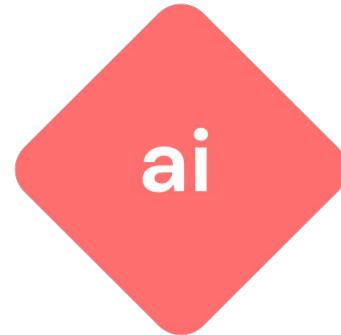




ACM Projects Team [TBD]



Colors!

here are the main colors we use at acm!

- binary blue
- big O(range)
- ctf cyan
- prototyping pink
- innovation indigo
- sentient scarlet



**Meet the
Team!**

Vincent Tu
Mentor



Sia Patodia



Aryaman Dayal



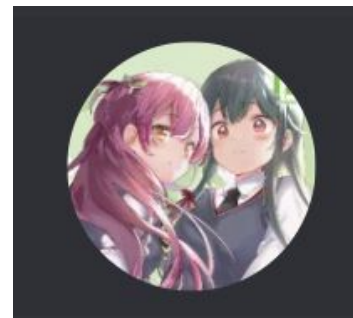
Hargen Zheng



Catherine Zhang



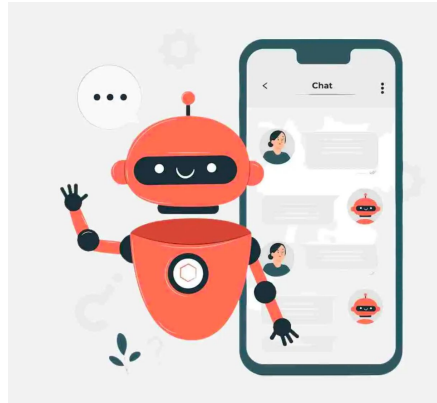
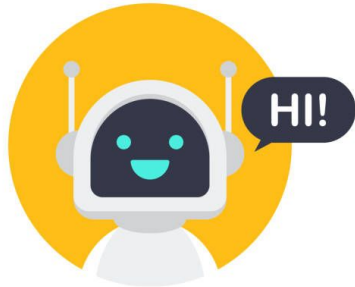
Ryan Wong



Phillip Wu

Inspiration & Background

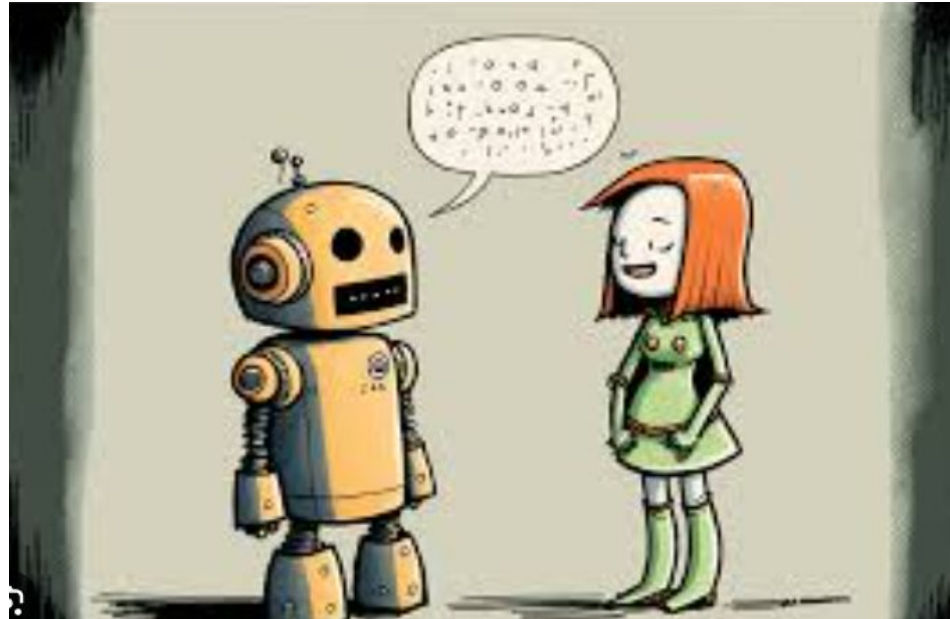
- AI Chatbot
 - Responds based on user input
- MBTI Personality Test
 - Can influence texting style



Features

- ✓ Predict MBTI personality with a decent accuracy
 - Have a chatbot talk back to user with that MBTI personality

ISTJ Responsible, logical, organized, reserved, realistic, systematic, hardworking and thorough in work, detail oriented, practical.	ISFJ Warm, conservative, gentle, responsible, pragmatic, thorough, concerned attention to detail, strong helpfulness.	INFJ Idealistic, original, insightful, dreamlike, compassionate, gentle, best harmony and cooperation, single-minded devotion.	INTJ Inventive, independent, strategic, logical, reserved, insightful, concerned that they might miss important ideas to achieve their goals.
ISTP Action-oriented, logical, analytical, spontaneous, reserved, independent, pragmatic, skilled at understanding how mechanical things work.	ISFP Calm, sensitive, nurturing, helpful, flexible, realistic, built to move a personal environment that is both beautiful and practical.	INFP Sensitive, idealistic, idealistic, sensitive, caring, loyal, future-oriented, harmony and personal growth, focus on ideas and possibilities.	INTP Intellectual, logical, creative, reserved, flexible, independent, original thinker who enjoys speculation and creative problem solving.
ESTP Outgoing, realistic, action-oriented, rational, socially spontaneous, pragmatic, problem solver, excellent negotiator.	ESFP Idealistic, enthusiastic, friendly, spontaneous, fun, fun, fun, fun, strong common sense, enjoying people in tangible ways.	ENFP Idealistic, creative, spontaneous, optimistic, imaginative, playful, strong imagination, enjoy learning new concepts, see potential in others.	ENTP Inventive, intellectual, strategic, generating, negotiator, versatile, enjoy new ideas and challenges, value innovation.
ESTJ Efficient, outgoing, logical, organized, responsible, realistic, like to run the show and get things done in an orderly fashion.	ESFJ Friendly, outgoing, idealistic, organized, respectful, practical, like to be helpful and please others, enjoying active and practical.	ENFJ Caring, enthusiastic, idealistic, organized, diplomatic, responsible, ideal communication skills, like to communicate with people.	ENTJ Strategic, logical, efficient, outgoing, ambitious, independent, effective organizer of people, solving large problems.

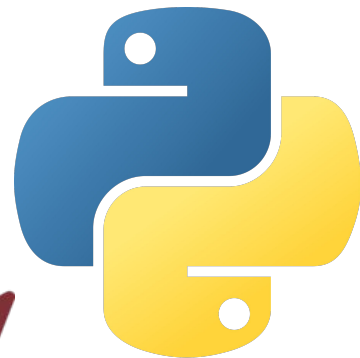


Look at our tech stack !!

 pandas



kaggle™

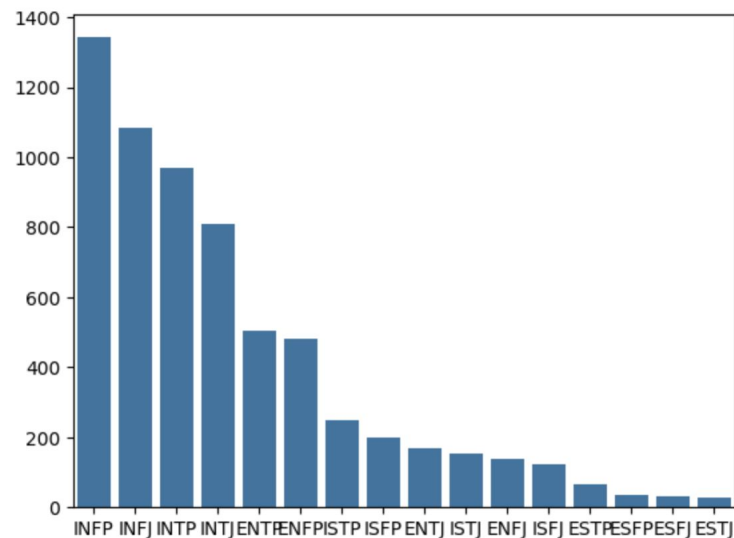


Technical Process

Dataset link: [\(MBTI\) Myers-Briggs Personality Type Dataset](#)



- EDA results:
 - Dataset has more introverts (super unbalanced)
 - Each entry has user's MBTI type and words from user's last 50 posted tweets
- Preprocessing:
 - Tokenization using BERT pre-trained tokenizer
 - Encode the MBTI type into integers 0-15
 - Only keep words between 3 and 30 characters



INFP	21%
INFJ	17%
Other (5373)	62%

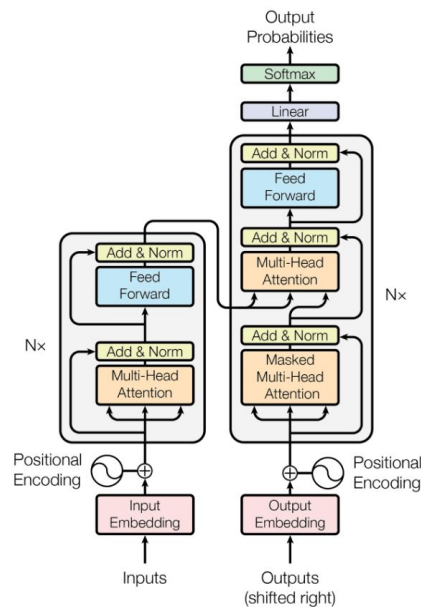
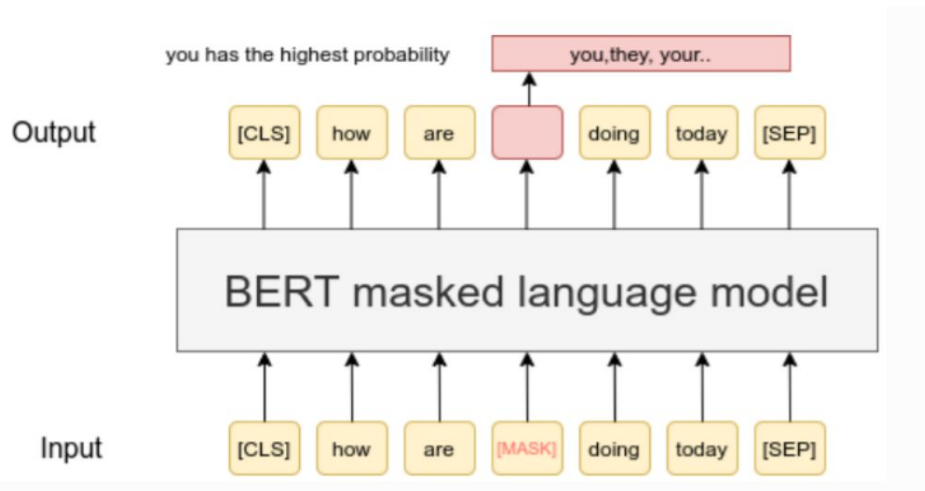
8675
unique values

Here are the original notes vincent wrote down

- Talk about dataset (e.g. dataset size, modality, feature engineering, cleaning, EDA, insights, conclusions, motivation behind this dataset)
- Talk about preprocessing (tokenization, stemming, lemmatizing, etc)
- Talk about the model (BERT other neural language models -> talk more about how these models are made and how they work; LSTM -> long-short-term memory loosely inspired by human brain)
- Talk about the training (and finetuning), discuss training configuration and set up; if you CV, then you can also talk about that
- Talk about your experiments (what worked and what didn't work; possibly hypotheses for why things didn't work or why things worked)
- Talk inference (how do we run the model)?

Technical Process (cont.)

- Model: **BERT (Bidirectional Encoder Representations from Transformers)**
 - Unique since it is considered a bidirectional model (captures context well)
 - Uses a transformer architecture
 - Combined predicting words along with predicting if sentences belonged to each other to create the BERT model



Technical Process (cont.)

Training the BERT Model

- Transfer learning (cold start issue due to size of parameters, limited computation power). We used Colab T4 GPU to train the model.
- Embed each text input into a vector with maximum length of 256 (padding).
- Shuffle the data and use 80/20 train-validation split.
- Batch size of 16 (drop the remainder). This results in 433 batches in each epoch.
- Add a fully connected layer with 512 hidden units, with ReLU activation function.
- Use a softmax layer with 16 output units to represent the probabilities of each MBTI type, given the input text corpus.
- Used the built-in Adam Optimizer with decay and Cross Entropy Loss function.



Technical Process (cont.)

BERT Model Summary

Layer (type)	Output Shape	Param #	Connected to
input_ids (InputLayer)	[(None, 256)]	0	[]
attention_mask (InputLayer)	[(None, 256)]	0	[]
bert (TFBertMainLayer)	TFBaseModelOutputWithPoolingAndCrossAttentions(last_hidden_state=(None, 256, 768), pooler_output=(None, 768), past_key_values=None, hidden_states=None, attentions=None, cross_attentions=None)	108310272	['input_ids[0][0]', 'attention_mask[0][0]']
intermediate_layer (Dense)	(None, 512)	393728	['bert[0][1]']
output_layer (Dense)	(None, 16)	8208	['intermediate_layer[0][0]']
=====			
Total params: 108712208 (414.70 MB)			
Trainable params: 108712208 (414.70 MB)			
Non-trainable params: 0 (0.00 Byte)			

Technical Process (cont.)

Experiments

- Though $1e-5$ is the recommended learning rate for most of the transformer models, it was super slow to train for us. As a rookie I changed to $1e-4$ and we were overshooting – after 1 epoch, the accuracy went from 35% down to 18% and it kept decreasing.
- Used $5e-5$ as learning rate and decay of $1e-6$ – accuracy keeps increasing all the way to ~95% on the training set. This is way better than the original Softmax regression model, which obtained a 84% on the training set.
- Could've tune the maximum size of word embeddings and the size of hidden layers

Technical Process (cont.)

Eventual Model Performance

```
Epoch 1/2
433/433 [=====] - 423s 976ms/step - loss: 0.1813 - accuracy: 0.9433 - val_loss: 0.0714 - val_accuracy: 0.9776
Epoch 2/2
433/433 [=====] - 414s 956ms/step - loss: 0.1713 - accuracy: 0.9479 - val_loss: 0.0697 - val_accuracy: 0.9794
```



Technical Process (cont.)

Inference

- We clean the text corpus in the same way as we did for the training and validation examples.
- Then, we used the BERT tokenizer to embed the input text corpus.
- The feature vector of the input text is then fed into our transformer model, which gives us 16 probabilities corresponding to the likelihood of each personality type.
- We then extract the top three most likely personalities and display to the user.

Technical Process (cont.)

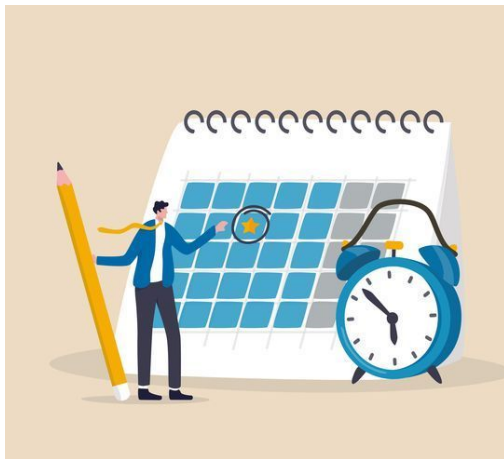
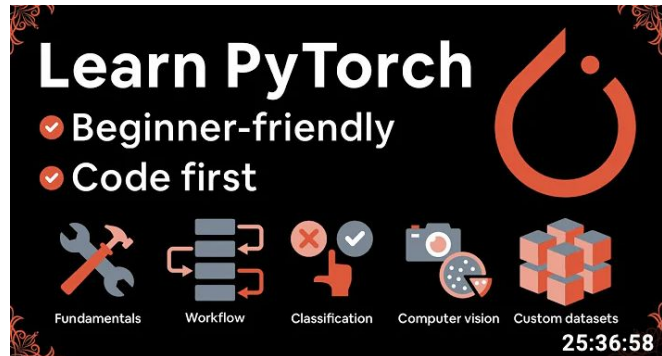
Streamlit App

- We created a Streamlit application that shows the distribution of MBTI personality types in our dataset.
- User could input paragraphs of text and the app will predict their top three likely MBTI personalities using our model.
- WHY TOP 3, not just 1???
- After the prediction result pops up, there will be additional MBTI information of the three likely personalities, so the user can learn more about their potential MBTI type.



Challenges

- Meeting scheduling conflicts
- Balancing classwork and project
- Lack of experience (PyTorch, Model Choice, Training)
- Model overfitting



Reflection

- Smaller group sizes may make it easier to meet
- Self-guided learning is difficult
- Talk about how RoBERTa could potentially work better (as it was trained on a larger text corpus)

Where do we go from here?

- Create mobile application for interacting with chatbot
- Learn more about NLP and deep learning

Demo



Questions?

Thank you!

- Mentor: Vincent Tu [LinkedIn](#) | [GitHub](#)
- Catherine Zhang [LinkedIn](#) | [GitHub](#)
- Aryaman Dayal [LinkedIn](#) | [GitHub](#)
- Hargen Zheng [LinkedIn](#) | [GitHub](#)
- Sia Patodia [LinkedIn](#) | [GitHub](#)
- Phillip Wu [GitHub](#)
- Ryan Wong [LinkedIn](#) | [GitHub](#)

