EEG confusion

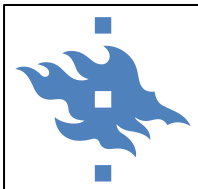
EDM | Fall 2018

Keith Davis | Sami Sarsa

The Goal

1. Predict students' self-labeled confusion for a video, given time series 'EEG' data and a boolean label indicating whether or not a student was confused when watching the video.
   a. Confusion labels are binary for each (SubtitleID, VideoID) pair. Alas, no confusion label per interval.

2. Improve upon the results using the following:
   a. Sentence embeddings (ELMo) for caption data
   b. Video features



The Data

- Available in Kaggle
  - https://www.kaggle.com/wanghaohan/confused-eeg
- 10 subjects watched 20 videos with length of 2 minutes.
  - Half made “difficult” by showing only the middle minute.
    - We have only this half*
  - Data points for each 0.5 second interval of watching the video for every (SubjectID,VideoID) pair
- Confusion is a boolean, assigned per video

*see slide notes

The videos were also classified by the authors as ‘difficult’ or ‘not difficult’ in terms of subject matter. So, while all videos in this dataset were ‘made difficult’ by showing students only the middle minute, they can also be classified into easy and hard based on the subject matter discussed. Slides 4, 10, and 11 also explain this further.

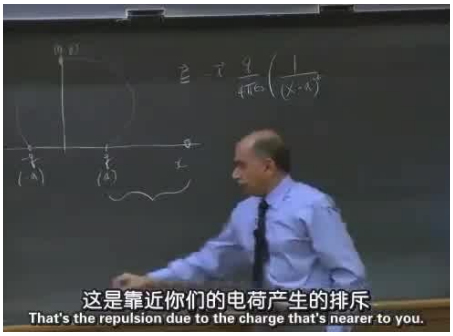
The Data - ‘EEG’

	SubjectID	VideoID	Attention	Mediation	Raw	Delta	Theta	Alpha1	Alpha2	Beta1	Beta2	Gamma1	Gamma2	predefinedlabel	user-definedlabeln
0	0.0	0.0	56.0	43.0	278.0	301963.0	90612.0	33735.0	23991.0	27946.0	45097.0	33228.0	8293.0	0.0	0.0
1	0.0	0.0	40.0	35.0	-50.0	73787.0	28083.0	1439.0	2240.0	2746.0	3687.0	5293.0	2740.0	0.0	0.0
2	0.0	0.0	47.0	48.0	101.0	758353.0	383745.0	201999.0	62107.0	36293.0	130536.0	57243.0	25354.0	0.0	0.0
3	0.0	0.0	47.0	57.0	-5.0	2012240.0	129350.0	61236.0	17084.0	11488.0	62462.0	49960.0	33932.0	0.0	0.0
4	0.0	0.0	44.0	53.0	-8.0	1005145.0	354328.0	37102.0	88881.0	45307.0	99603.0	44790.0	29749.0	0.0	0.0
5	0.0	0.0	44.0	66.0	73.0	1786446.0	176766.0	59352.0	26157.0	15054.0	33669.0	33782.0	31750.0	0.0	0.0
6	0.0	0.0	43.0	69.0	130.0	635191.0	122446.0	90107.0	65072.0	36230.0	53019.0	62938.0	59307.0	0.0	0.0
7	0.0	0.0	40.0	61.0	-2.0	161098.0	12119.0	1963.0	809.0	1277.0	3186.0	3266.0	2518.0	0.0	0.0
8	0.0	0.0	43.0	69.0	17.0	492796.0	120998.0	63697.0	68242.0	10769.0	88403.0	73756.0	22676.0	0.0	0.0
9	0.0	0.0	47.0	69.0	-59.0	82048.0	116131.0	47317.0	26197.0	41642.0	28866.0	32551.0	41810.0	0.0	0.0
10	0.0	0.0	48.0	38.0	-14.0	757165.0	186196.0	3242.0	3841.0	18854.0	43021.0	46799.0	11928.0	0.0	0.0
11	0.0	0.0	44.0	48.0	72.0	667513.0	141854.0	75050.0	16234.0	45926.0	34496.0	74875.0	31839.0	0.0	0.0

Not actually EEG data, although somewhat similar. Output is from a Mindset wearable device.

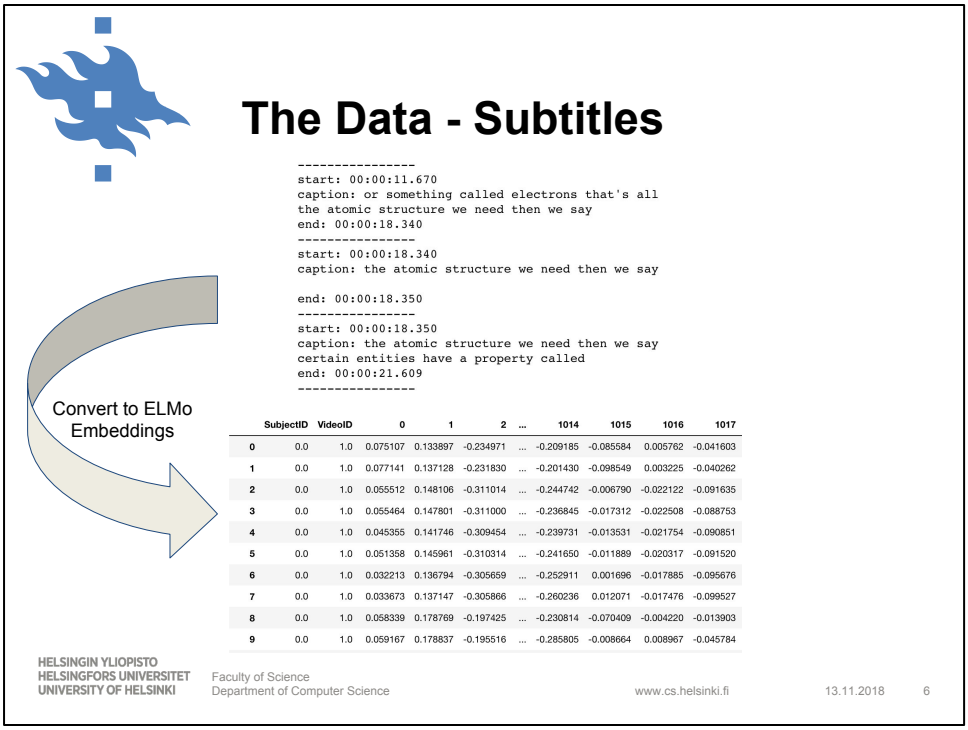
The ‘EEG’ data was not actually the typical 10/20 standard EEG data. It is more aptly described as pseudo-EEG data, with two additional and proprietary variables denoted as “Attention” and “Mediation”, whose calculations are not explained. Column ‘predefinedlabel’ is boolean indicating difficulty, where 1 = Difficult and 0 = Easy. The boolean column ‘user-definedlabeln’ indicates whether or not a given Subject was confused by the video, where 1 = Confused and 0 = Not Confused

The Data - Video




```
array([[0.24881373, 0.24881373, 0.24881373, ...,
        0.33064   ],
       [0.24881373, 0.24881373, 0.24881373, ...,
        0.33456157],
       [0.24881373, 0.24881373, 0.24881373, ...,
        0.33456157],
       ...,
       [0.21554431, 0.21554431, 0.21554431, ...,
        0.29366314],
       [0.21554431, 0.21554431, 0.21554431, ...,
        0.30934941],
       [0.21554431, 0.21554431, 0.21554431, ...,
        0.31719255]])
```


Each video was sliced into a frame every 0.5 seconds. This frame was then converted to grayscale and stored as a (360, 640) Numpy array.



Each video was uploaded to YouTube. The automatic subtitles generated by YouTube were then downloaded. These subtitles were converted into vectors using ELMo word embeddings. Each embedding contains 1024 dimensions. More information regarding ELMo can be found here: https://allennlp.org/elmo and here:https://github.com/allenai/allennlp/blob/master/tutorials/how_to/elmo.md

What We Have Done

- Data Structuring
  - Generated subtitles and associated sentence vectors for each video.
  - Sliced each video into half-second image files.
  - Merge(d) the subtitle and video data with the EEG data

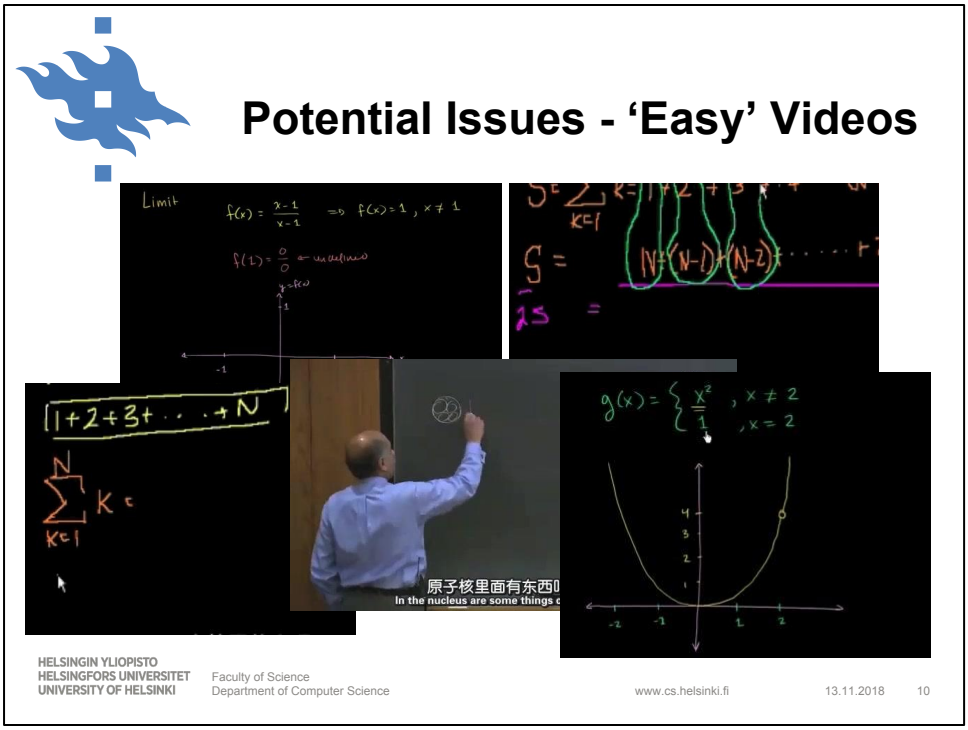
What We Have Done

- Experimentation
  - Attempted to replicate the paper “Confused or not Confused?: Disentangling Brain Activity from EEG Data Using Bidirectional LSTM Recurrent Neural Networks” by Ni et al. https://dl.acm.org/citation.cfm?id=3107513
    - Despite following the paper, achieved mediocre results
  - Used a Bidirectional LSTM with an added time-distributed convolutional layer as per A. Mehmani[1]
    - Achieved slightly better results
  - Tested less ‘deep’ methods, such as GBT, SVC, and RF
  - Used subtitle vectors to predict video difficulty

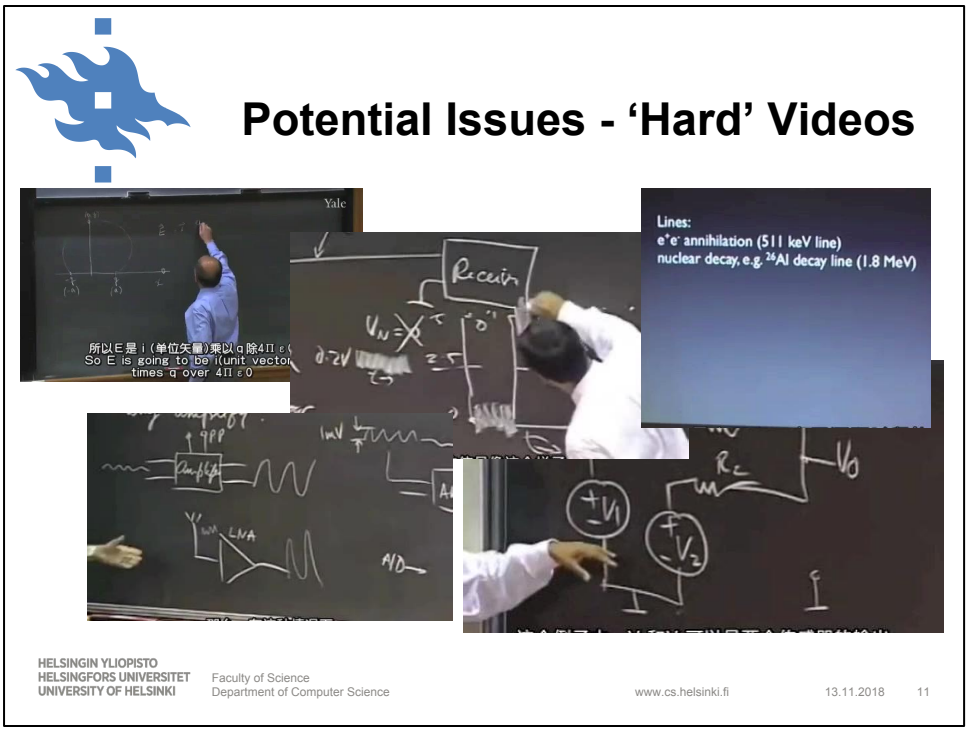
1: github:https://github.com/mehmani/DNNs-for-EEG-Signals

To Do:

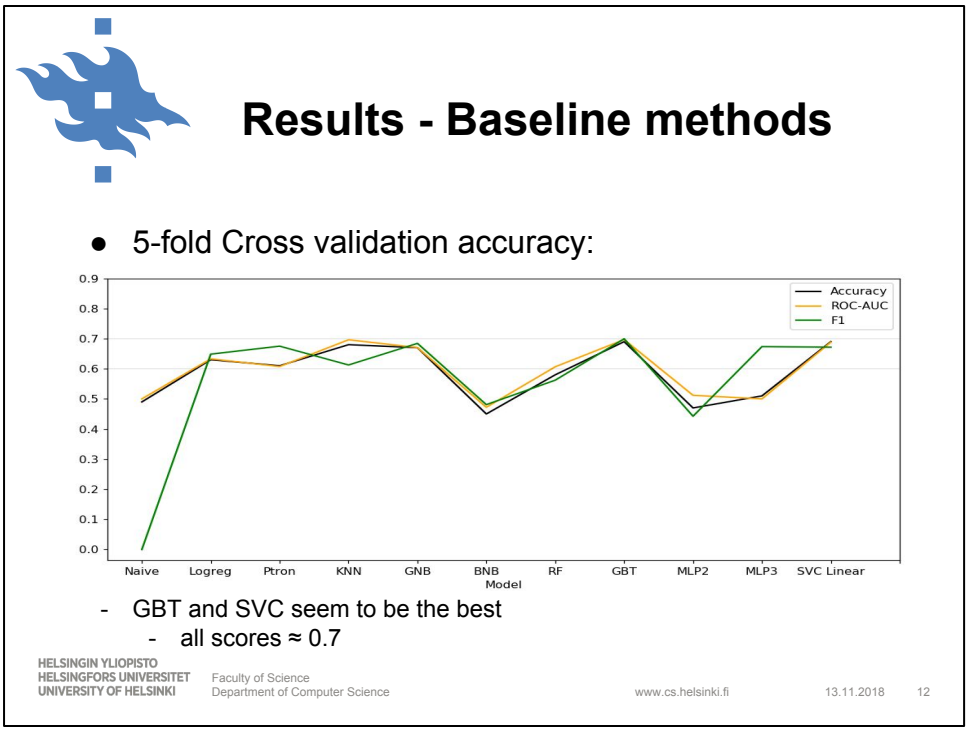
- Experimentation
  - Design a neural network that can take all three inputs: EEG data, sentence vectors, and video data, and produce a boolean (confused/not confused) predicted output.



One of the challenges with training a properly working model using the visual data from each video, is that the ‘Easy’ videos are trivial to distinguish from the ‘Hard’ videos, using nothing more than grayscale value. The videos with more black pixels, which come from Khan academy, can be classified as ‘Easy’...

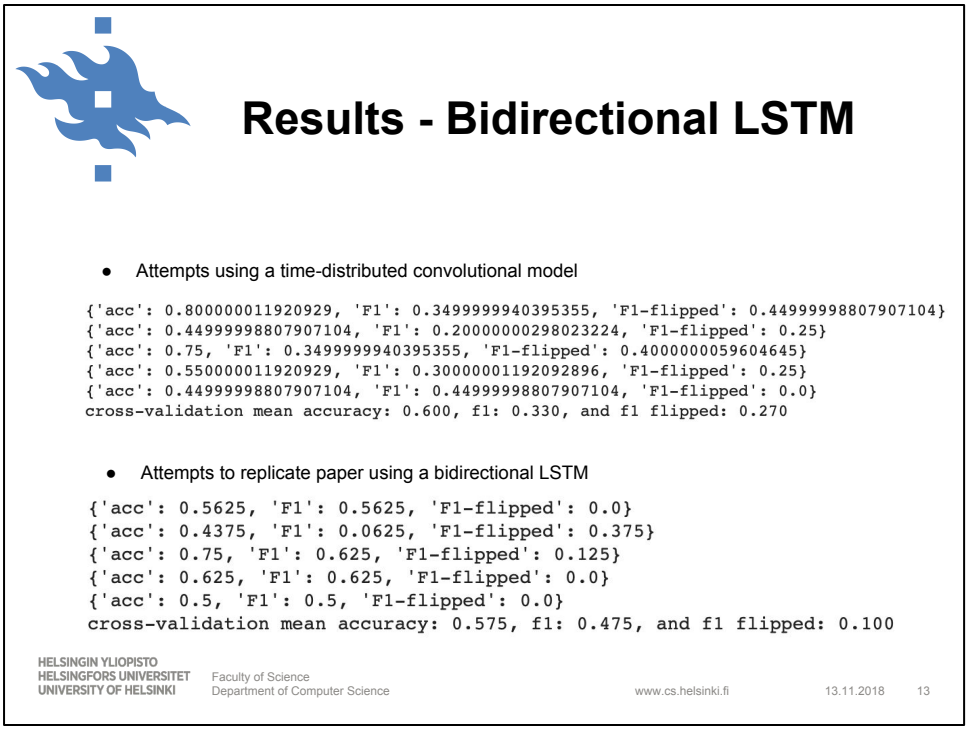


...whereas the videos with more ‘grey’ values are likely to be ‘hard’. One method to get around this would be to augment the dataset with modified images, such as ones where posterization has been applied, or by inverting the colors. Nonetheless, it is likely that the model will simply learn to distinguish the two categories based on the appearance of a person or a blackboard.



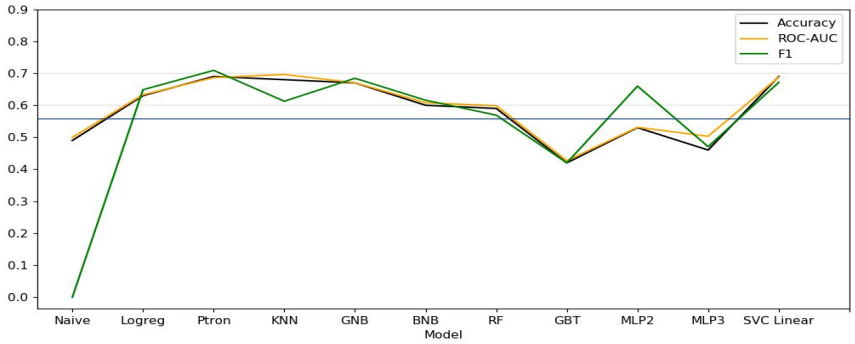
While the intention of this project is to use more sophisticated neural network models, it is worth considering simpler, generally faster models. Testing a variety of different method offers a worthwhile baseline with which to compare our LSTM.

Gradient boosted decision trees (GBT) and a support vector classifier (SVC) with a linear kernel gave the best performance. The SVC was much faster to train and gave results in a trivial amount of time (less than three seconds), whereas the GBT took considerably longer.

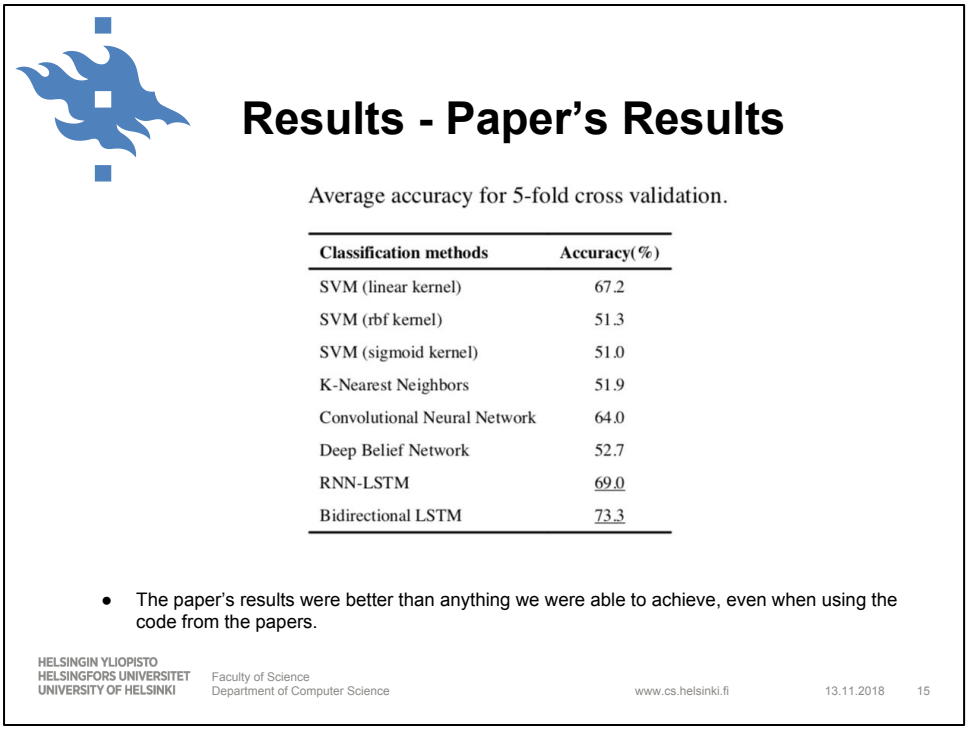


Both model architectures yield fairly inconsistent results. The way the data was cross-validated, 10 students were grouped into pairs (for a total of 5 pairs for 5-fold cross-validation). It appears that some students were easier to predict for than others, however these findings weren't consistently replicated across the two model architectures.

Results - Baselines with subtitle vectors



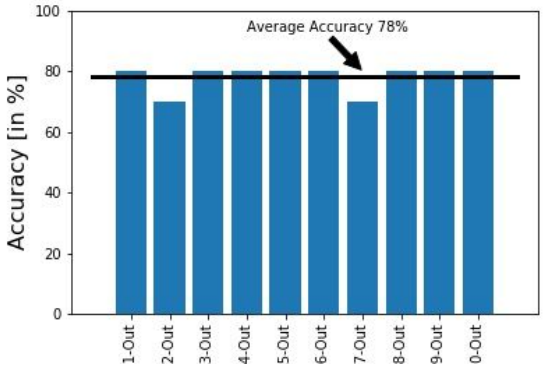
- Not much change except for GBT
  - Most likely due to reducing number of estimators
- Perceptron became better



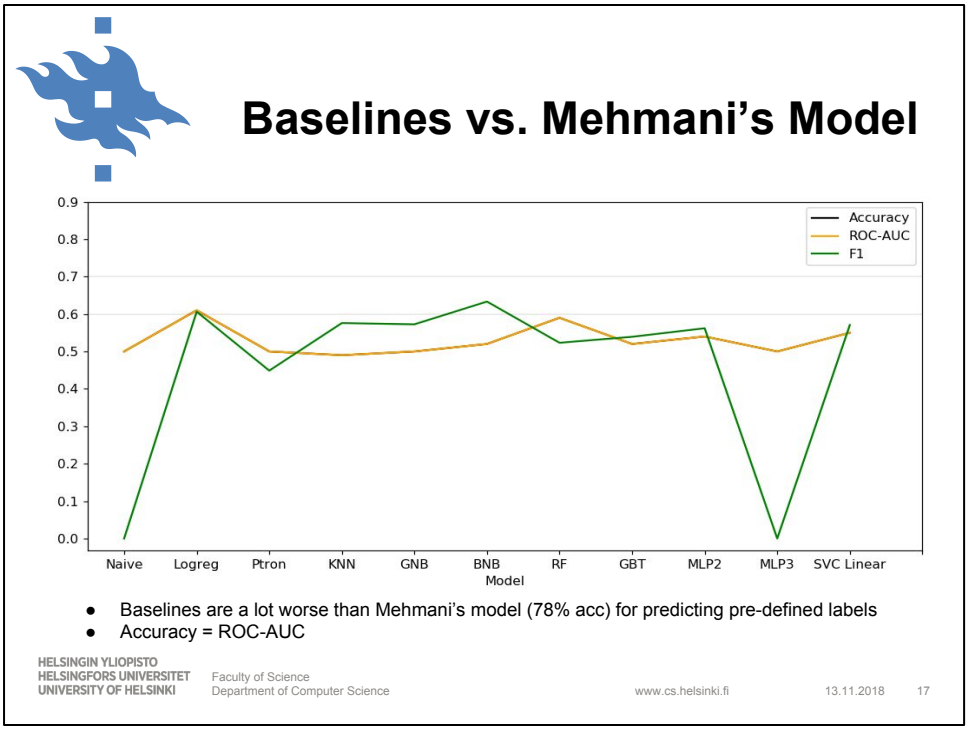
The paper's results significantly outperformed our models, even when using the exact same code that was presumably used to write the papers (identified through github here: https://github.com/nateanl/EEG_Classification

We are not exactly sure how the authors achieved such good results, especially because we are operating on the same data. We speculate that perhaps it is possible to achieve better results through a more advanced architecture than described in the paper.

Results - Mehmani's Model

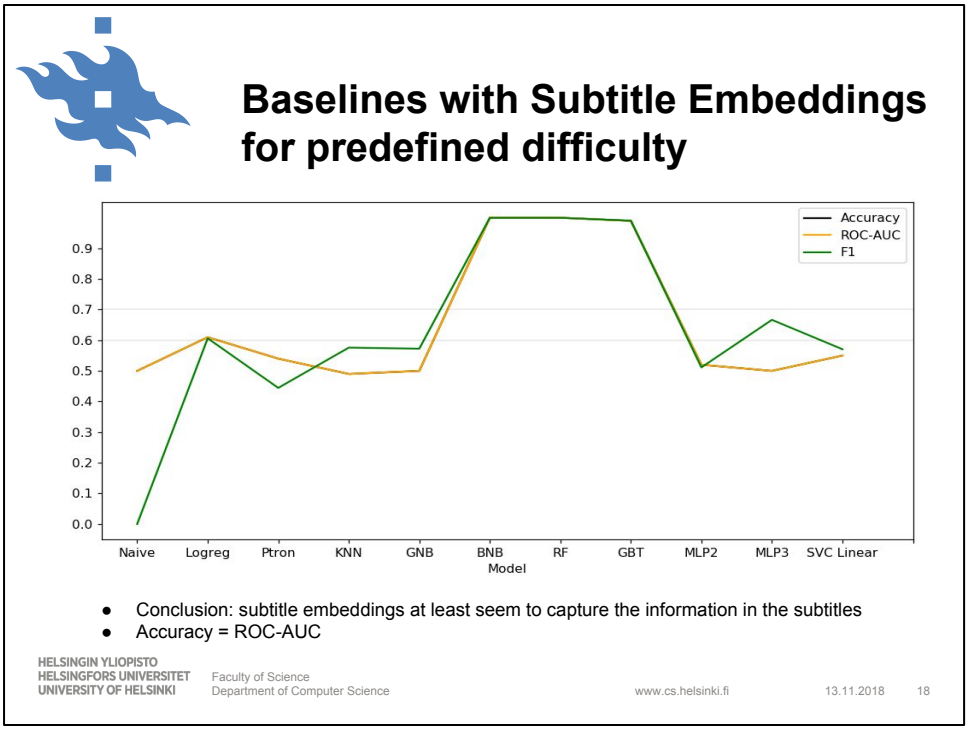


- Oddly enough, Mehmani is using the video difficulty label in his model, so he is actually predicting whether or not a video *might* be confusing, rather than if a student was actually confused. Still, the increased accuracy might be a compelling detail worth investigating.



Using baseline methods to attempt to predict the pre-defined 'hard' vs. 'easy' labels proved difficult when using only the EEG data.

Note: because of how the data is balanced, ROC-AUC and Accuracy lines happen to be the same.



As a sanity check, including subtitle embeddings demonstrates that it is possible to distinguish the difficult videos from the easy ones. This is likely due to the nature of the material discussed, as most of the ‘easy’ videos were algebra or geometry, whereas the ‘difficult’ videos involved electrical engineering or particle physics.

Note: because of how the data is balanced, ROC-AUC and Accuracy lines happen to be the same.

Thank you!