GENERATIVE CHATBOTS

ALEX FIOTO

HELLO!

I am Alex Fioto

Let’s talk about chatbots!

DATA SCIENCE TASK:

BUILD A REALISTIC, ENTERTAINING AND FUNCTIONAL GENERATIVE CHATBOT

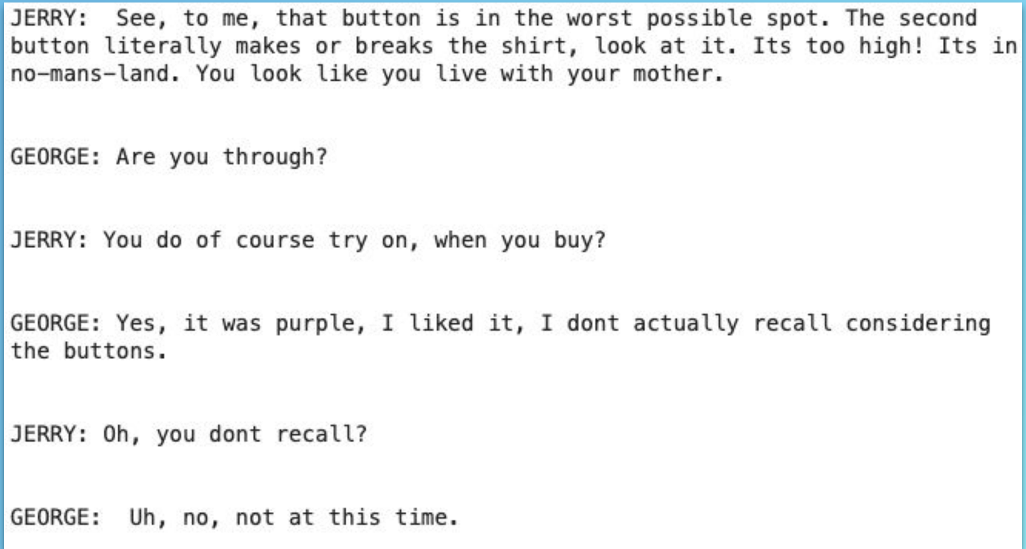
Kaggle Link

GPT2 / AITEXTGEN

DATA PREPROCESSING

Cleaning and preparation

- Drop NaNs
- Remove stage directions
- Add character names to dialogue
- Convert to txt file



```
JERRY:  See, to me, that button is in the worst possible spot. The second
button literally makes or breaks the shirt, look at it. Its too high! Its in
no-mans-land. You look like you live with your mother.


GEORGE: Are you through?


JERRY: You do of course try on, when you buy?


GEORGE: Yes, it was purple, I liked it, I dont actually recall considering
the buttons.


JERRY: Oh, you dont recall?


GEORGE:  Uh, no, not at this time.
```

MODEL TRAINING

Training Preparation

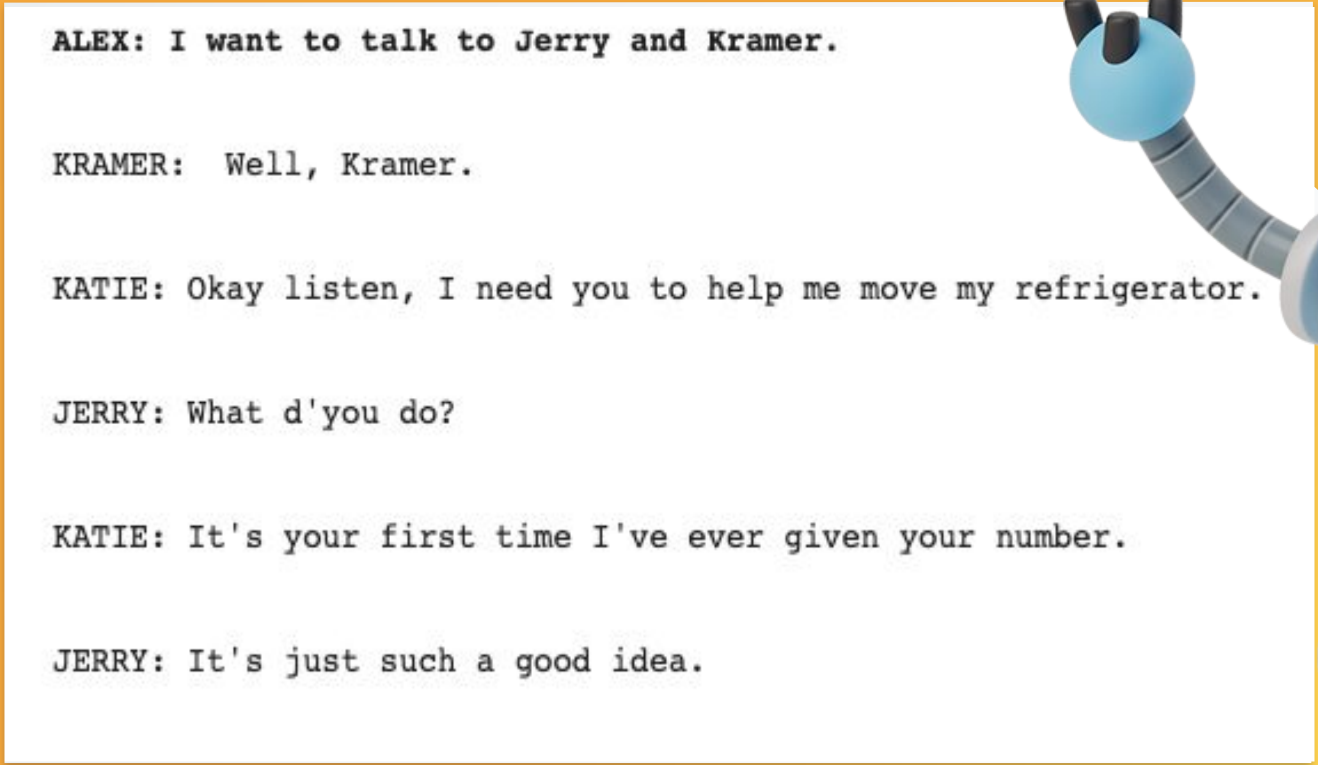
- Mount Google Drive
- Train tokenizer on txt file
- Build GTP2 Configuration

Parameters

- max_length=100
- dropout=0.0
- n_layers=8

Training

- epochs=5000
- batch_size=100
- Learning rate=.0004



ALEX: I want to talk to Jerry and Kramer.

KRAMER: Well, Kramer.

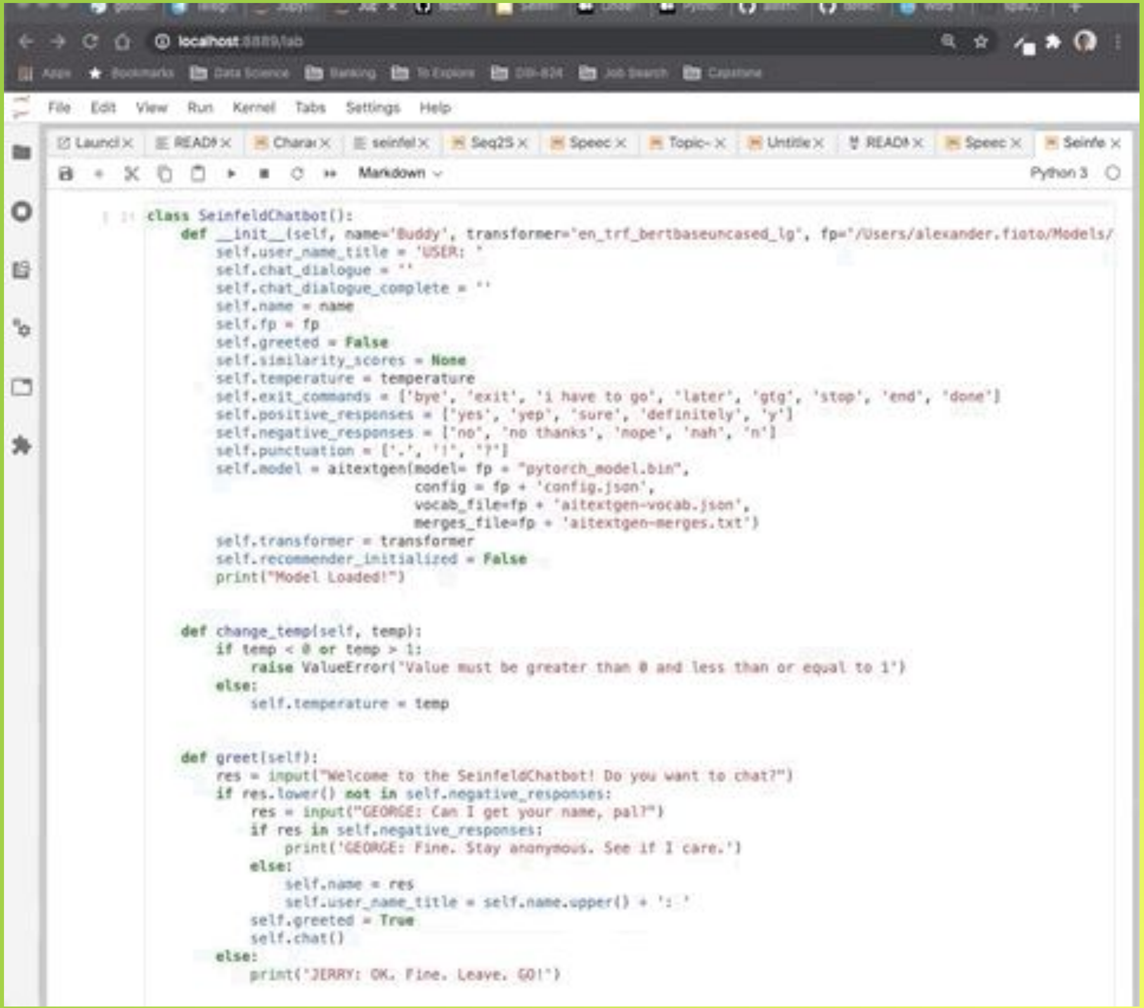
KATIE: Okay listen, I need you to help me move my refrigerator.

JERRY: What d'you do?

KATIE: It's your first time I've ever given your number.

JERRY: It's just such a good idea.

CHATBOT CLASS



```python
class SeinfeldChatbot():
    def __init__(self, name='Buddy', transformer='en_trf_bertbaseuncased_lg', fp='/Users/alexander.fioto/Models/
        self.user_name_title = 'USER: '
        self.chat_dialogue = ''
        self.chat_dialogue_complete = ''
        self.name = name
        self.fp = fp
        self.greeted = False
        self.similarity_scores = None
        self.temperature = temperature
        self.exit_commands = ['bye', 'exit', 'i have to go', 'later', 'gtg', 'stop', 'end', 'done']
        self.positive_responses = ['yes', 'yep', 'sure', 'definitely', 'y']
        self.negative_responses = ['no', 'no thanks', 'nope', 'nah', 'n']
        self.punctuation = ['.', '!', '?']
        self.model = aitextgen(model= fp + "pytorch_model.bin",
                               config = fp + 'config.json',
                               vocab_file=fp + 'aitextgen-vocab.json',
                               merges_file=fp + 'aitextgen-merges.txt')
        self.transformer = transformer
        self.recommender_initialized = False
        print("Model Loaded!")


    def change_temp(self, temp):
        if temp < 0 or temp > 1:
            raise ValueError("Value must be greater than 0 and less than or equal to 1')
        else:
            self.temperature = temp


    def greet(self):
        res = input("Welcome to the SeinfeldChatbot! Do you want to chat?")
        if res.lower() not in self.negative_responses:
            res = input("GEORGE: Can I get your name, pal?")
            if res in self.negative_responses:
                print('GEORGE: Fine. Stay anonymous. See if I care.')
            else:
                self.name = res
                self.user_name_title = self.name.upper() + ': '
            self.greeted = True
            self.chat()
        else:
            print('JERRY: OK. Fine. Leave. GO!")
```

FUNCTIONALITY

- Control Flow
- Episode Recommender
- Character Prediction
- Change “Creativity”

SPACY RECOMMENDER

RECOMMENDER SPECS

- BERT Word Embeddings from Hugging Face Transformers
- Dictionary of episode keys and spaCy objects
- Cosine similarity of chat dialogue and episodes

THE GOOD

- Simple
- Fast
- Effective

THE BAD

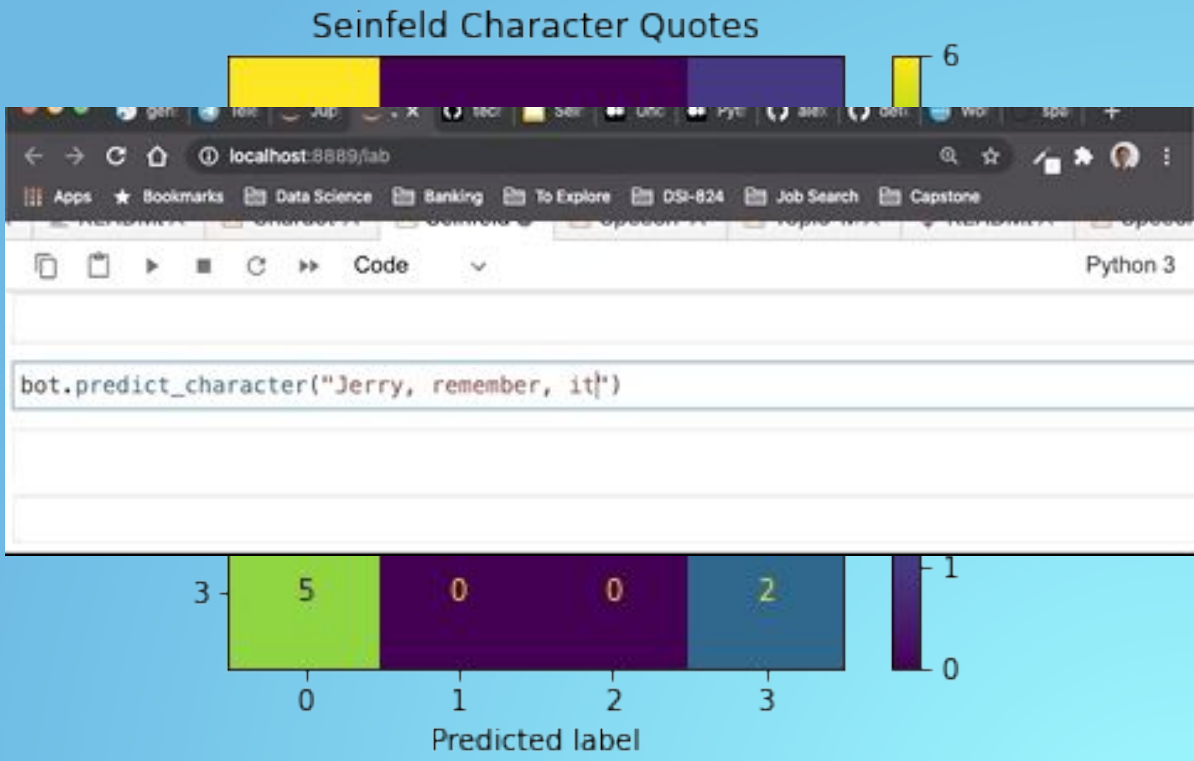
- Subjective
- Could be faster
- Large

IMDbPY

CHARACTER PREDICTION

PREDICTION MODELING

- Null Model - Jerry 37%
- TFIDF
- Support Vector Classifier
- Logistic Regression
- Better on Quotes



```
bot.predict_character("Jerry, remember, it")
```

DEMO TIME

MOVING FORWARD

Reduce Footprint

- Text Summarization
- Topic Modeling

Deployment/GUI

- Flask
- Heroku

Feedback

- Recommender
- Star System

QUESTIONS?

Thank you!

FIRST ITERATIONS

- Seq2Seq
- Chatterbot

GITHUB REPO