Machine Learning for Behavioral Data
Project Report

Julian Blackwell, Thomas Berkane, Ahmed Ezzo
Team 🦖 - Classtime

***Abstract*—Nowadays, digital education tools are becoming more and more popular, as their potential for augmenting students' learning is getting increasingly recognized. As most online services in today's world, the usage of these educational platforms generates considerable amounts of data, such as records of users' interactions with the service. Thus, there is the potential for the platform to put this data to good use by analyzing its contents in order to gain insight into ways to improve their service. This can take the form of making adjustments to the platform's design to maximize user contentment, finding out which aspects of the service are popular or not, personalizing the experience for each user, etc.**

In this report, we focus on a dataset of student interactions with the Classtime [1] startup's platform. In particular, using machine learning methods, we generate predictions and visualizations which aim to gain insight into the satisfaction of users of this service, and into what triggers this satisfaction.

I. Introduction and Research Questions

The research question that this report attempts to answer is: Can we predict reflection responses to the ”How do you feel about your learning progress” question based on the students' session interactions and the characteristics of the session? These may include response time, response correctness as well as additional session details such as the number of questions, feedback mode, et cetera.

Two approaches are attempted: an approach using aggregated features and machine learning models, and an approach using time series data as input to neural network models.

As extension work, we then turn to the task of identifying what an ideal learning session would look like. To this end, we look for clusters in our data which would indicate what session characteristics provide the best feeling of learning for students. As the results of this method are lackluster, though not totally devoid of insights, we introduce the approach of session-level happiness prediction, which proves to be more promising.

All the code used for this report can be found here.

II. Data Description and Exploratory Analysis

Through exploration of the raw dataset provided by Classtime, we made the observation as shown in Fig.1 (or A.1) that students which perform better on the platform also seem to be happier about their learning. On the other hand, poorly performing participants are more likely to be upset. The motivation of our research was thus to figure out whether other factors than correctness of answers could be used to explain student happiness and make predictions of it.

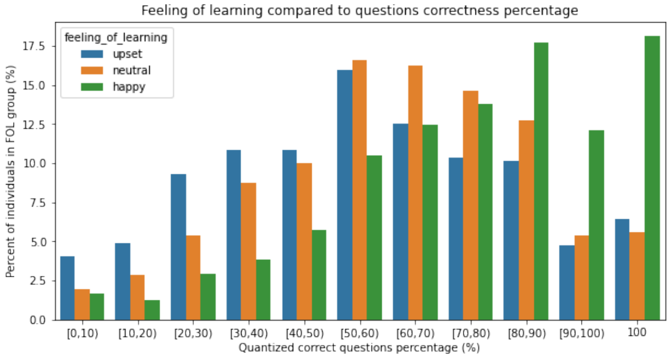

Figure 1. Participants' feeling of learning response compared to their performance

We build a dataframe from the raw data with columns participant_id, answer_time, mode, feedback_mode, force_reflection, timer, is_solo, video, image, correctness, n_answers, title_lang, title_len, title_topic and response. This dataframe can be used to aggregate participant answers, as well as be used for time series analysis.

The title_lang feature contains the language of the session's title and was extracted using the Google Cloud Translate API [2]. Fig.2 shows the most frequent languages in the dataset.

The title_topic feature corresponds to the topic's title. We noticed that most of the titles fell into one of the categories listed in Table I. We thus decided to craft word lists corresponding to each topic using the Empath library, and classify each title by looking at which topic its words fall into most frequently. Fig.3 shows the frequency of the different topics. The top 5 words for each topic are also shown in Table I. We notice that the topics are quite coherent within themselves and also quite diverse among each other.

In the aggregated dataset, we include the minimum, maximum, mean, and standard deviation of the answer time, the means of the video and image columns, and the mean and standard deviation of the correctness column.

To preprocess our data we take the following steps:

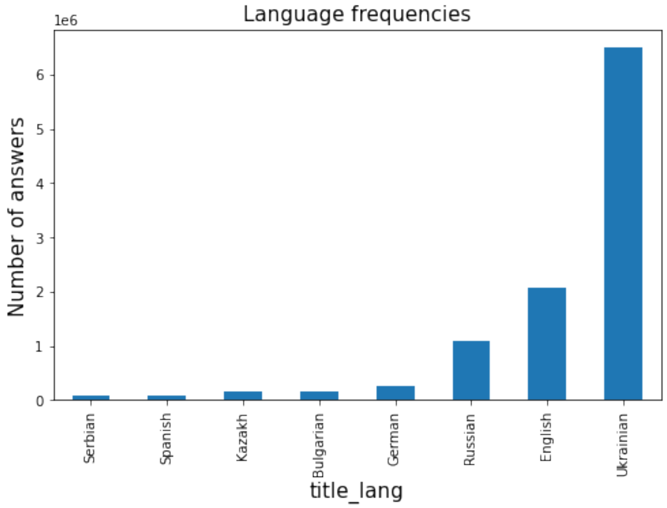

Figure 2. The 8 most frequent languages.

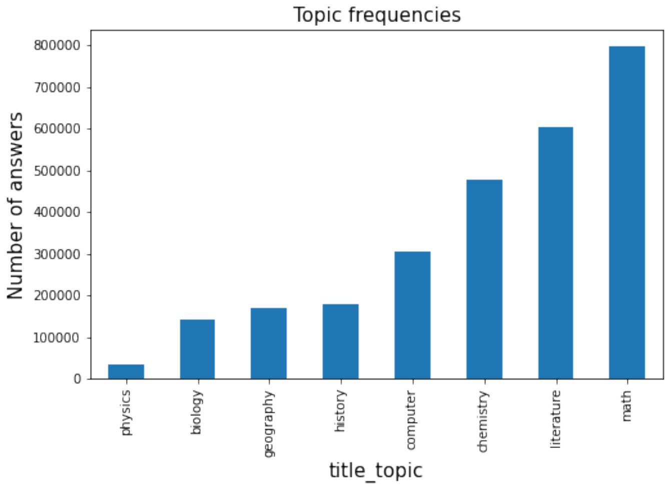

Figure 3. The frequency of each topic.

1) We impute missing data with a different strategy depending on the nature of the data. For categorical features, we replace missing data by the most frequent class. For numerical features, we simply replace with the mean of defined values.
2) Categorical features are also encoded so that they are mapped to integers (numerical scale).
3) In order for all features to be on the same scale, we also normalize our data.
4) Finally, in the case of the aggregated dataset, we also tried to compute pairwise products for all the features and add them to our dataset. However this did not improve the performance so we discarded this method.

Computer	**History**	**Chemistry**	**Geography**	**Math**	**Literature**	**Physics**	**Biology**
Computer	History	Equations	Geography	Mathematics	Reading	Processes	Diseases
Systems	history	Chemistry	Culture	Multiplication	Literature	Force	Brain
Programming	Mythology	Physics	Map	Science	Writing	Mechanics	Metabolism
Technology	Reformation	Biology	Region	Geometry	reading	Gravity	Fungi
Spreadsheet	Chronicle	Organisms	Country	Algebra	Text	Apparatus	Ecology

Table I
TOP 5 WORDS FOR EACH TOPIC.

Since the class labels are unbalanced (50% happy, 35% neutral, 15% upset), we generate a balanced version of the dataset using random undersampling to remove bias towards a specific label. This also slightly reduces the size of the dataset, leading to faster training but also to less data for training.

To perform predictions on the time series data, we use a fixed number of time steps (10 in our case). Since not all participants have enough answers, we add some padding to our data.

Additional features such as the time of day were also considered but did not seem to affect our prediction outcomes from our exploration. Answer correctness as well as feeling of learning appear independent to the hour of the day, with only some observable variance from smaller sample sizes in certain hour frames (see A.3, A.4).

To further motivate the scope of our experiments, we found that less than 5% of participants actually answer feeling of learning questions (see A.2). Only about 25% of sessions have at least one reflection question answered. This unfortunately means that there lacks valuable feedback for a majority of interactions and would be useful for teachers and schools to predict if their students are happy or not with the material based on their session interactions. To perform these predictions, we simplify our data by aggregating sessions' answering participants to equally weigh their input.

III. The Proposed Approach

A. Models for participant classification

We try two different approaches to classify the data: one which is based on an aggregated version of our dataset, and one which uses a time series version of the data.

1) Aggregated data: For the classification using the aggregated data approach, we try three different models: multi-class Logistic Regression, Support Vector Machines, and Random Forest. 80% of the data is used for training the remaining 20% for model evaluation and testing. To tune the hyperparameters, we apply 5-fold Grid Search Cross Validation.

We first attempt multi-class classification (classes = {happy, neutral, upset}), and then binary prediction (classes = {happy, not happy}). Then to get some insight into which features are essential to our task, we also look into the Random Forest feature importance using mean decrease in impurity (MDI).

2) Time series data: For time series data as input, we train different recurrent neural network architectures for the binary prediction problem (happy or not). We consider a simple RNN, a LSTM, and gated recurrent units (GRU). We compare these against a baseline simple fully connected neural network with a single hidden layer. We split the data such that 10 percent is for testing, and the remaining 90 percent are split 80-20 for training and validation data. The neural networks are trained with binary cross entropy loss.

B. *Predicting session happiness*

To predict session happiness, we first assume a happy-or-not scale for participants: 1=happy and 0=otherwise. Overall session happiness is then simply calculated by taking the mean of its participants' mapped response. This gives us a [0, 1] scale to represent the session and will be the initial prediction target we aim to model. To be clear, this score is equivalent to the fraction of students replying ”happy” to the reflection question, e.g a score of 0 would correspond to no participants answering ”happy” to the reflection question while on the contrary a score of 1 is equivalent to only receiving ”happy” feedback.

We first try some generic regression models and simple neural network architectures. More specifically, we compare Linear Regression, Ridge Regression (with grid search cross validation to find the optimal regularization parameter), and a Gradient Boosting Regressor. We also try two simple fully connected neural networks, one with only two hidden layers and another with eight, both trained with mean squared error loss. Most importantly, for model evaluation we contrast their performance against a baseline average-predictor, i.e the mean of training data labels is always predicted.

We then move onto a binary prediction task where we aim to predict whether a session passes a ”happiness quota” (e.g. at least 60% happy students) as hypothetically required by a school. For this we map the session labels to 1 if the happiness score is greater or equal the chosen quota, to 0 otherwise. Once more, we compare a simple fully connected neural network (2 hidden layers, binary cross entropy loss) with a Gradient Boosting Classifier and a Random Forest Classifier. These are evaluated against a most-frequent class baseline predictor.

For all above mentioned approaches, we use a 90-10 train-test split for model training and evaluation. Once again we try to gain some insight into which features are meaningful to our task so we inspect a Random Forest regressor's feature importances using MDI alongside a Ridge regressor's coefficients importance.

Finally, a large number of sessions only have a few reflection answers (see Fig. 4). This might introduce some unwanted variance into our data as sessions with very few data points might be an unreliable measure of the session's happiness. If there are not enough answering participants, one cannot guarantee that they are representative of all the session's students. We therefore also explore the impact on model performance of setting and increasing a minimum number of reflection answers requirement. That is, a session can only be used in training or testing a model if it has at least the required minimum number of answering participants. As can be observed in A.5 compared to A.6, increasing the limit ignores unreliable scores and moves to a more natural score distribution. However, as can be observed in Fig. 4, the number of available sessions quickly decreases. Indeed, if we simply set the minimum to 2, the dataset is already only 79% its original size. This decreases to about 47% for a minimum of 5, and down to 18% for a minimum of 10. We therefore need to be cautious not to reduce our training data to the point where we observe diminishing gains in performance such that models simply overfit to a small set of data. For this we do not increase the minimum requirement above 20 (at which we still have around 4000 datapoints $\sim$ 3% of the original size).

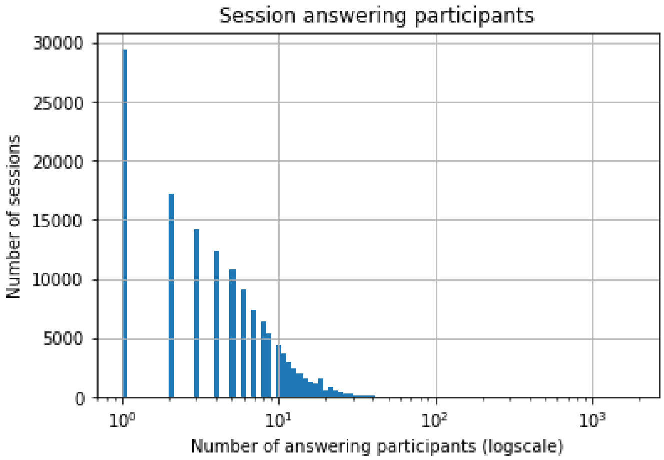

Figure 4. Number of session reflection answers

C. *Identifying clusters*

We look for clusters, meaning groups of datapoints which share common session characteristics or interactions, in our data with the objective of identifying what an ’ideal’ session would look like with regards to student feeling of learning. This is done by projecting the aggregated version of the data to two dimensions, using either Principal Component Analysis or t-SNE[3], and then plotting them. Note that we have only projected a randomly sampled subset of our data for visual clarity and performance reasons.

IV. Results and Evaluation

A. *Participant classification*

1) Aggregated: For the multi-class classification task on aggregated data, the baseline most-frequent class predictor achieves 0.5604 accuracy and $0.\overline{3}$ balanced accuracy.

Our considered models all improve (see Table II) upon the baseline accuracy (see A.7) and balanced accuracy (see Fig. 5). Although accuracy improvements are marginal, we are more interested in balanced accuracy especially since the class labels are unbalanced. The models better predict across all classes, most notably the Random Forest classifier who

Model	Accuracy	Balanced accuracy
Baseline	0.5604	0.3333
Logistic Regression	0.571	0.3870
SVM	0.572	0.3795
Random Forest	0.571	0.3988

Table II
AGGREGATED BEST CV-SCORE PERFORMANCE

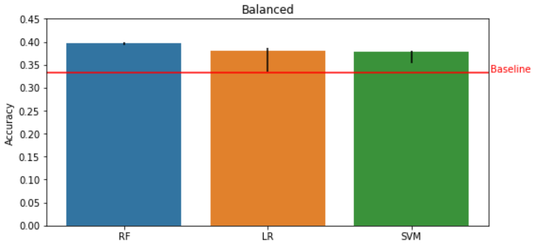

Figure 5. Aggregated multi-class classification

reliably achieves a better score regardless of our considered hyperparameters. The logistic regression and support vector machine alternatives are less robust as there is a greater performance variance given the considered hyperparameters in the cross-validation step (see Fig. 5).

Dataset	Accuracy	Balanced Accuracy
Full	0.5691	0.4092
Balanced	0.4539	0.4539

Table III
BEST RF RETRAINED

Results when retraining the best Random Forest model on the full dataset compared to a balanced subset are in Table III. Although in terms of overall accuracy performance largely decreases, there is a major gain in balanced accuracy, so training on a balanced dataset improves predictions for all class labels.

For the binary prediction task, the baseline most-frequent class predictor achieves 0.5604 accuracy and 0.5 balanced accuracy (also equal to the ROC AUC).

Model	Accuracy	Balanced accuracy	ROC AUC
Baseline	0.5604	0.5000	0.5000
Logistic Regression	0.648	0.627	0.6867
Random Forest	0.645	0.629	0.6864

Table IV
AGGREGATED BINARY CLASSIFICATION

As seen in Figs. 6, 7, 8 and in Table IV, our considered models perform similarly and both improve upon the baseline. The Random Forest performance is slightly more robust than the Logistic Regression from a given choice of hyperparameters.

Results when retraining the models with the best hyperparameters on the full dataset compared to a balanced subset are in Table V. Once again, although accuracy sees a small decrease, balanced accuracy and ROC AUC (which are equal in this case) see noteworthy gains. Training on a balanced dataset has again helped performance. Moreover, it seems that the Random Forest now slightly outperforms the Logistic Regression.

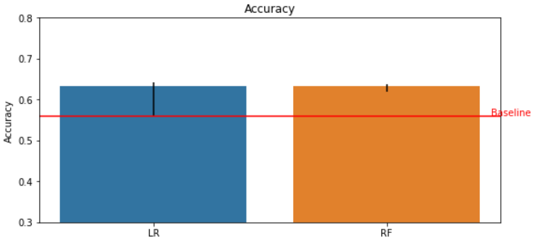

Figure 6. Aggregated binary accuracy

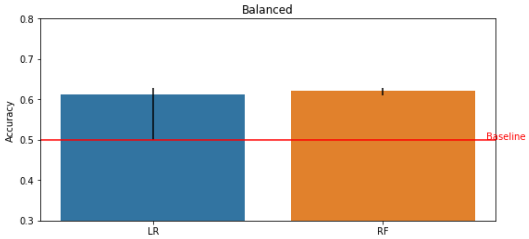

Figure 7. Aggregated binary balanced accuracy

Dataset	Model	Accuracy	AUC
Full	Logistic Regression	0.6410	0.6251
——	Random Forest	0.6483	0.6367
Balanced	Logistic Regression	0.6368	0.6368
——	Random Forest	0.6432	0.6432

Table V
BEST AGGREGATE BINARY MODELS RETRAINED

Training with added polynomial features was then tried but gave diminishing returns (0.6395 v.s original 0.6432 for the Random Forest) so we discard this method.

Finally, we inspect the feature importances using MDI of the Random Forest classifier (see A.8). Features correlated with participants' answers correctness seem to carry large importance (as hypothesized), followed by time to answer questions, and the session title length. Features with little to no importance include session settings such as the mode, forcing a reflection answer, or self-study mode.

2) Time Series: The test performance of our considered models on the 10 time series steps data is noted in Table VI and Fig. 9.

Within the recurrent neural network architectures, the gated recurrent unit model seems to slightly outperform its alternatives. However, there is no observable improvement in comparison to the fully connected baseline.

When varying the number of time steps (see Fig. 10), notable improvements are seen by increasing from 5 to 10

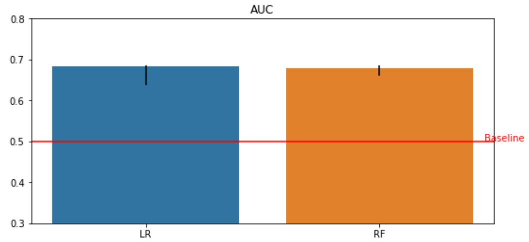

Figure 8. Aggregated binary ROC AUC

Model	Balanced accuracy	AUC
Baseline (fully connected)	0.6162	0.6728
Simple RNN	0.6116	0.6684
LSTM	0.6149	0.6739
GRU	0.6162	0.6740

Table VI
TIME SERIES MODELS PERFORMANCE (10 STEPS)

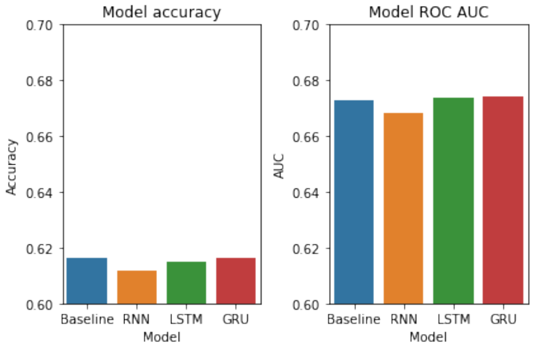

Figure 9. Time Series Models Visualized

and again to 15 but beyond that only marginal gains are observed. The computational tradeoff from increasing the number of time steps is not worth the gains (370MB dataset for 5 steps, 10 steps is 740MB, 15 steps is 1.09GB, 20 steps is 1.45GB and 30 steps is 2.17GB).

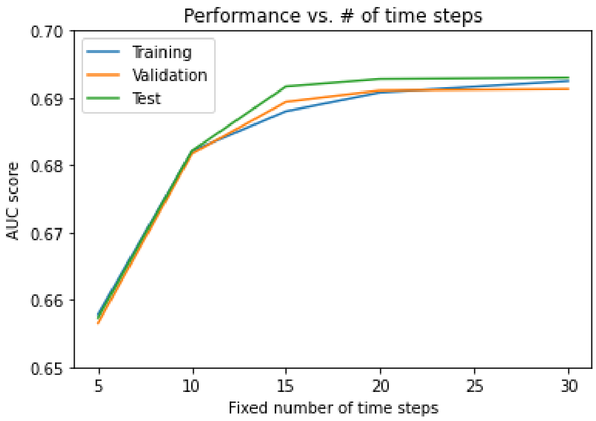

Figure 10. Time steps varied

B. Session happiness prediction

Now to predict a session's happiness score, the considered models in order of performance are listed in Table VII.

Model	MSE	MAE
Baseline	0.1146	0.2846
LinearRegression	0.0989	0.2540
Simple NN	0.0976	0.2514
Deep NN	0.0964	0.2513
GradientBoostingRegressor	0.0964	0.2505

Table VII
SCORE PREDICTORS PERFORMANCE (FULL DATASET)

All improve upon the baseline, with the Gradient Boosting Regressor performing the best. Now by increasing the minimum number of required reflection answers (see Fig. 11), although the baseline error gradually decreases, so does the MAE of the Gradient Boosting Regressor and of an added Ridge Regressor (regularization parameter chosen by cross validation). On the other hand, the neural network architectures appear to sometimes be victim of being stuck in a local (sub-)optima (likely due to problematic initialization, see A.9).

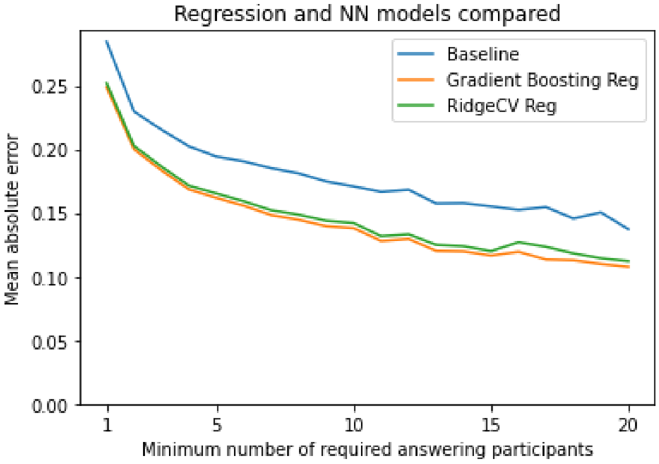

Figure 11. Increasing required minimum of reflection answers

We inspect the Ridge Regression coefficient features (see A.10) as well as a Random Forest regressor's feature importances (see A.11). Once again, the same features relating to correctness and answer time seem to hold the most importance, while session mode settings are least significant.

Moving onto the binary prediction of whether a session passes a ”happiness quota”, the baseline predictor achieves a balanced accuracy of 0.5. We observe that on the full dataset a simple fully connected neural network already achieves a balanced accuracy of 0.6722 in comparison.

Increasing the required minimum reflection answers also has a positive trend on balanced accuracy (see Fig. 12). Performance returns only slightly become unpredictable above

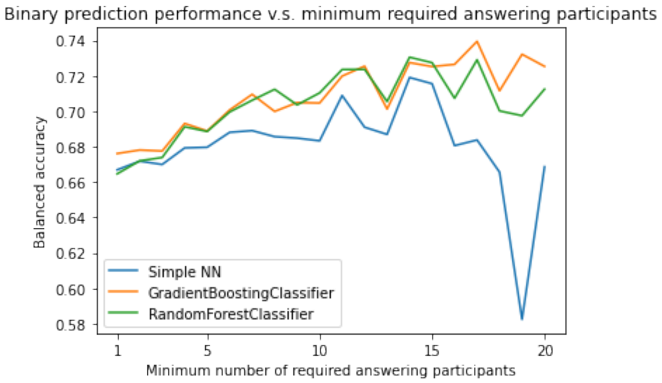

Figure 12. Binary prediction: increasing minimum answers

a quota of 10, and sees diminishing returns above a quota of 15, notably for the neural network which falters (likely due to insufficient training data). These still perform better than the baseline nonetheless. Overall the obtained results are quite promising, especially since we are performing predictions on session data aggregated into a single row. More complex models with added features that might take a higher dimensional input (e.g. n representative participant rows) are left to be explored.

C. Identifying clusters

The resulting projection using PCA can be seen in Fig.A.12 and those using t-SNE in Fig.13. As we found the results of t-SNE to be more interpretable, perhaps due to the method's superior handling of nonlinear data, we focus on it in further analyses.

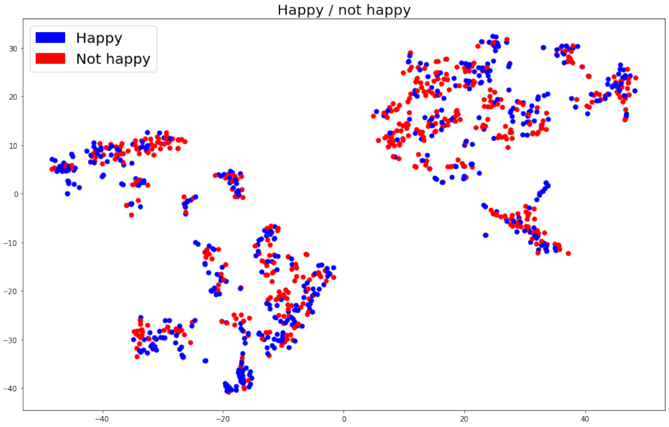

Figure 13. Projection of a subset of the datapoints using t-SNE.

In both of the plots, we show in blue the datapoints whose response to the feeling of learning reflection was 'happy', and in red all the other ones. We notice that the red and blue points are not well separated. This lack of separation also persists after attempting the same projection using only subsets of the features in our data.

The fact that happy and not happy datapoints are not separated well gives further evidence for why our models' performance is limited for the prediction of the students' answers. Indeed, perhaps the boundaries between the happy and not happy points in the data at our disposal are inherently muddled up, making prediction difficult.

Nevertheless, the way that the points are being projected by t-SNE can reveal interesting information about the structure of our data. Fig.14 shows in black the points where feedback mode is manual and in orange where it is per-question. We observe that the two main clusters in our projection are separated because of this feature. We can thus infer that this feature is important in structuring our data into two broad categories, although these seem to be mostly independent of student feeling of learning.

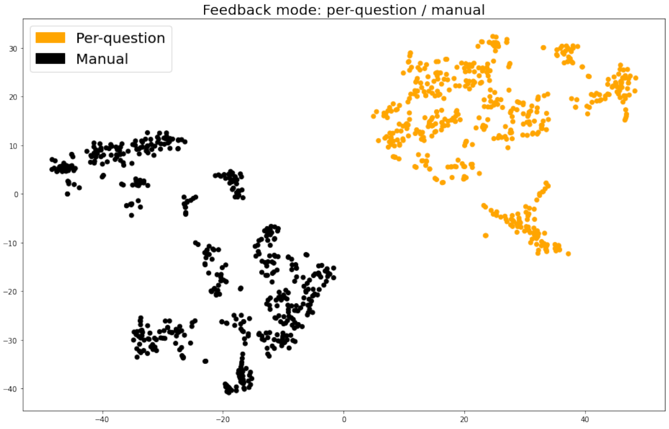

Figure 14. Projection of a subset of the datapoints using t-SNE.

Moreover, Fig.15 shows the same data projection, but this time the darkness of the blue corresponds Ahto how high the datapoint's mean_correctness is. We can see that this feature does not really contribute to forming clusters in the projection, but instead we can observe a continuum in the shades of blue. This indicates that there are no strong groups of datapoints with certain values of mean_correctness, but that many different values are present in the dataset.

Thus the method of looking for clusters in our data, while it still gives us insights into its structure, is not as effective as the session-level happiness prediction method for finding 'ideal' session characteristics.

V. Conclusions and Future Works

A. Limitations and possible improvements

First, the data used in this report considered each unique participant identifier as a unique person. However, this presents a big limitation as we only have access to a user's interactions within a single session before performing a prediction on his/her response to the feeling of learning question. These responses can be highly variable between sessions and topics and likely depends on the user's previous

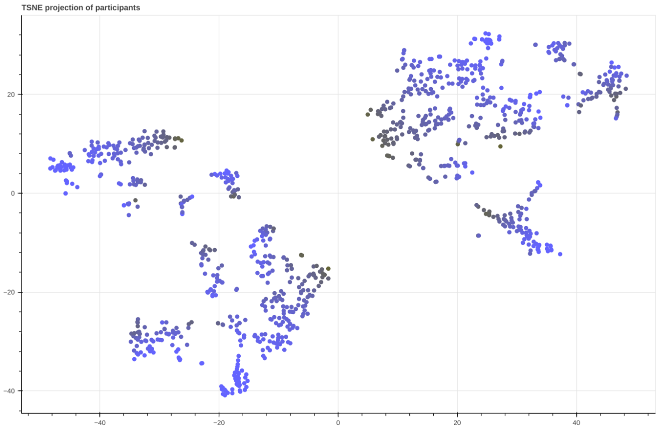

Figure 15. Projection of a subset of the datapoints using t-SNE. The darker the shade of blue, the larger the mean_correctness.

interactions. In other words, the data does not account for ”*behavioral variance*”, and it is difficult to build robust models without associating multiple session interactions to a single user. For future work, it might have been interesting to point out if some questions better help students’ progress if they are just beginning or if they have already had experience on a same/similar topic. It would have also been helpful to identify users previously struggling with some material for future reference, e.g. one might detect student frustration with repeated low performance on a given set of questions/topics and predict an upset response.

Another possible improvement, given availability of computational resources, would be to increase the number of features used in training our models. This could for example take the form of using one-hot encoding for our categorical features, which up to now are encoded on a numerical scale. This is in general not a good practice but one-hot encoding was an obstacle to our data aggregation and augmentation techniques as well as prohibitive in terms of dataset size.

Other features which could be added if dataset size wasn’t an issue include features extracted from the question titles and teacher-specific features using the teacher IDs.

B. *Overall insights and conclusion*

In this report, we started from the observation that highly performing students were likely to be happier about their learning. We found other features such as answer time, among others, that also slightly affected responses. However, due to the nature of the data and the construction of our task, we found that we could only predict participant happiness to some degree, and that predicting session happiness might be a more feasible task. Given appropriately prepared data and an optimized model, there is potential to achieve performance that is valuable for teachers using Classtime to generate feedback on their sessions in order to maximize their students’ happiness.

References

[1] Classtime, https://www.classtime.com/.

[2] Google, https://cloud.google.com/translate.

[3] L. van der Maaten and G. Hinton, “Visualizing data using t-sne,” 2008. [Online]. Available: https://jmlr.org/papers/volume9/vandermaaten08a/vandermaaten08a.pdf

Appendix

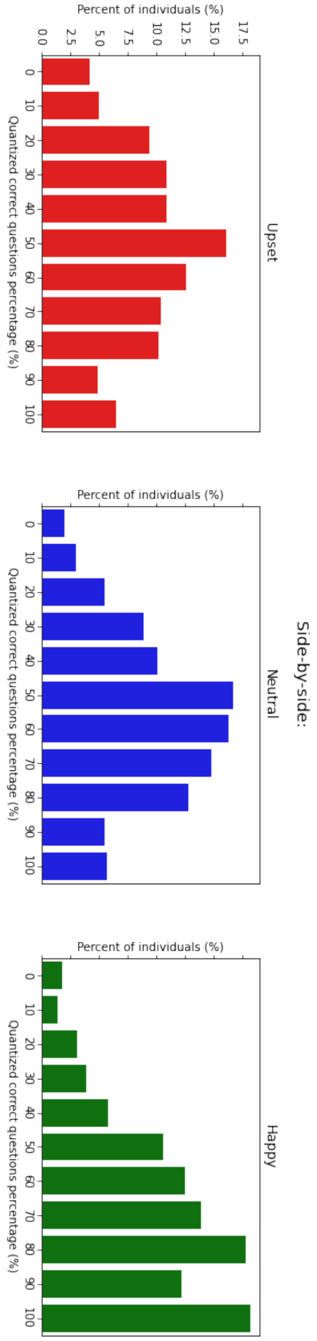

Figure A.1. Participants’ performance and feeling of learning

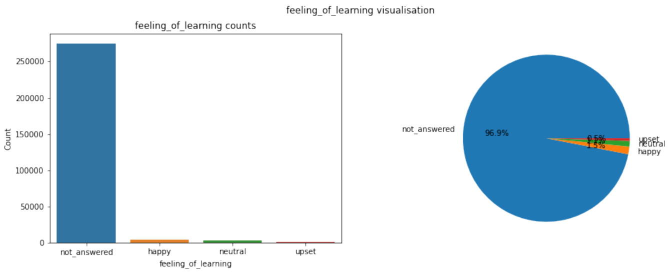

Figure A.2. Reflection answers

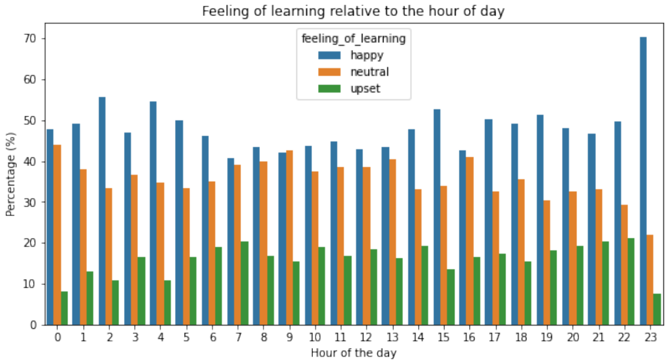

Figure A.3. Feeling of learning relative to hour of the day

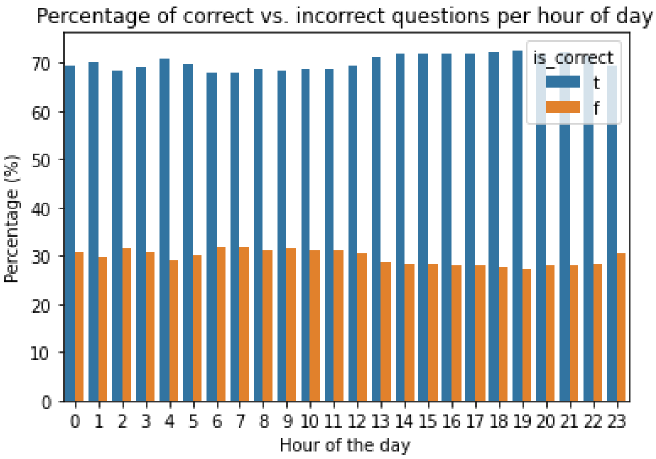

Figure A.4. Answers correctness relative to hour of the day

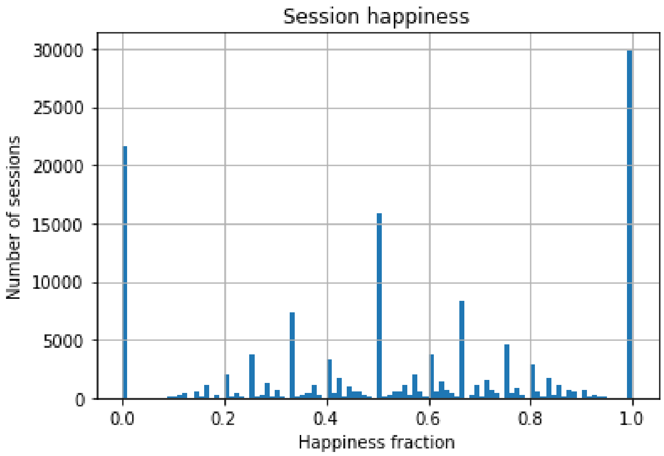

Figure A.5. Sessions happiness score

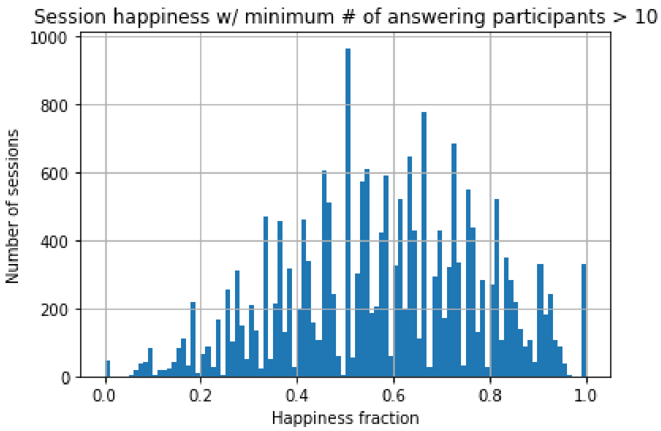

Figure A.6. Sessions with at least 10 reflection answers happiness score

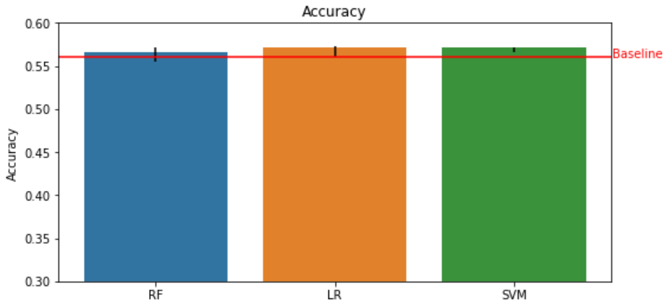

Figure A.7. Aggregated data multi-class accuracy

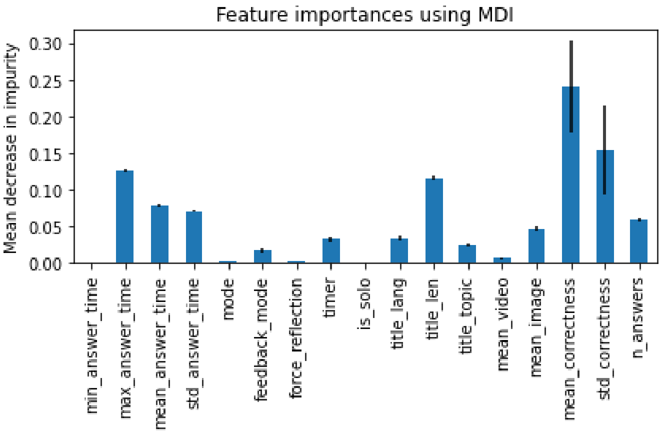

Figure A.8. RF feature importances using MDI

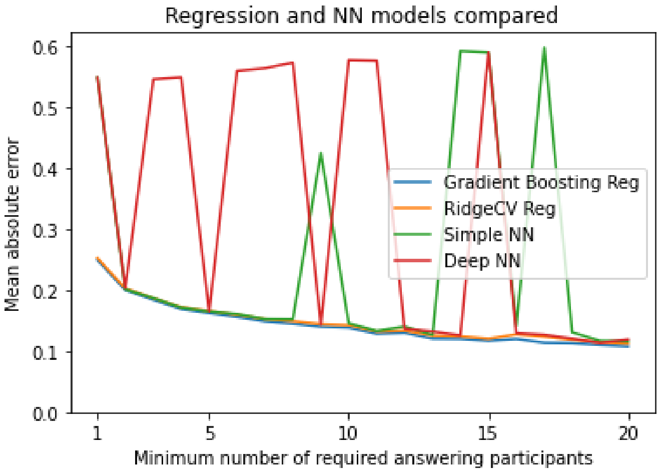

Figure A.9. Increasing required minimum of reflection answers

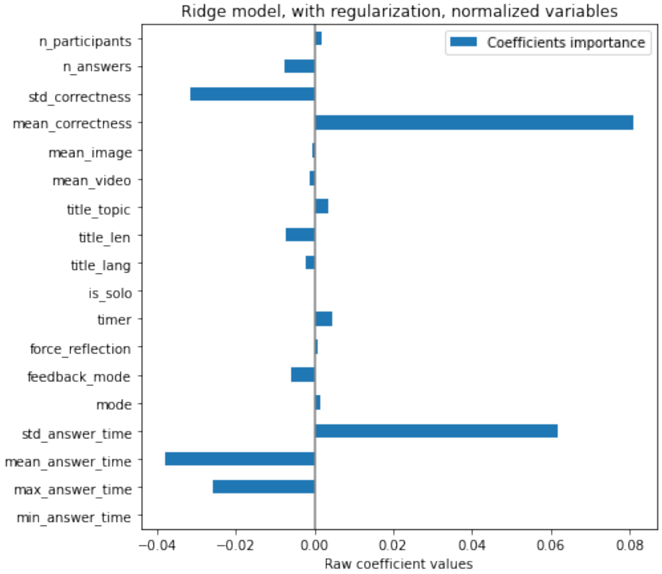

Figure A.10. Ridge regression coefficients (min answers = 20)

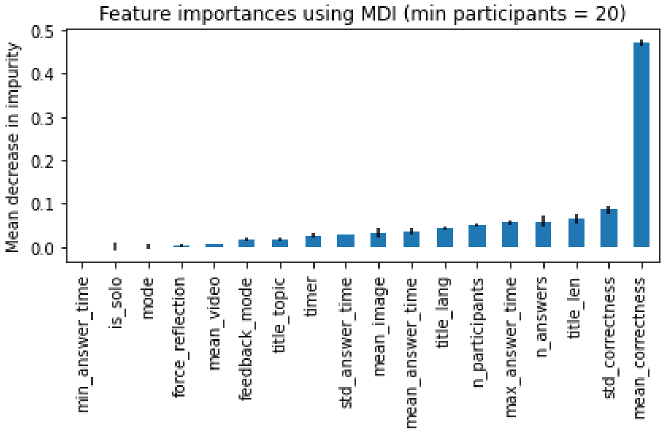

Figure A.11. Random Forest regressor feature importances using MDI

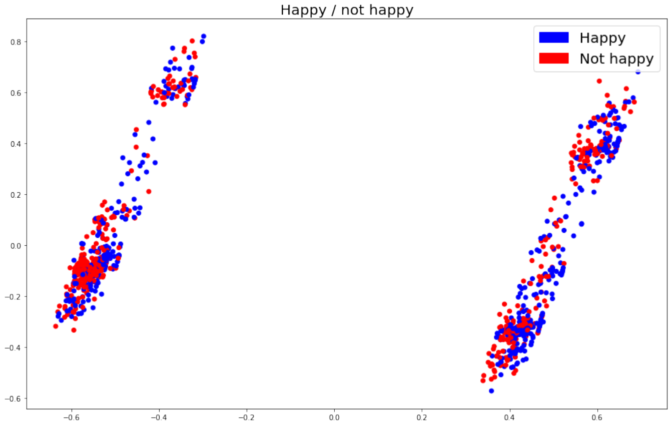

Figure A.12. Projection of a subset of the datapoints using PCA.