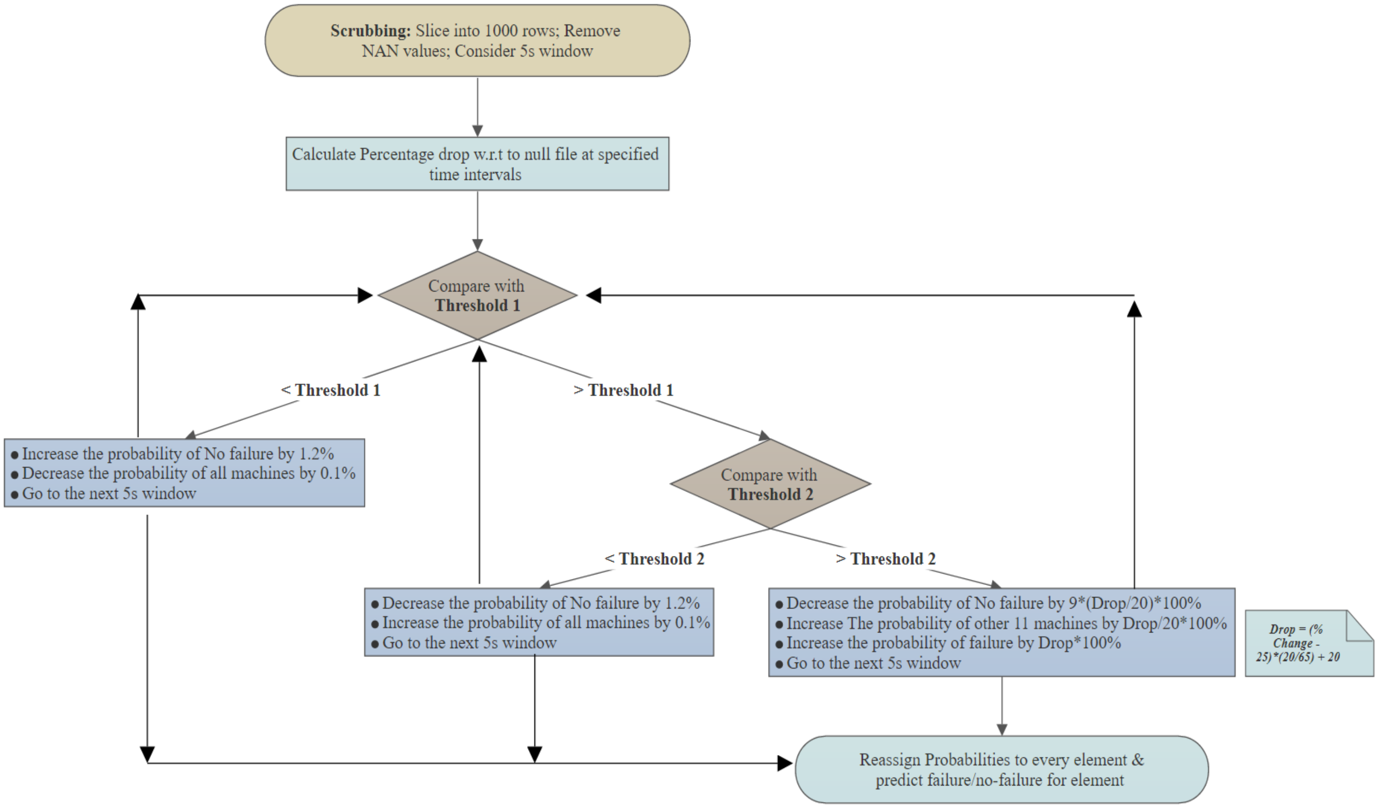
CISCO NETWORK TESTBED PROJECT

(Team 6)

Yi-Chen Lee
Prabhakar Nanduri
Leidan Ruan
Shakti Vachhani

Model Flowchart


Scrubbing: Slice into 1000 rows; Remove NAN values; Consider 5s window
Calculate Percentage drop w.r.t to null file at specified time intervals
Compare with Threshold 1
< Threshold 1
> Threshold 1
Increase the probability of No failure by 1.2%
Decrease the probability of all machines by 0.1%
Go to the next 5s window
Compare with Threshold 2
< Threshold 2
> Threshold 2
Decrease the probability of No failure by 1.2%
Increase the probability of all machines by 0.1%
Go to the next 5s window
Decrease the probability of No failure by 9*(Drop/20)*100%
Increase The probability of other 11 machines by Drop/20*100%
Increase the probability of failure by Drop*100%
Go to the next 5s window
Drop = (% Change - 25)*(20/65) + 20
Reassign Probabilities to every element & predict failure/no-failure for element

Model Development:

The approach here is to compare the percentage change in mean values of input/output data for failure files in reference to no-failure file at specific time intervals. The first step is to scrub the data by removal of time stamps with no data provided. Slice the total data in 1000 rows with no more than 5-second time interval. Later, determine Thresholds for failure and no failure. Then, assign probabilities accordingly. The detailed steps are outlined here.

Deciding Thresholds:

1) Calculate global mean of I/O packet rates for individual machines in File 2.
2) Calculate local means of I/O packet rates for individual machines in File 1, File 2 and File 3 at approx. 5seconds time interval.
3) Determine percentage change in local means w.r.t global means of File 2 for individual machines in File1, File2 and File3 at different intervals.
4) Select the maximum percentage change among those intervals for each machine.
5) The max. value of percentage change for machines in File 2 serves as Threshold 1.
   Note: Threshold 1 is lower limit which accounts for no failure. Fluctuations below Threshold 1 can be neglected and is considered a normal function of the network.

Leaf 1	Leaf 2	Leaf 3	Leaf 4	Leaf 5	Leaf 6	Leaf 7	Leaf 8	Spine 1	Spine 2	Spine 3	Spine 4
1.67	1.51	1.65	1.58	1.67	2.33	1.96	1.74	1.76	2.02	5.27	1.98

6) The max. value of percentage change among File 1 and File 3 for each machine serves as Threshold 2.
   Note: Threshold 2 is upper limit which accounts for failure. Fluctuations above this limit is considered a failure. But, here for simplicity we have considered a uniform value of 25% as Threshold 2.

After deciding thresholds for failure and no failure, the training data is run using Python code which determines percentage changes in test files at specified time intervals w.r.t to mean values of null File2. The program then assigns probabilities of failure and no failure based on thresholds.

Deciding Probability Distribution:

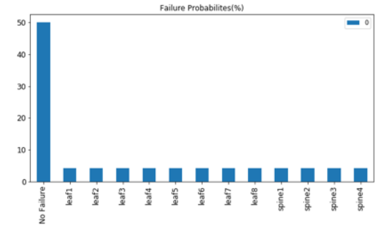
1) If the change is lower than Threshold 1, it is ignored and is a no failure event. The probabilities will be reassigned by increasing the no failure probability to 1.2% and decreasing the probabilities of 12 machines by 0.1%.

2) If the change is between Threshold 1 and Threshold 2, the fluctuations indicate the signs of potential failure. Sine, we are unsure about a failure in a fixed machine at his stage, the probabilities of all machines will be increased by 0.1% and that of a no failure event will be decreased by 1.2%.
3) If the change is larger than Threshold 2, it is considered to be a potential failure. The goal here is to reassign probabilities to make a failure certain from a no failure event. To make a sensitive model, the probabilities are increased in direct proportion to percentage change. Higher the drop, higher the addition of probability. Considering, 20% increase for 25% drop and 40% increase for a 90% drop; we get Drop = (% change – 25%)*(20/65) + 20%. To reassign probabilities, decrease no failure event by 9*(Drop/20)*100%; total decrease in other machines will be 11*(Drop/20)*100% which in all increases the failure probability by Drop*100%.

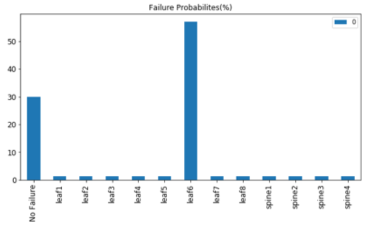
Model Optimization:

The challenging part here was distribution of probabilities after execution of every 5s window. Due to continuous drops below threshold limit some elements showed negative probabilities at certain time intervals. The model solved this issue by adding probabilities to negative elements till the outcome is 1%. This additional probability is taken proportionally from every positive outcome. In this way, the sum of probability will always be 1.

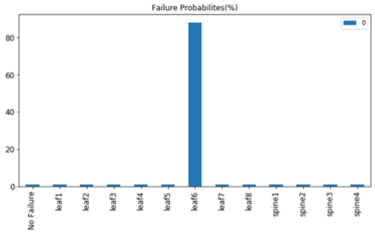
Performance on Training Set



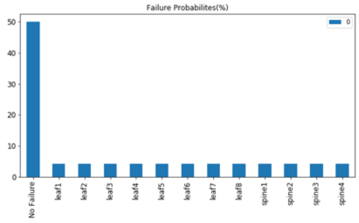
File 1 start



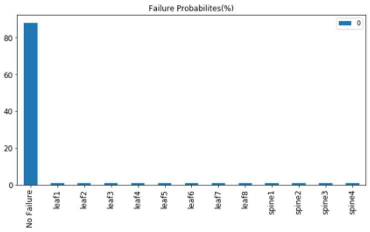
File 1 after 10000 rows



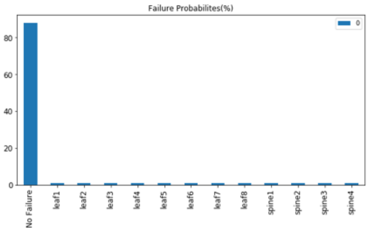
File 1 reaching conclusion



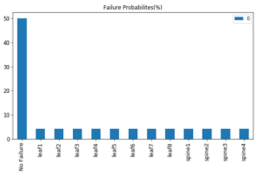
File 2 start



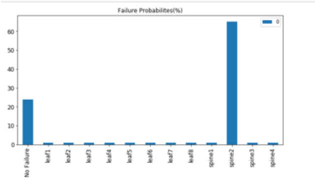
File 2 after 10000 rows



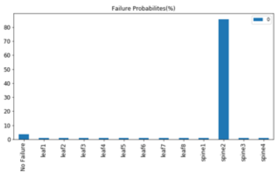
File 2 reaching conclusion



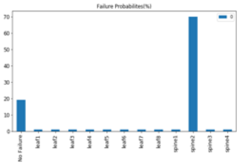
File 3 start



File 3 after 10000 rows



File 3 reaching conclusion



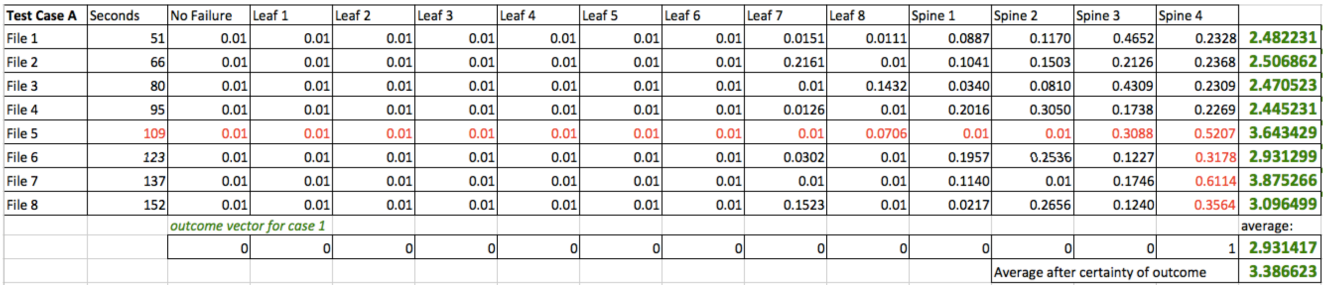
File 3 showing recovery

Analysis:

After running our model, the model found Leaf 6 was failed in File1; there's no failure in File2; In File3, Spine2 was failed. After File3 reaching conclusion, the model also captured the recovery of Spine2 from failure. The average information gain for these 3 files on the test above is: 4.11(File1), 0.82(File2), 4.02(File3)

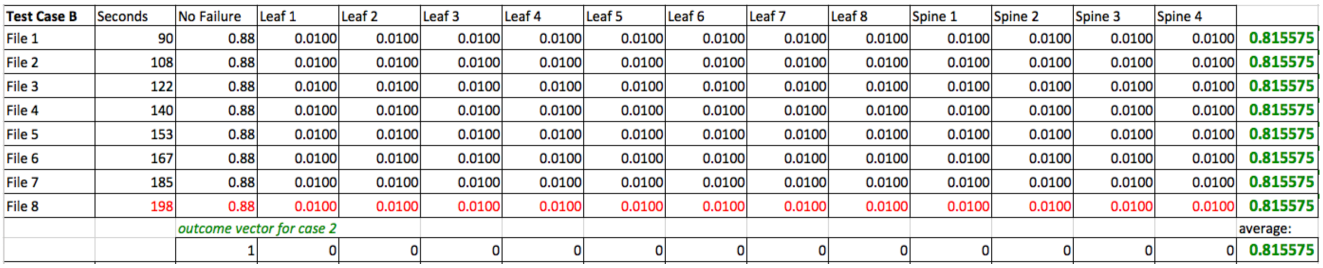
Performance on Test Set

Case A: The model when run on the 8 file-parts of Case A detects failure for spine4 with certainty, after about 109 seconds from the start time. When the information gain is compared to the file3(spine failure) of training set, the average information gain on test set is 3.383 bits (considering after the certainty is made) and on the training set is 4.016 bits. The difference of 15% is within the ranges to discard the possibilities of over-fitting of the model to the test set data.



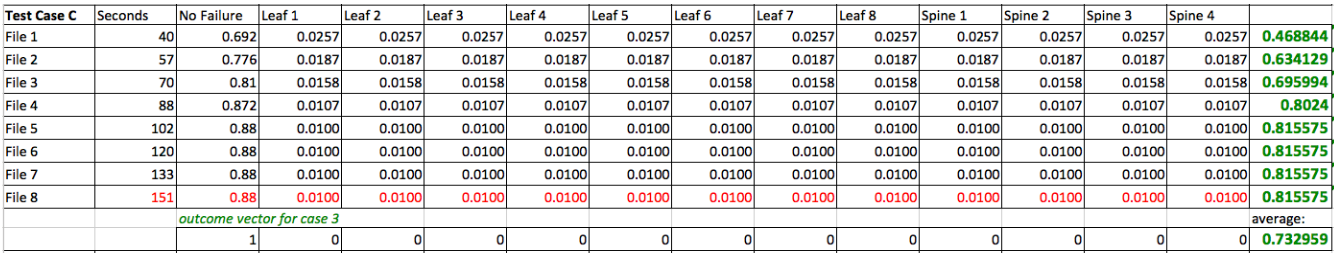
Test Case A	Seconds	No Failure	Leaf 1	Leaf 2	Leaf 3	Leaf 4	Leaf 5	Leaf 6	Leaf 7	Leaf 8	Spine 1	Spine 2	Spine 3	Spine 4	
File 1	51	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.0151	0.0111	0.0887	0.1170	0.4652	0.2328	**2.482231**
File 2	66	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.2161	0.01	0.1041	0.1503	0.2126	0.2368	**2.506862**
File 3	80	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.1432	0.0340	0.0810	0.4309	0.2309	**2.470523**
File 4	95	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.0126	0.01	0.2016	0.3050	0.1738	0.2269	**2.445231**
File 5	109	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.0706	0.01	0.01	0.3088	0.5207	**3.643429**
File 6	*123*	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.0302	0.01	0.1957	0.2536	0.1227	0.3178	**2.931299**
File 7	137	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.1140	0.01	0.1746	0.6114	**3.875266**
File 8	152	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.1523	0.01	0.0217	0.2656	0.1240	0.3564	**3.096499**
		outcome vector for case 1													average:
		0	0	0	0	0	0	0	0	0	0	0	0	1	**2.931417**
												Average after certainty of outcome			**3.386623**

Case B: The model when run on the 8 file-parts of Case B ascertains that there would be no failure in the near future within the first 90 seconds itself. Since the failure event can be triggered at any near future time, the certainty of the no-failure event is done only at the end of the test set file. Information gain comparison between the training and test file set shows that there is no over-fitting to the training set data. [The average is 0.8155 bits for both training and test files].



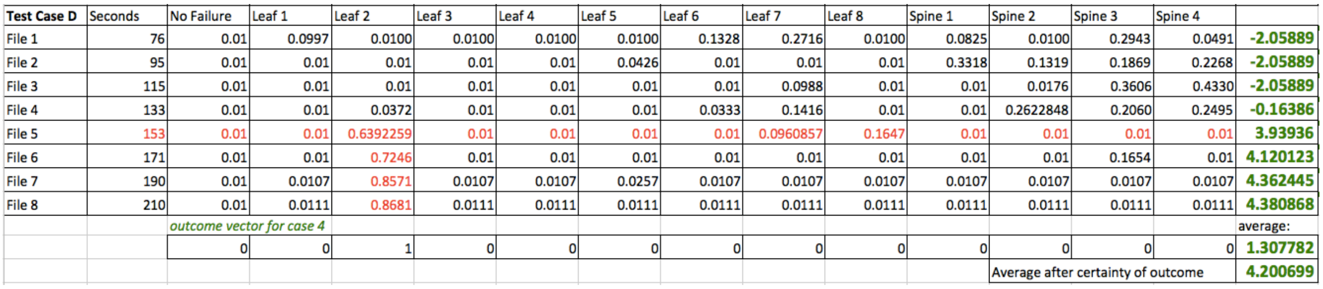
Test Case B	Seconds	No Failure	Leaf 1	Leaf 2	Leaf 3	Leaf 4	Leaf 5	Leaf 6	Leaf 7	Leaf 8	Spine 1	Spine 2	Spine 3	Spine 4	
File 1	90	0.88	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	**0.815575**
File 2	108	0.88	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	**0.815575**
File 3	122	0.88	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	**0.815575**
File 4	140	0.88	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	**0.815575**
File 5	153	0.88	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	**0.815575**
File 6	167	0.88	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	**0.815575**
File 7	185	0.88	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	**0.815575**
File 8	198	0.88	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	**0.815575**
		outcome vector for case 2													average:
		1	0	0	0	0	0	0	0	0	0	0	0	0	**0.815575**

Case C: The model when runs on the 8 file-parts of Case C ascertains that there would be no failure in the near future within the first 40 seconds. Since the failure event can be triggered at any near future time, the certainty of the no-failure event is done only at the end of the test set file. Information gain comparison between the training and test file set shows that there is no over-fitting to the training set data. [The average is 0.8155 bits for both training and test files].



Test Case C	Seconds	No Failure	Leaf 1	Leaf 2	Leaf 3	Leaf 4	Leaf 5	Leaf 6	Leaf 7	Leaf 8	Spine 1	Spine 2	Spine 3	Spine 4	
File 1	40	0.692	0.0257	0.0257	0.0257	0.0257	0.0257	0.0257	0.0257	0.0257	0.0257	0.0257	0.0257	0.0257	**0.468844**
File 2	57	0.776	0.0187	0.0187	0.0187	0.0187	0.0187	0.0187	0.0187	0.0187	0.0187	0.0187	0.0187	0.0187	**0.634129**
File 3	70	0.81	0.0158	0.0158	0.0158	0.0158	0.0158	0.0158	0.0158	0.0158	0.0158	0.0158	0.0158	0.0158	**0.695994**
File 4	88	0.872	0.0107	0.0107	0.0107	0.0107	0.0107	0.0107	0.0107	0.0107	0.0107	0.0107	0.0107	0.0107	**0.8024**
File 5	102	0.88	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	**0.815575**
File 6	120	0.88	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	**0.815575**
File 7	133	0.88	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	**0.815575**
File 8	151	0.88	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	0.0100	**0.815575**
		outcome vector for case 3													average:
		1	0	0	0	0	0	0	0	0	0	0	0	0	**0.732959**

Case D: The model when run on the 8 file-parts of Case D detects failure for leaf2 with certainty, after about 153 seconds from the start time. When the information gain is compared to the file1 (leaf failure) of training set, the average information gain on test set is 4.2 bits (considering after the certainty is made) and on the training set is 4.11 bits. The difference of 2% is within the ranges to discard the possibilities of over-fitting of the model to the test set.



Test Case D	Seconds	No Failure	Leaf 1	Leaf 2	Leaf 3	Leaf 4	Leaf 5	Leaf 6	Leaf 7	Leaf 8	Spine 1	Spine 2	Spine 3	Spine 4	
File 1	76	0.01	0.0997	0.0100	0.0100	0.0100	0.0100	0.1328	0.2716	0.0100	0.0825	0.0100	0.2943	0.0491	**-2.05889**
File 2	95	0.01	0.01	0.01	0.01	0.01	0.0426	0.01	0.01	0.01	0.3318	0.1319	0.1869	0.2268	**-2.05889**
File 3	115	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.0988	0.01	0.01	0.0176	0.3606	0.4330	**-2.05889**
File 4	133	0.01	0.01	0.0372	0.01	0.01	0.01	0.0333	0.1416	0.01	0.01	0.2622848	0.2060	0.2495	**-0.16386**
File 5	153	0.01	0.01	0.6392259	0.01	0.01	0.01	0.01	0.0960857	0.1647	0.01	0.01	0.01	0.01	**3.93936**
File 6	171	0.01	0.01	0.7246	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.1654	0.01	**4.120123**
File 7	190	0.01	0.0107	0.8571	0.0107	0.0107	0.0257	0.0107	0.0107	0.0107	0.0107	0.0107	0.0107	0.0107	**4.362445**
File 8	210	0.01	0.0111	0.8681	0.0111	0.0111	0.0111	0.0111	0.0111	0.0111	0.0111	0.0111	0.0111	0.0111	**4.380868**
		outcome vector for case 4													average:
		0	0	1	0	0	0	0	0	0	0	0	0	0	**1.307782**
												Average after certainty of outcome			**4.200699**

Further Work:

The predictive model developed is completely depended on the Threshold1 and Threshold2 for accurate failure/no failure prediction. The thresholds currently have been entered based on only the training files. In future, once prediction for a particular event is made, the historical data can be used to optimize the two thresholds (based on the outcome obtained) for better probabilistic predictions. Simply stated, this would enable the model to LEARN from test data (that can be now interpreted as training data).

Reference (Model Code Repository): https://github.com/nanduriprabhakar/Data_Science_Team_Project