Jeffrey Dean

Mohit Agarwal

NLP: A Love Story

Natural language processing has an enormous potential to solve a lot of problems that arise in a society that increasingly finds its information online. One valuable form of NLP is applying analytics to dating profiles and to predict how pairs of people can be matched, just off small pieces of data. Dating as a process has drastically moved to online meetups rather than in person, partially due to the Covid-19 pandemic. Users want to provide the least amount of data as possible for privacy, so using language prediction for what information is provided, will be immensely powerful.

NLP Problem: Apply a series of NLP ML models on "biography" essays to predict attributes of interest about a person. For example: 'Age', 'Education level', 'Job_group' and 'drinking habits' taken as representative labels in this project.

Motivation:

- What makes two people a good match for one another?
- What factors determine accurate and relevant classification on online dating platforms?
- Could profile descriptions "Biography" text help classify users better?

Solution Planning/ Methodology:

- The NLP problem can be solved using the Classification **ML model.**
- **As part of the Classification problem** : Multinomial (Age, Education and Job) and Binomial for drinking labels will be classified based on analysis results from "Bio" text.
- Moreover, Clustering technique has also been explored to segregate similar user profiles (Un-Supervised ML). This is an additional objective in the project.

As part of **Literature Review** exercise team identified the following **paper** as a good starting point to develop a solution framework**:**

Research Paper Title: Dating Profile Age Analysis – a Supervised Learning Application, Arielle Friedman, Rushil Singh, Ao Ao Feng[1]

Paper Review Summary: In the reference paper, many aspects of the work overlapped with material from our course but also applied NLP methods in a different manner. The paper explores if we can use ML methods to feed an agent model input and predict the age of the user of a dating app or website. This is performed through a TF_IDF vectorizer process and then a multinomial logistic regression is used to run the tests assigned. Loss is evaluated to measure the success of the model through a 5 cross fold analysis and success is assessed based on percentage of correct predictions. For model validation, the data is evaluated against the probability of random chance that the words found are in one of three categories (20s, 30s, 40s) so 1/3 or 33% chance. Their proposed model performed at a success rate of about 54% which they claim is good because it's about 20% better than random chance but given our experience in seeing that a good model has an F-score of about >85%. Unless 54% in this domain can be regarded as high, there is scope of improvement in the implementation of the model.

The paper offered good insights to learn which methods could work on real world data and gave us perspective on future improvements which could be implemented in our project. For instance, TF_IDF vectorization worked well in the paper and showed us a good criteria of success: accuracy of predictions and this was used as reference in developing model pipelines in our project. One valuable lesson from the paper was to reduce the number of label categories available in a topic to attain more controllable analysis outcomes. Hence, we bracketed the ages into 6 distinct groups that were relatively balanced based on the ages of our users. The validation model approach for our project was more exhaustive compared to the paper. For analysis of results: heatmaps, iteration graphs and using F_score techniques were implemented. Referred paper also helped us include a clustering model portion in our project based on the research paper's use of looking for the most common words in a category. If they found words commonly associated with the factor of interest such as age, then we could also cluster words to understand the user better when there are multiple columns of data to address. We conjectured that these clusters could give an insight of who is searching for whom and help segregate similar people to understand what common words define their clustering pattern. This could help offer users suggestions on bio summary text during profile creation to increase the chance of favorable match on the dating app.

Summary of Models used in project to solve the identified NLP classification problem:

Baseline Model:

Model Name	Objective	Label(s) Predicted	Number of Cases (Code files)
Naïve –Bayes	Multinomial classification	Age, Education, Job, Drinking	4
Logistic Regression	Binomial Classification	Drinks_freq (0 or 1)	1

Advanced Model:

Model Name	Objective	Label(s) Predicted	Number of Cases (Code files)
RNN with LSTM (dropout)	Multinomial classification	Age, Education, Job	3
RNN with LSTM (dropout)	Binomial Classification	Drinks_freq (0 or 1)	1
CNN (with varying architecture)	Multinomial classification	Age, Education	2
Distill-Bert	Binary classification	Drinks_freq (0 or 1)	1
LSTM (with recurrent dropout)	Binary classification	Drinks_freq (0 or 1)	1
K-means	Clustering	Unsupervised (K = 6)	1

Data Acquisition: Data obtained from Kaggle. It was organized by column based on each piece of information provided. So, one column is age, another is education, ending with the biography section which is just a short paragraph describing the person. This led to our central question of the project: “Can we use just the biography (bio) column of responses to predict the different

attributes of the person applying to the dating site?" Effectively we used the bio as our input and use different columns as our labels. Four columns labels chosen for analysis: age, education, career, and interest level of drinking (drinking socially, drinking rarely, not at all). The idea was to check if we can make accurate predictions, using NLP models on their "Biography" text.

Data Cleaning & Pre-processing: Based on true label type, NLP models framed to attain useful multinominal and binomial label predictions. Multinominal classification will be for all categories in different models, but binomial evaluation will look at if a person drinks or not at all. Label consolidation was done for each true label columns to standardize into definite groups. Age was bracketed into 6 relatively even groups and named as age_group. Same procedure repeated for job, education into job_group and education_group respectively. Even though such platforms have highly skewed representation from certain age groups, in our analysis we tried to form a dataset that had a balanced or relatively even distribution of people at least by Age_group column.

Dataset cleaning for experimentation & model-scale up: Dataset needed to be managed to set up and perform analysis. Raw dataset had 60000 lines of entries, each one being an individual applicant. Blank spaces for important places in the 4 label columns of interest and the bio were removed first. Some rows had mis formatted columns due to incorrect fillings performed by the user. All of these rows were removed. The data cleaning step implemented can be summarized as: Text consolidation on "Biography" column, Labels refined and merged, Convert drinking habits into Boolean data type, remove all wrong or missing entries in columns of interest, Regex cleaning, convert all string to lower, remove all emoticons from biography, remove stop words and tokenize the data to obtain TF-IDF and embeddings for respective models. We experimented with different dataset sizes: For development 5000 'bio' passages were used. To test the model accuracy and computational capabilities of our PC we also tried following dataset sizes: 8000, 10,000, 12,000, 15,000 and 25,000. The 12,000 was the max our system was able to handle properly and hence we settled to 10,850 refined and cleaned dataset file for the project. The dataset we obtained from Kaggle was had profiles of user till the year 2011-12.

Finalized dataset: Making these adjustments, there were a remainder of 10,850 rows cleaned dataset for experimentation. This is still very high amounts of data and would normalize a resulting prediction in our models to offer decent scale-up on developed NLP models.

For testing and training, the split ration was decided 0.8 or 80% of data being in training data, while 0.2 or 20% of data is included in the testing data.

Baseline Models:

1. **Naïve Bayes** : TF_IDF vectorizer used and then pipelined the tokens into a naïve bayes classifier model. The imported file was converted to panda data frame for our analysis. Four Naïve Bayes classifier models were built. Model performance was evaluated using set of heatmaps and accompanying F-scores.

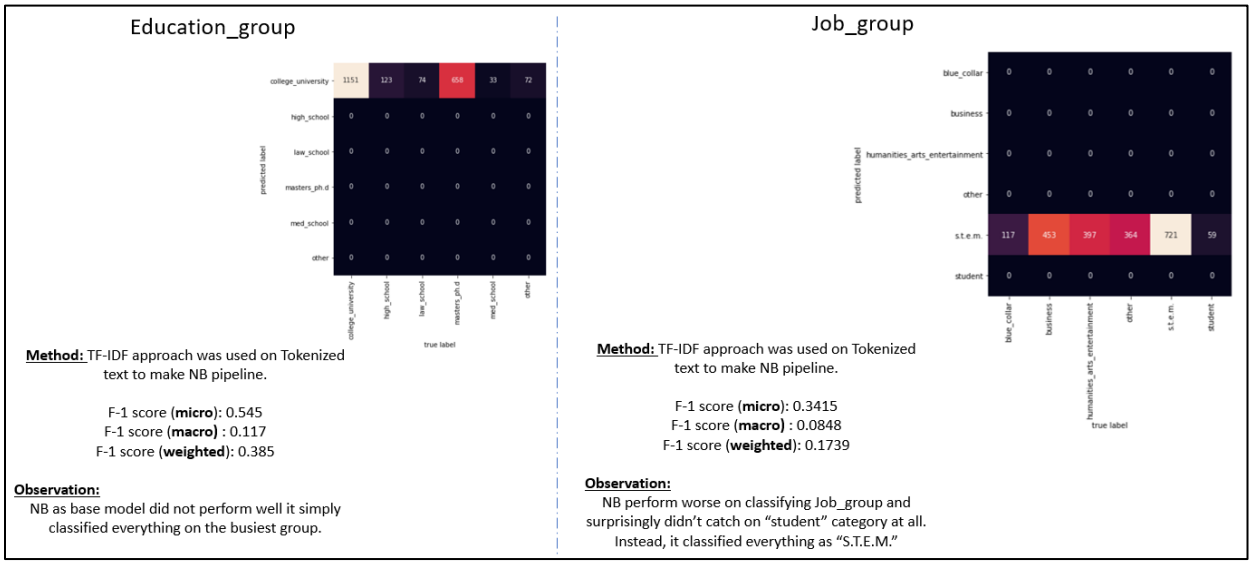


Figure: Naïve Bayes heatmap for age, job, education and drinking labels along with F-scores.

The naïve bayes model's prediction capability was not strong enough. The F-scores ranged from less than 0.001 and 0.65 with most averaging at about 0.3-0.4. The age category performed poorly

with less than 1% success rate. This could be because NB model has doesn't learn deeper syntactic context and look for direct references to age and values in the biographies. More often, people don't explicitly repeat their age value in the bio summary. Similar challenges persist when using NB to classify other categories. This clearly indicated the need to explore advanced models.

2. **Logistic Regression (LR)**: LR works well with binary data inputs as it keeps our category predictions simple. Hence, to attain binary inputs a new category based on drinking frequency label was created. As anticipated, we did see a large volume of profiles categorizing as '1'. This attribute classification is still an important problem because some non-drinkers prefer to strictly pair with similar people.

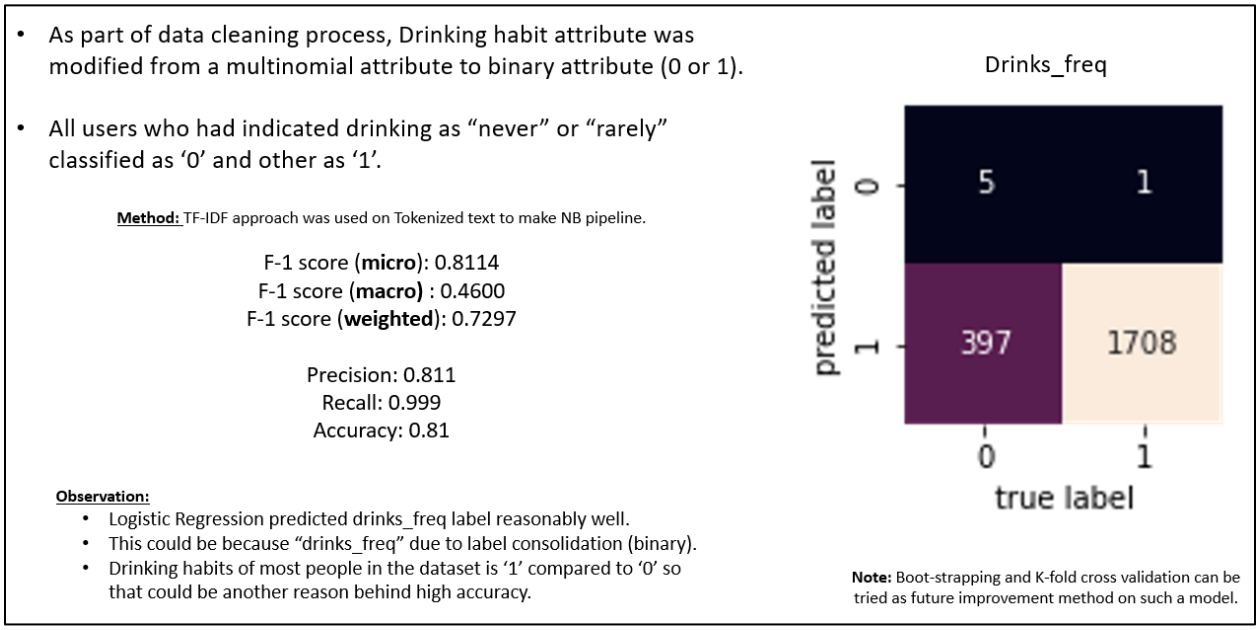


Figure: LR heatmap and F-scores for drinks_freq classification

TF_IDF vectorizer was used to define the inputs. Output was binary label type. For LR model, "1" category was overrepresented partially because this category has a high number of responses. Very few predictions placed in the "0" category. F-score was about 0.73 with a precision and recall that were both very high. This model performed better than our naïve bayes models, but the models may require a higher level of complexity to capture appropriate context. K-fold cross validation and boot strapping can be implemented to improve performance.

Advanced Model 1: CNN

Advanced NLP models implemented to test if prediction improved. The age category input was used for the test (Input was defined as TF-IDF vectorizer). The CNN model was an important

component of our work because the model can be scaled to higher architecture, so possibly one of these options can perform well in the analysis. The two options considered were one layer of 100 nodes and another with three layers, also with 100 nodes each. The result was only a marginal improvement in the F-score, moving from an F-score (weighted) of 0.4292 to 0.4374 between the one-layer setup and the three-layer setup respectively. This is less than 1% improvement in prediction, but that could be due to chance as its simulated trials for the model to test. One notable change was that the number of looped iterations fell by a wide margin. The first model with one-layer has 57 iterations to reduce loss, while the second run only required 26 runs to reduce loss to its minimum state. Heatmaps were used to visualize the results. Diagonal pattern showed that prediction labels loosely followed the true label and had some deviation

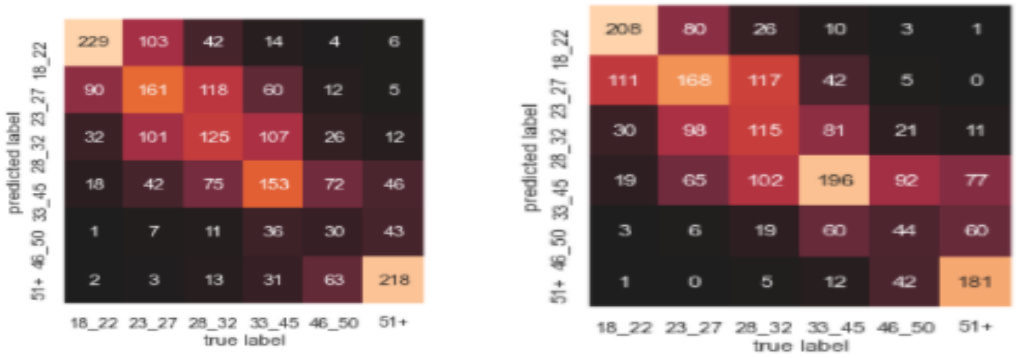


Figure : Iteration 1: CNN (left) – 1 layer 100 units and Iteration 2: CNN (right) – 3 layers, 100 units (for Age_group label)

CNN models were tried for different label categories. The second category was the education category, also with six true labels. The F-score (weighted) for the first layer was 0.5631, while the F-score (weighted) for the 3-layer was 0.5461. F-score actually fell from adding additional layers.

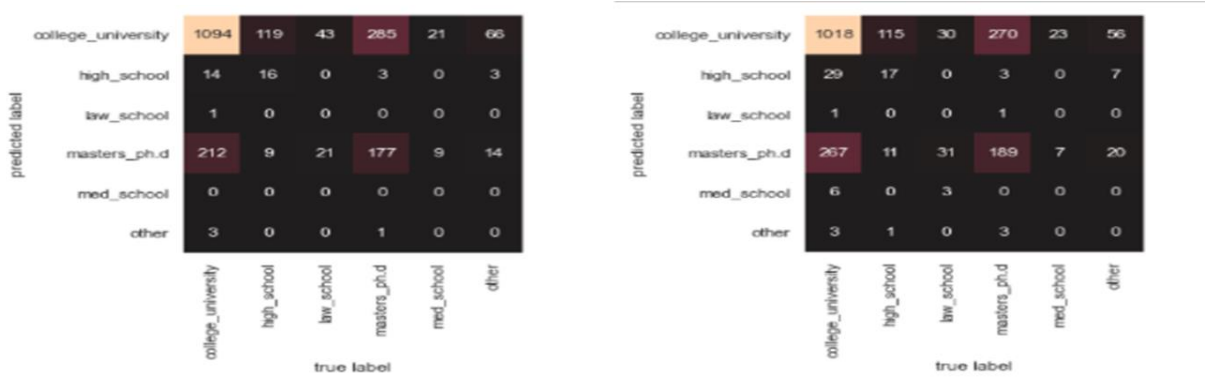
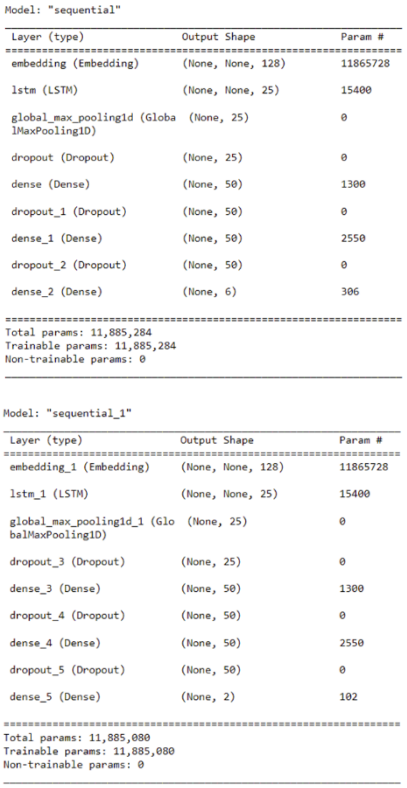


Figure : Iteration 1: CNN (left) ,1 layer 100 units and Iteration 2: CNN (right), 3 layers, 100 units (Education_group)

Both Heatmaps showed a “quadrant” pattern. Out of six labels, people from “College/University” education and “Masters/PhD” education showed heavy representation. This stands to reason because at the age and life point of people looking to find a partner, many people have this level of degree. Some of the “College/University” labels were mistaken for “Masters/PhD” and vice versa. As follow up analysis, smaller heatmap representing only these categories can be examined. Classification label outputs are greatly influenced due to unbalanced data.

Advanced Model 2: RNN

RNN structure was used next to assess category labels. Sequential model for RNN was implemented with LSTM layer. The RNN model was also adjusted to include dropouts (refer architecture diagram) in order to improve performance and reduce potential bias and variance.



```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding (Embedding)        (None, None, 128)         11865728

lstm (LSTM)                  (None, None, 25)          15400

global_max_pooling1d (Globa  (None, 25)                0
lMaxPooling1D)

dropout (Dropout)            (None, 25)                0

dense (Dense)                (None, 50)                1300

dropout_1 (Dropout)          (None, 50)                0

dense_1 (Dense)              (None, 50)                2550

dropout_2 (Dropout)          (None, 50)                0

dense_2 (Dense)              (None, 6)                 306

=================================================================
Total params: 11,885,284
Trainable params: 11,885,284
Non-trainable params: 0
_________________________________________________________________

Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_1 (Embedding)      (None, None, 128)         11865728

lstm_1 (LSTM)                (None, None, 25)          15400

global_max_pooling1d_1 (Glo  (None, 25)                0
balMaxPooling1D)

dropout_3 (Dropout)          (None, 25)                0

dense_3 (Dense)              (None, 50)                1300

dropout_4 (Dropout)          (None, 50)                0

dense_4 (Dense)              (None, 50)                2550

dropout_5 (Dropout)          (None, 50)                0

dense_5 (Dense)              (None, 2)                 102

=================================================================
Total params: 11,885,080
Trainable params: 11,885,080
Non-trainable params: 0
_________________________________________________________________
```

Figure: Sequential RNN model details (with LSTM) and dropout. Top figure is for multi-class label and bottom for binary classifier case

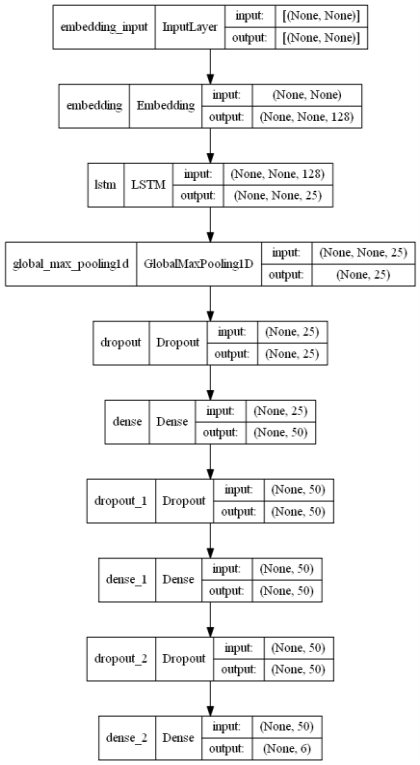


Figure: RNN architecture for multinomial class (6 labels) – education, age, and job group

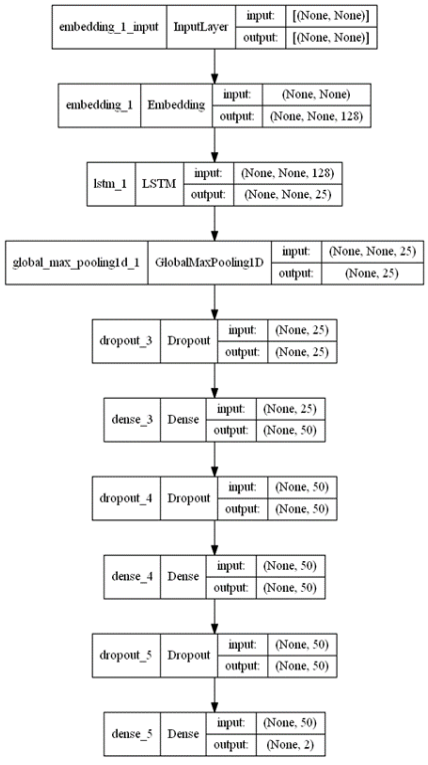
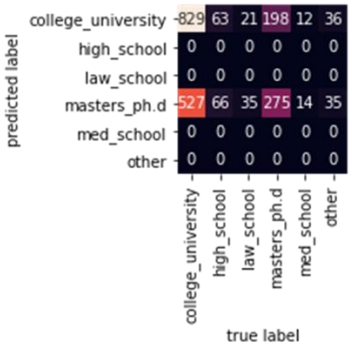
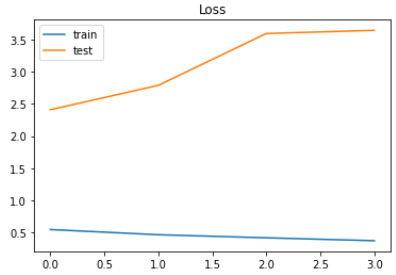
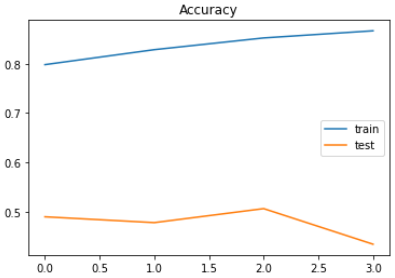
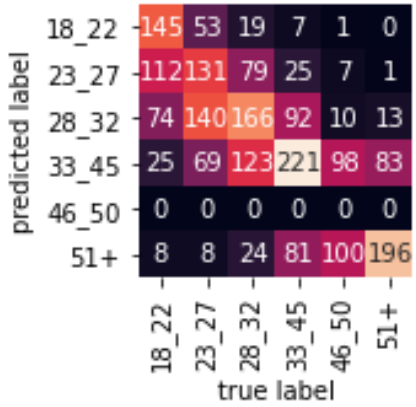
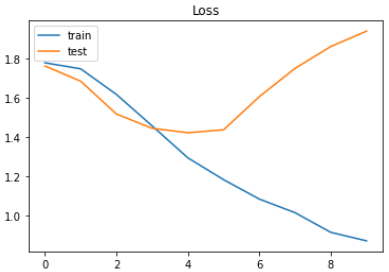
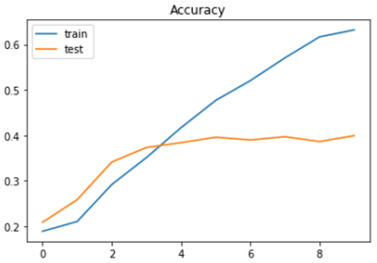
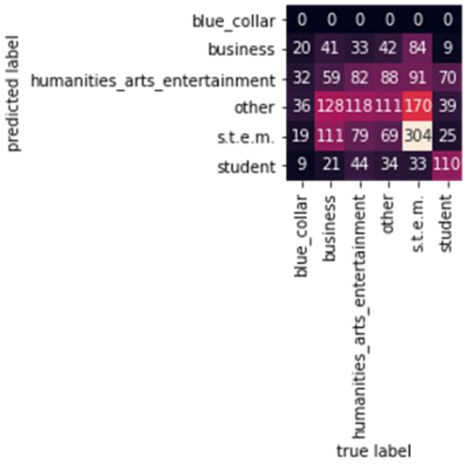
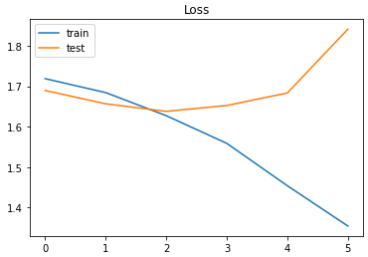
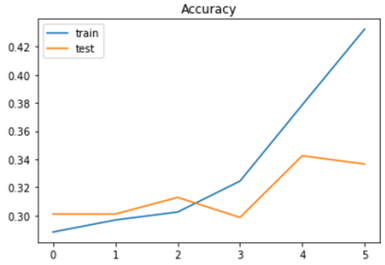
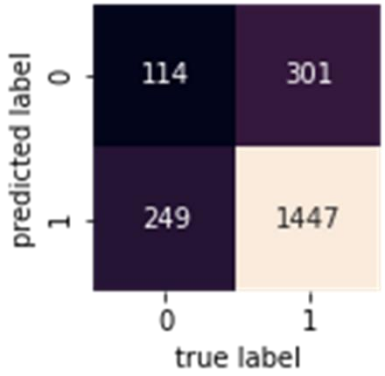
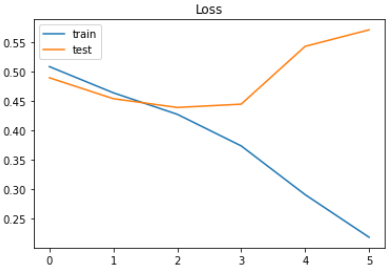
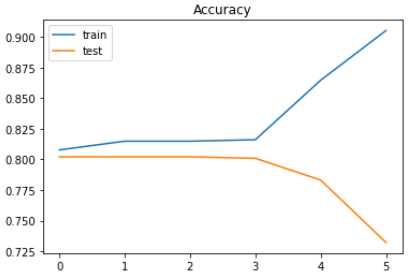


Figure: RNN architecture for binary class (2 labels) – drink_freq label

In the RNN model, embedding size taken as 128. Model is unidirectional (sequential). LSTM layer defined next , followed by Global max pool 1D layer. Dropout =0.5 is applied and for the dense

Performance Metrics for RNN model for experimented cases:

Confusion Matrix/ Heatmap	Loss Plot	Accuracy Plot	Observations for label group:
			Label class: Education_Group Overall accuracy: 0.5229 Training error (MSE): 0.0328 Validation error (MSE): 3.8925
			Label class: Age_Group Overall accuracy: 0.4069 Training error (MSE): 0.0678 Validation error (MSE): 1.5561

			Label class: Job_group Overall accuracy: 0.3069 Training error (MSE): 0.10023 Validation error (MSE): 3.0833
			Label class: drinks_freq Overall accuracy: 0.73945997 Training error (MSE): 0.05238813 Validation error (MSE): 0.260540

layers – activation functions such as “RELU” and “SOFTMAX” used for the hidden and final layer respectively. The loss function is

defined to be “categorical_crossentropy”, optimizer used = “adam” and metrics for performance defined as “accuracy”. Refer architecture diagram for details . RNN model had a low accuracy for the job-group classification (accuracy : 0.3069) and did fairly well in education_group and drink_freq classification (see table). RNN model even though didn’t have great accuracy numbers did better in generalizing class labels and understand context in people’s bio text to some extent. The loss and accuracy plots can be referred in table to understand model behavior with iterations. The loss graph has a divergence between training and testing sets at a point, with loss growing in the testing set, similarly around this point the accuracy also fell for the testing data. Thus, RNN model suffered from overfitting. RNN model also applied for Case 4 (drink_freq): binary classification type. RNN had better performance in classifying drinking category; overall accuracy was 0.7395. Even though RNN had better accuracy, the loss and accuracy plot suggested that it still had overfitting issues. Compared to the baseline models, RNN offered better classification and far more generalized outcomes. Although, noise in the dataset and diversity of writing by users does challenge the model performance especially when predicting multinomial labels.

Advanced Model 3: LSTM

Another variant of RNN was experimented as a LSTM model (using embeddings in input layer). This version had a different architecture, dropout value and used recurrent dropouts. Besides this LSTM model had similar framework and hence we can directly move to discuss model results. LSTM model implemented is Sequential and not bi-directional. LSTM model was experimented for ‘drinking’ column. Results indicated overfitting with the loss and accuracy graphs diverging for training and testing dataset. Accuracy observed to be 0.73 or 73%. The LSTM model exclusively appended to only one output label (during testing) out of 6 potential categories. This LSTM architecture showed less generalization cases and only predicted one output label for all given inputs and didn’t capture the context from specific inputs. Thus, higher accuracy wouldn’t necessarily mean that LSTM architecture is a better. Refer Accuracy and loss plots below.

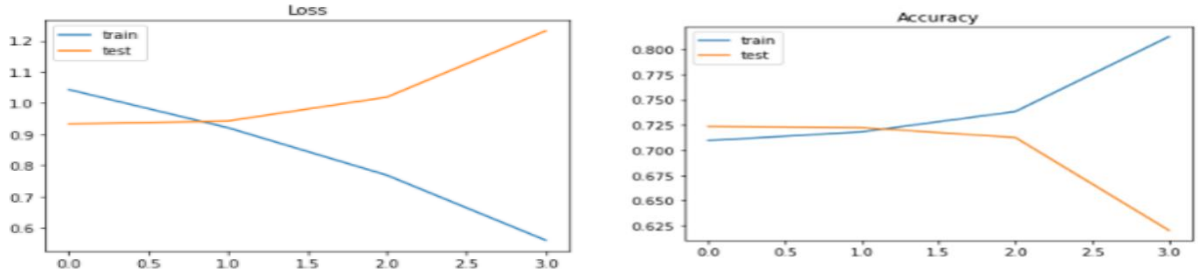


Figure: Loss and Accuracy plots for the LSTM (with recurrent dropout), implemented for drinking category

Because of the unique nature of the problem statement to predict labels from non-structured text (biography essays), the team had to experiment on multiple models to determine which model would suit best. Hence, we used RNN, CNN, and LSTM architectures in this work and tried to explore patterns from obtained results about model acceptability. For instance, RNN with dropout surprisingly had more generalized results compared to LSTM with recurrent dropout. Thus, underpinning the importance of model tuning. Different dropout value and number of layers could alter the outcome considerably.

Advanced Model 4: Distill-Bert Model

The last set of models used for prediction was our Distil-Bert. It is expected to be effective in prediction given its comprehensive examination of words in their context in ways beyond other models previously developed (within the constraints of PC capabilities). The Distil-Bert model was only applied to the drinking frequency data (binary classification) due to a limitation on resources and time. Keeping the same architecture and parameters in the code as provided to us for two categories classification (class assignment), we could use binary classifiers without extraneous issues that could occur with a multiclass classifier.

Distil-Bert model has two methods of model being run: a simpler portion (untuned) and tuned versions. Number of categories were set in the model, and we tuned on train split to 0.2/0.8. Upon implementation, loss steadily fell to about 0.485 and remained there for training. Testing loss did not move beyond 0.460. Accuracy of 0.813 observed for training, 0.83 for the testing data without much change over the iterations. Steady loss and accuracy plot indicated that the model was not learning over iterations. The accuracy is high which means that the model works well for the two categories to predict, so a binary indicator was most likely a good intuition, but we looked to see if adding complexity could improve our results.

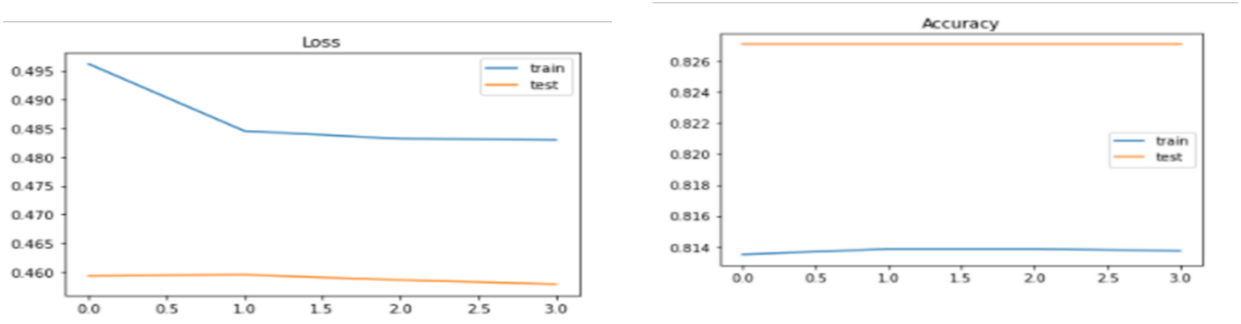


Figure: Iteration 1 of Distill-BERT (Untuned model) – Loss and Accuracy plots

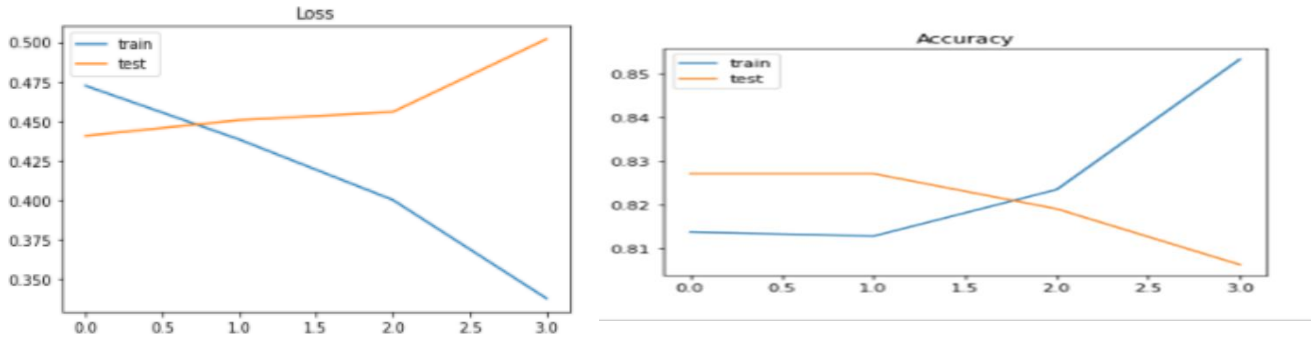


Figure: Iteration 2 of Distill-BERT (Tuned model) – Loss and Accuracy plots

The fined tuned model fluctuated in resulting loss and accuracy with testing loss increasing rapidly and the accuracy falling equally, but still around 0.47 for loss and 0.82 for accuracy. This supports the model stabilizes around this point, with the fine-tuned model having started higher in performance and then falling to a point. Because total loss and accuracy remained in the same range for both measures, there was more processing used in the fine-tuned case but without higher resulting performance. Accuracy of 80+% in predicting our labels which is very high and the best in performance of all models experimented (for binary label classification case).

In this project, we had to work with data having lot of noise and a wide array of writing styles and even foreign language, so higher accuracy from BERT is encouraging for real-world application. By scaling up the dataset size, we could potentially get even higher label prediction success rate. This would imply that one predict with high certainty from a person's bio if they drink or not and place them into suitable categories.

Advanced model 5: Clustering

As an additional objective, K-means clustering was implemented to group profiles based on biography text similarities (unsupervised ML). TF-IDF was used to vectorize values. Different clusters were tried, **K = 3, 4 and 5** to obtain scatter plots. By choosing **K =3** and **K =4** silhouette scores improved but they still lied close to 0, indicating that many labels are not fully classified and lying on borderline. Elbow method was used to determine appropriate number of clusters.

To begin with since we have six true labels for age, job, and education category. It was expected that K=6 would give six distinct clusters. But the resultant scatter plot showed that some of the groups were wrongly clustered together. Thus, centers of cluster were too close/ overlapping indicating poor clustering. We also analyzed top 10 words in each cluster. While each cluster top

words didn't yield any useful information. For K =6 case, cluster 2 had top words as: "offense, loud, wolves, especially" and cluster 4 had top words as: "distance, trust, running, swimming, trail, somebody, swim". This clearly indicated that while the model wasn't doing great in clustering profiles into 6 refined clusters but to some extent top words in specific cluster were guiding the clustering to achievable level of efficacy.

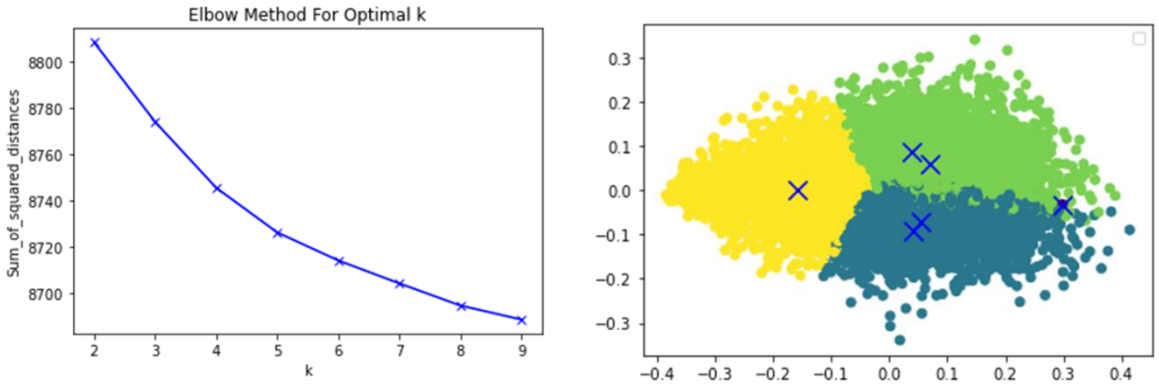


Figure: Elbow method to determine appropriate number of clusters (left) and scatter plot for (K=6) clustering case, only 3 distinct groups realized.

Clustering Performance scores:

- silhouette_score: -0.0605 (negative value indicates some bio placed in wrong cluster.
- Silhouette_score lies between -1 to +1. 0 indicating (border-line)

Estimating Clustering Model performance against true labels:

- homogeneity_score (age_group): 0.0111

Table: Comparison of models tested in the project (summary):

Label/ Category being investigated	Baseline Model & Accuracy	Advanced Model & Accuracy	Observations
Education_group	NB, 0.385	RNN, 0.5229	36% improvement in RNN
		CNN, 0.5461	3-layer, 100 units CNN model had 41.8% higher accuracy
Age_group	NB, 0.0009	RNN, 0.4069	NB model couldn't classify at all, so RNN is significant improvement
		CNN, 0.4374	3-layer, 100 units CNN architecture perform better than RNN model with dropout
Job_group	NB, 0.1739	RNN, 0.3069	76% improvement with RNN
Drinks_freq	LR, 0.81	RNN, 0.7394	RNN <LR, but it is more generalized than base LR model.
		BERT, 0.82	BERT model edged LR for accuracy

Summary of Steps Taken to overcome High Bias/ Variance issues & other challenges:

- **Bigger dataset**: For Age RNN case, model was experimented with 12000 datasets and accuracy improved from 30% to 34%. Thus, larger datasets can help reduce High Variance.
- **To mitigate Higher Variance:** Dropout and Recurrent dropout were incorporated. As shown, advanced models such as RNN, LSTM performed comparatively better due to introduction of dropout. Regularization could be tried in future as another strategy to reduce variance.
- **Changing Epoch length, batch size** : Batch sizes were changed from 32 to 50, 500 even for higher volume dataset (12000 entries). For data size of (10,000 to 15,000 entries), high batch size helps reduce computation time, but the accuracy didn't improve.
- **Bigger Architecture tried with CNN** : 3 layers (100 units each).

Conclusions & suggestions to improve results:

- **Size of experimented dataset** directly influences classification results.
- **Nature of data used (essays)** could be directly responsible for low accuracies. For example: "**Age_group**" cannot be discernably inferred from essays description unless someone explicitly mentions it out. So demarcating **b/w 23-27 and 28-32 age group** was particularly challenging even with sophisticated models.
- **Deeper Data investigation** and **pre-processing** could be necessary to understand if the sample dataset had out of English language words (foreign language bio).
- **Mathematical symbols** eliminated during text cleaning and emoticons could also have played a role. For instance: a doctor could have described their profession by an emoji, but the data cleaning removed it hence classification accuracy got impacted
- **Binary labels** classified better in both baseline and advanced ML models.
- **Poor performance from base models** such as NB, clearly indicated the importance of RNN - LSTM architecture to **capture context in long text (essays).**
- **Advanced and higher ML architecture helped classify multinomial labels** to some extent.
- In general, most of the discussed models suffered from **over-fitting.**

Learning from the project & Future Improvements:

Going forward we can view the collection of models built and their interpretation as components of a larger framework of predicting compatible people based on different attributes of their profile. In isolation a person's age or education may not be enough to action matches between users, but if a bio can predict multiple attributes and these are combined, the attributes can be weighted to create a quantified score for each user. The closer in score two people are, the more likely they should be chosen to be recommended for one another. This just requires controlling criteria, like location or preference of gender, but it is very practically feasible. Sites compete to best match users to attract paying clients and this work is foundational to support the goal of a company to build the best site possible. More work in this area can be performed to build a balanced scorecard and lead to valuable predictions just utilizing a person's bio and associated labels.

Contribution from each team member

Name: Jeff Dean

Key contributions:

Research Report:

- Researched and found appropriate academic research paper from qualified online sources
- Developed research paper review and discussed methods and how they were considered for the development of our work.
- Identified shortcomings or places we could add further in our report relative to what others have tried to establish.

Models:

- Naïve Bayes Models: Drinking, Job, Education, Age
- LSTM: Drinking
- CNN: Age, Education
- Distill-BERT: Drinking_Frequency

Name: Mohit Agarwal

Key contributions:

Data cleaning:

- Prepared original clean dataset - 'Bio' consolidation and getting rid of empty, NaN values
- Tokenization and removal of emoticons, NaN values, Stop words removal - to create input for various models created in our project
- Created an equal distribution sample file (age category) and prepared the dataset ready for all models by "age_category". This file was used for all the models.

Models:

- Naive Bayes Models : Education, age, job.
- Logistic Regression: Drinks_frequency
- RNN with LSTM (dropout): Education, age, job
- Clustering: Bio clustered unlabeled and then checked with true labels to determine clustering accuracy (K-means clustering tried) – Additional objective**

Reference:

1. Friedman, Arielle, Rushil Singh, and Ao Feng. "Dating Profile Age Analysis–a Supervised Learning Application."