Real-Time Risk Evaluation System for Aviation Safety
Saakaar Bhatnagar saakaar@stanford.edu, Nicolas Tragus ntragus@stanford.edu
https://youtu.be/o4oJxVlkbYE

What’s new
We aim at improving aviation safety by creating a tool capable of assessing the risk of a developing situation in flight based on previous reports from pilots in such situations. The novelty of this is that the tool will work in real-time, with the pilot able to communicate with the algorithm.

Database
We take the data from the ASRS database: 210,000 accident reports on every type of planes in the US.

Input data used from the report (given as string):
- Situation
- Crew Size (integer)
- Narrative

We need to map every outcome of situation (e.g. aircraft damaged, general maintenance needed) to a one-hot encoded vector of size 5, to depict the risk categories in order to get a classification problem.

We notice we have a class imbalance issue. This leads the network to overpredict class 2 when testing

<table>
<thead>
<tr>
<th>Class imbalance</th>
<th>17,621</th>
</tr>
</thead>
<tbody>
<tr>
<td>1,709</td>
<td>Removed</td>
</tr>
</tbody>
</table>

Data preprocessing
We are dealing with the categorical strings and the narrative separately.
- Remove Nan/Junk in the data
- Tokenize the data passed as categorical string
- Pad the data with zeros and pass it on to the network

Models

<table>
<thead>
<tr>
<th>Input Layer</th>
<th>Embedding Layer</th>
<th>LSTM Layer (4 units)</th>
<th>Realization Layer</th>
<th>Dense Layer and Binarization</th>
<th>Output Layer (2 Classes Activation)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weather</td>
<td>Crew Size</td>
<td>Flight Phases</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Flight Type</td>
<td>Flight Phases</td>
<td>Flight Phases</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Situation</td>
<td>Flight Phases</td>
<td>Flight Phases</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Event</td>
<td>Risk level</td>
<td>Risk level</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

We tested three types of architecture:
- Sentiment score analysis of the narrative, based on 0.9*optimism calculated + 0.1*subjectivity of narrative. Outputs a float for the narrative that can be used by the network.
- Learn a word embedding for every input and pass it to the LSTM layer further down the network.
- Use a pre-trained word embedding (GloVE) for the narrative. The other categorical text inputs still have a trainable embedding.

We use an embedding layer followed by a couple of LSTM layers and dense layers, and the output is given by a softmax activation.

Future work
A few ways to improve our algorithm:
- Add technical data to the input, such as speed of ascent/descent, engine revolution per minute, etc...
- The narrative used for training are in the past tense, but the pilots in real-time would use present, thus improving the language processing part could help improve the performances

Results/Discussion
After class balancing, our confusion matrix has improved but isn’t as diagonal as we would like.
As this is a safety application, we chose recall as our primary metric to optimize. The third algorithm performs better in that regard, especially for situations with a high-risk level.

The recall numbers obtained might seem relatively low but are due to two factors:
- High complexity of the problem: the same set of circumstances can lead to different outcomes
- Relatively high Bayes/Human error for the problem

References