing at the wrong airport, a takeoff without a clearance, or being stuck on a taxiway. However, they all did have sufficient similarity to be of potential interest.

Only 2 of the top 10 most similar ASRS reports to the written-up runway incursion narrative produced by the BERT algorithm were related in some way to runway safety issues. Most of the reports involved an in-flight anomaly. What's more, none of the reports produced by BERT were also in the top 10 most similar reports produced by Word2Vec.

Because the top most similar report produced in the first search was used to drive the second search, and because this report was an ASRS report, as expected, both algorithms returned the same report as the most similar to the example used for the search. Of the additional 9 reports returned by Word2Vec, only 2 were also among the 10 reports produced in the first search. Most reports involved surface operations though not necessarily a runway incursion. Similarly, only 2 of the 9 most similar reports returned by the second search in BERT were among the reports produced in the first search. Most of the reports returned by BERT involved in-flight anomalies.

The most striking similarity across all 38 reports was their length. The average length of the narratives in the sample ASRS database used in this study was 230 words (with a median of 191 words). The written-up runway incursion narrative used as the example in the first search had 763 words. The average number of words in the 10 most similar reports returned by Word2Vec was 668 words, and 813 words as the average for the narratives returned by BERT. The second search with Word2Vec returned an average length of 946 words, whereas BERT returned an average narrative length of 847 words. All these reports are significantly longer than the vast majority of reports in the database, and certainly longer than the average report length in the database (see Fig. 1 below).

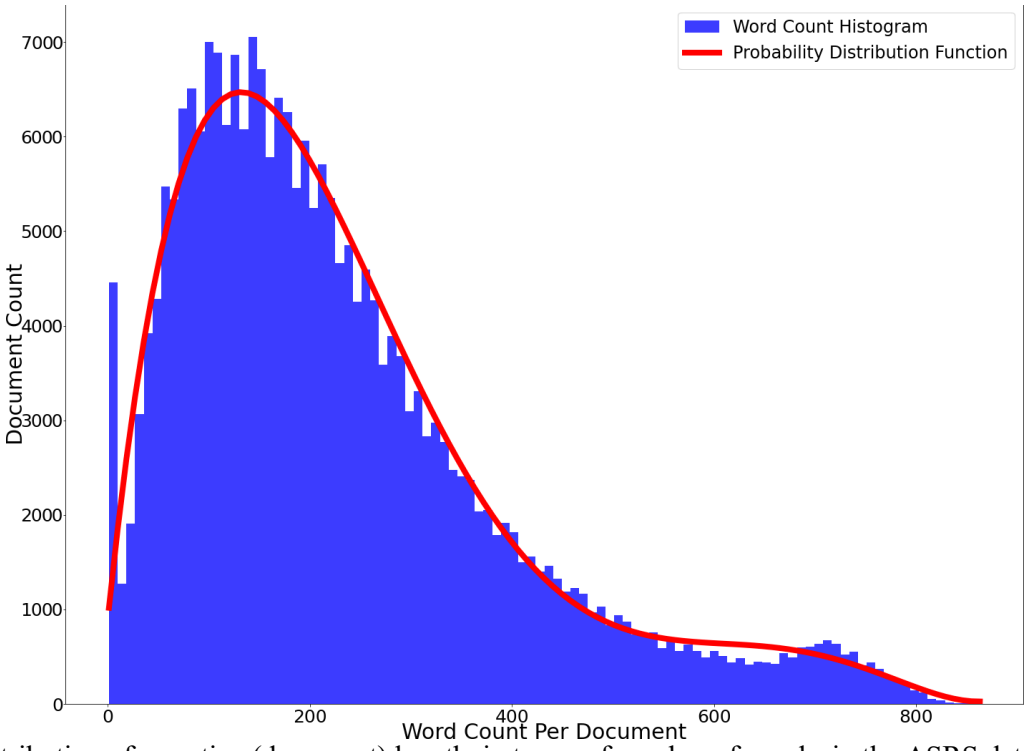

Figure 1. Distribution of narrative (document) length, in terms of number of words, in the ASRS database used in this study.

Discussion

The longer the narrative, the richer it is. Rich ASRS reports typically include much detail about the circumstances involved, including various pressures the crew might have been under such as fatigue or schedule constraints. Thus, the richer the report, the more elements in it could be used by an algorithm to determine similarity independent of the specific event or malfunction at the core of the event. These similarities are of potential interest as they could involve similar resilient strategies. However, that determination is left to the human analyst. Similarly, because of the high context-dependency of determining any behavior as "resilient," as discussed above, current search algorithms may not be sensitive enough to support the extraction of lessons of resilience from a narrative database such as ASRS. These algorithms can help narrow the search to some extent and thus allow the human analyst to focus on the most relevant reports to one's interest. But that relevancy too must be examined as a report could be deemed "similar" based on parameters outside the analyst's interest.

The difference in relevancy to the initial runway incursion narrative writeup between the two algorithms might be explained in part by the different texts used in their initial training. An algorithm trained on a database of newspaper articles or scientific articles might return very different results from those returned by an algorithm trained on social media posts. Thus, when choosing to use an algorithm in the analysis of narrative texts, one must be mindful of the database used in the training of the algorithm, and ensure that the vocabulary, grammatical structures, and language style are appropriate to the texts to be analyzed.

Acknowledgements

The work reported here was funded by NASA's Human Contribution to Safety part of the System-Wide Safety Project, of the Aeronautics Research Mission Directorate's Aviation Operations and Safety Program.

References

Aviation Safety Reporting System: ASRS Database Online. (2023). https://asrs.arc.nasa.gov/search/database.html. Accessed April 16, 2023.

Bridle, J. S. (1990). Probabilistic Interpretation of Feedforward Classification Network Outputs, with Relationships to Statistical Pattern Recognition. In: Soulié, F.F., Hérault, J. (eds) *Neurocomputing.* NATO ASI Series, vol 68. Springer, Berlin, Heidelberg. Retrieved from https://doi.org/10.1007/978-3-642-76153-9_28

Chandra, D., Sparko, A., Kendra, A., & Kochan, J. (2020). *Operational complexity in performance-based Navigation (PBN) arrival and approach instrument flight procedures (IFPs).* Retrieved from https://rosap.ntl.bts.gov/view/dot/43835

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.* Retrieved from https://arxiv.org/abs/1810.04805.

Espejel, O. (2022). *Train and fine-tune sentence transformers models* [Data model]. Retrieved from https://huggingface.co/sentence-transformers/all-mpnet-base-v2/blob/main/README.md

Feldman, J., Barshi, I., Smith, B., & Matthews, B. (2021). Reports of resilient performance: Investigating operators' descriptions of safety-producing behaviors in the ASRS. In *Proceedings of the 2021 International Symposium on Aviation Psychology* (pp. 122-127).

Flight Safety Foundation. (2021). *Global action plan for the prevention of runway excursions*. Retrieved from https://flightsafety.org/wp-content/uploads/2021/05/GAPPRE-Parts-1-2-2021-FINAL.pdf

International Civil Aviation Organization. (2017). *Runway safety program – Global runway safety action plan.* Retrieved from https://www.icao.int/safety/RunwaySafety/Documents%20and%20Toolkits/GRSAP_Final_Edition01_2017-11-27.pdf

Loper, E., & Bird, S. (2002). *Nltk: The natural language toolkit*. Retrieved from *https://arxiv.org/abs/cs/0205028*

McGreevy, M. W. (2005). *Perilog text mining methods and software*. Moffett Field, CA: NASA Ames Research Center.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). *Efficient Estimation of Word Representations in Vector Space*. Retrieved from https://doi.org/10.48550/ARXIV.1301.3781

Paradis, C., Kazman, R., Davies, M. D., & Hooey, B. L. (2021). *Augmenting topic finding in the NASA Aviation Safety Reporting System using topic modeling*. AIAA SciTech Forum. Retrieved from https://arc.aiaa.org/doi/abs/10.2514/6.2021-1981

Song, K., Tan, X., Qin, T., Lu, J., & Liu, T.-Y. (2020). *MPNet: Masked and Permuted Pre-training for Language Understanding.* Retrieved from https://arxiv.org/abs/2004.09297.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). *Attention Is All You Need.* Retrieved from https://arxiv.org/abs/1706.03762.