Sky Moodslider

DOCUMENTATION AND USER GUIDE

WALKER, AARON

Sky Moodslider

Table of Contents

A full version of the code can be found at the following links:

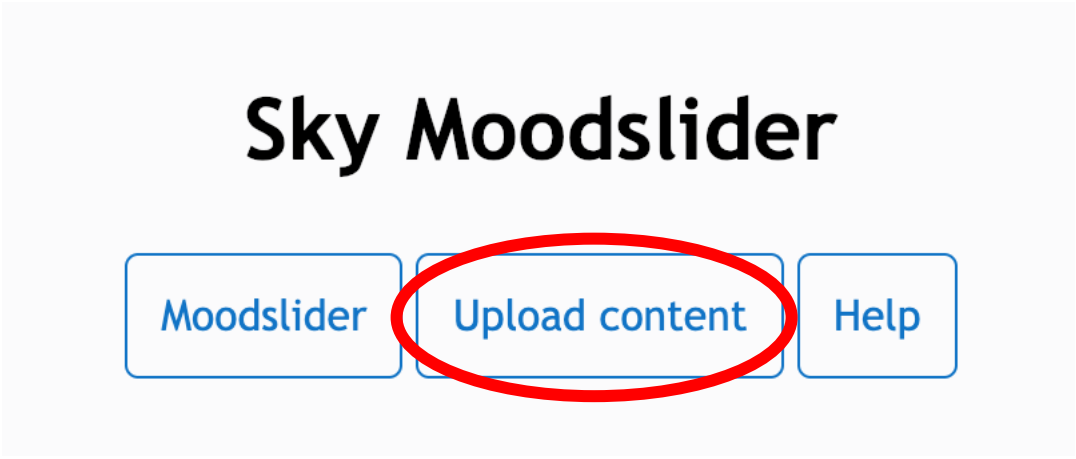
- GitHub - https://github.com/AaronWalker96/SkyMoodslider
- Google Drive - https://drive.google.com/open?id=15b1R6-5kZ6n_TKsr5sNSrbNJM5vwBYx3

User Guide

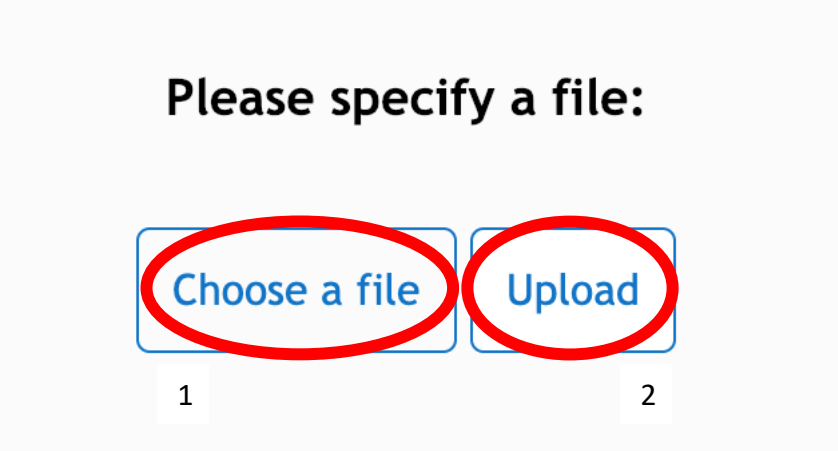
When the application is opened for the first time the user will be shown the Moodslider page with all the sliders in their default position and the content suggestion sections will show “No content”.

Uploading Content

To add programmes to the application, navigate to the “Upload Content” page by clicking the following button:



From this page, the user can select a file that contains a list of programmes using the "Choose a file" button. The selected file must be in a JSON format. Validation is carried out on the selected file to ensure it is of a valid file type.



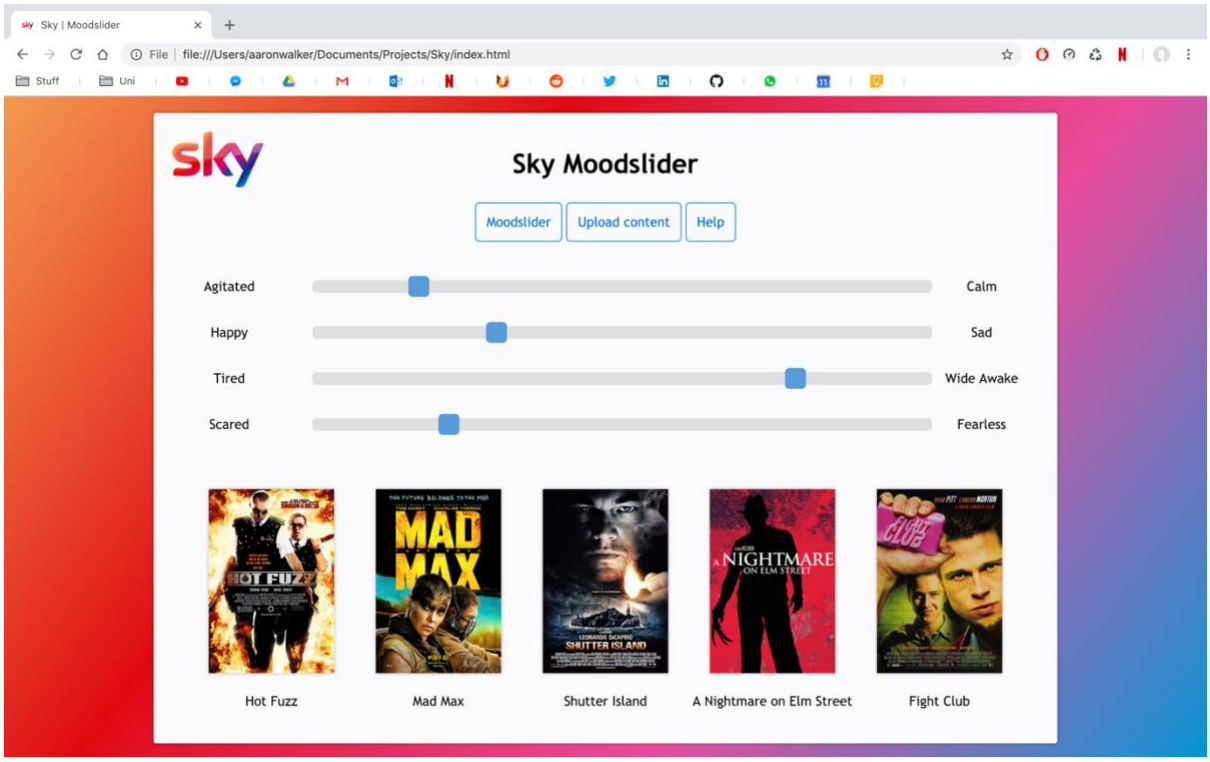
Once a valid JSON file has been selected, the user can then click the upload button to upload their file to the application. The user will then be taken to the Moodslider page.

I have included 2 test JSON files in the "Test JSON Files" folder.

- full.json – A full list of 65 programmes
- oneFilmPerMood.json – One film per mood

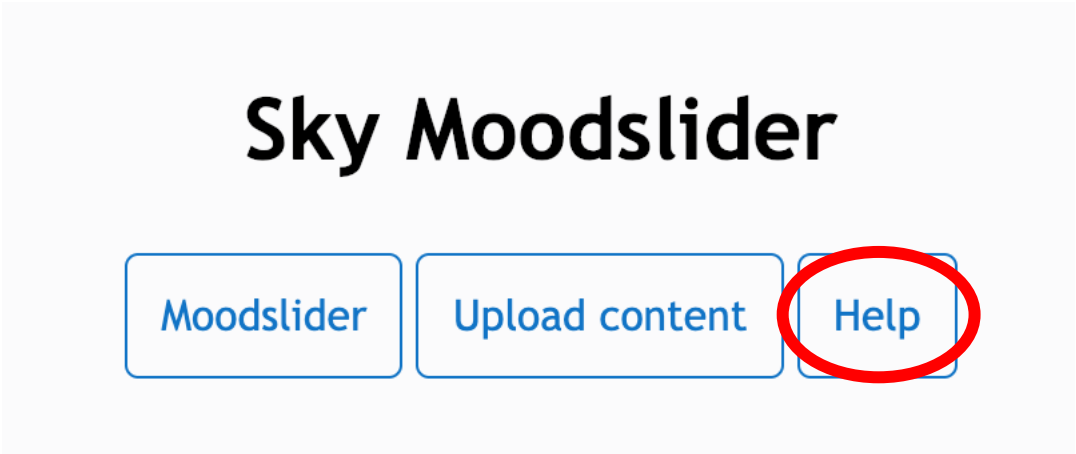
Using the Moodslider

The user can then begin to adjust the sliders to match their current mood. With each slider adjustment the recommended content will be updated to provide the user with useful programme recommendations.



Help

For additional help, the user can consult the Help page that can be accessed from the navigation bar. The help page contains a user guide to help the customer use the application.



Technical Details

Browser Details

The application has been built and tested in Google Chrome Version: 71.0.3578.98. The application has not been tested in other browsers.

Languages Used

The Moodslider application has been built using the following technologies:

- HTML – Website structure
- CSS – Website styling
- JavaScript – Upload document functionality, programme recommendation
- JSON – Data file containing programme information
- Git – Version control

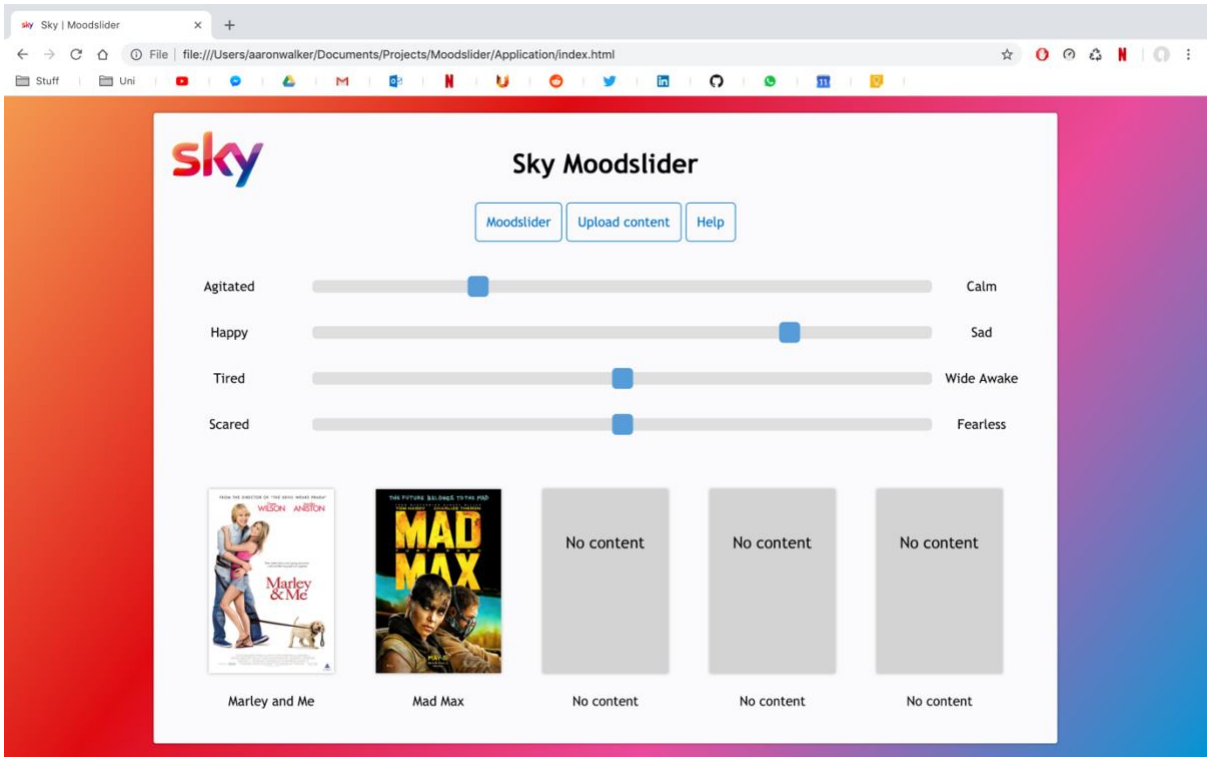
Features

Some key features of the file upload include:

- Once uploaded, the data file is validated and stored using `localStorage`. This means that the user does not have to upload the document each time they open the application. An alternative option for this feature is `sessionStorage` which allows data to be saved for one session, meaning that the data is lost when the browser window is closed.
- Once the "Upload" button is clicked, the file is uploaded, and the user is returned to the Moodslider page.

Some key features of the Moodslider page include:

- The programme suggestions will be automatically updated each time the user moves the slider, with no need for any further input.
- The user can move multiple sliders at the same time, this creates a mood profile that recommended programmes are selected to match.
- The user will never be offered duplicate programmes in the content slots. They will always be offered 5 unique programmes. If the JSON data file uploaded by the user does not contain 5 programmes that match the mood profile specified by the user, the matching content will be displayed, and the remaining slots will show "No content". This means that the user is not recommended the same programme multiple times.



Future Improvements

Machine Learning

As Sky is always looking to use innovative solutions, I wanted to include the use of machine learning in the Moodslider project. The implementation of machine learning would allow the application to suggest programmes that other users with the same mood profile have selected, meaning that more accurate suggestions can be given to the user.

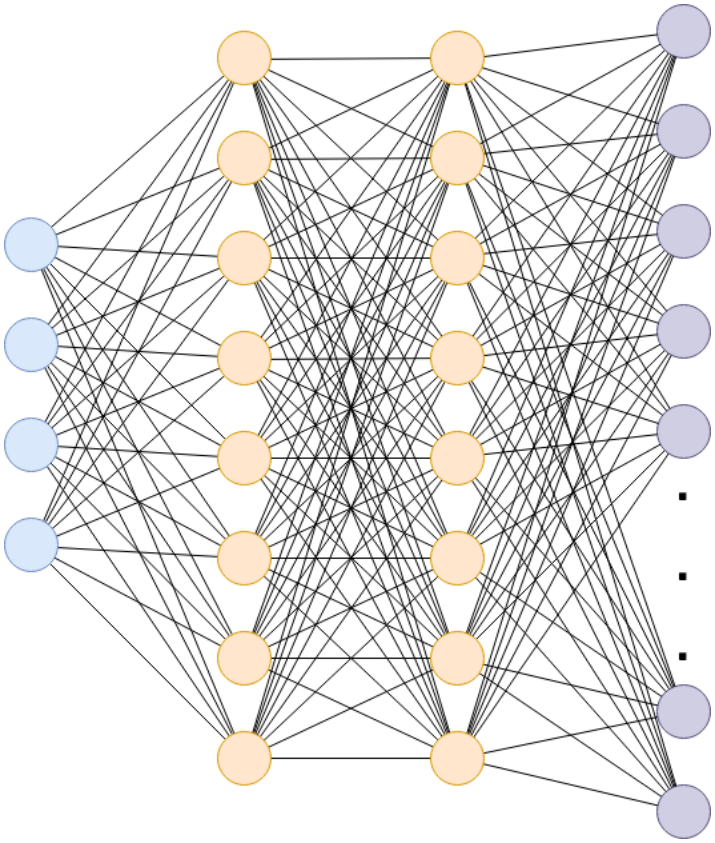
I have thought of 2 different machine learning techniques that could be effective in the Moodslider application context. The machine learning solution would need to be a classification model, where each programme would be represented by a category.

Logistic Regression

The first solution is logistic regression for machine learning. In the scenario of the Moodslider application the input would be the value of each of the mood sliders. The input is then processed by the algorithm and outputs the probability of the input matching a certain category. As each category is a programme the output probability would represent the likelihood that the user would enjoy the programme. The highest 5 probabilities would be recommended to the user.

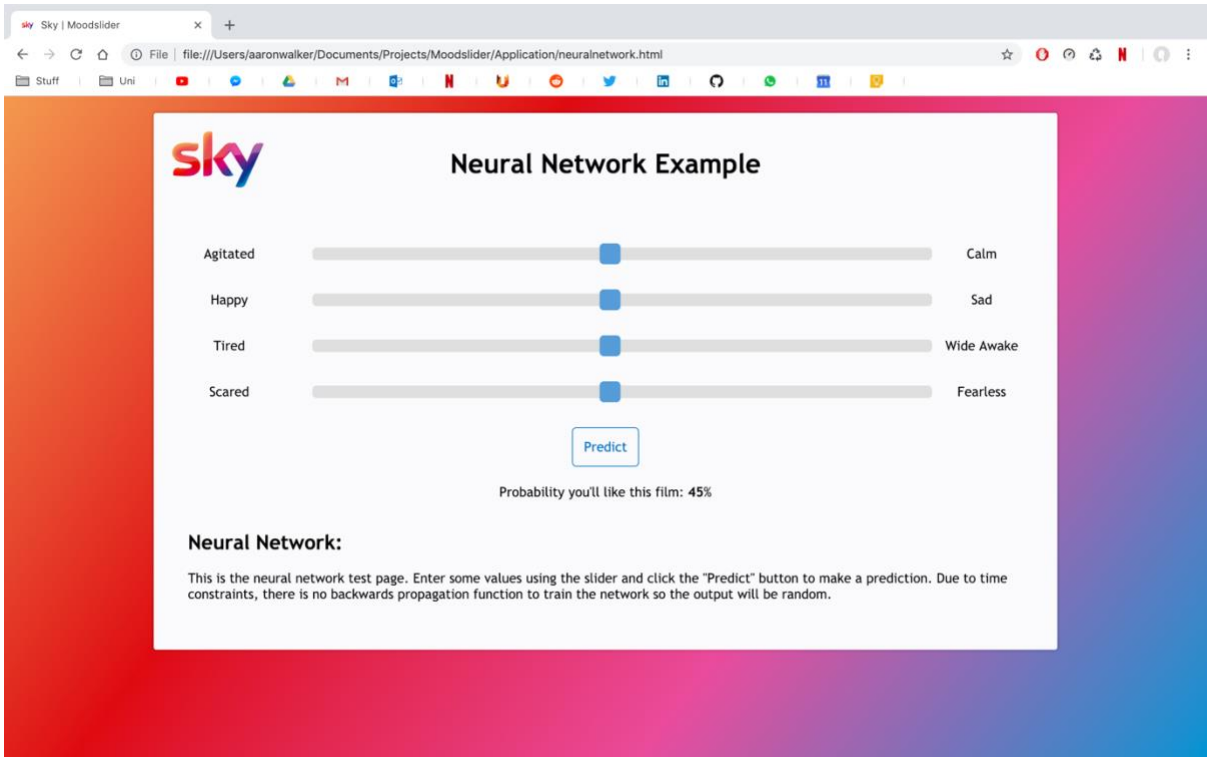
Neural Network

The second machine learning solution is a neural network. The neural networks design would consist of 4 input nodes that represent the values of the 4 mood sliders and X number of output nodes where X is the number of programmes loaded into the application.



As can be seen by the diagram, I have decided to include 2 hidden layers of 8 nodes each. The thought behind this design is that the nodes in the first part of the hidden layer would each represent one of the 8 moods, the next 8 nodes would represent the relationship between these moods and the compatibility of the programme in the output layer. The highest 5 probabilities would be recommended to the user. As neural networks are still being researched, there is no fully agreed upon way to construct a network so this design may need altering if the results are not as expected.

Due to the time constraints of this project I did not include any machine learning in the final version of the Moodslider application. However, I did include a partially built neural network to demonstrate how this solution might work. I started the construction of a neural network written in JavaScript in the `neuralnetwork.js` file. The neural network can be tested by accessing the `neuralnetwork.html` file and following the on-screen instructions. As this is a test of the neural network, there is only 1 output node to represent 1 programme.


Sky | Moodslider
file:///Users/aaronwalker/Documents/Projects/Moodslider/Application/neuralnetwork.html
Stuff
Uni
Neural Network Example
Agitated
Calm
Happy
Sad
Tired
Wide Awake
Scared
Fearless
Predict
Probability you'll like this film: 45%
Neural Network:
This is the neural network test page. Enter some values using the slider and click the "Predict" button to make a prediction. Due to time constraints, there is no backwards propagation function to train the network so the output will be random.