FinTech Bootcamp Project 2

Back-End Algorithmic Trading Model

Project Outline

Classification and regression model comparison
Use FinTA package for technical indicators
Pull in OHLCV data from Santiment for BTC

Notebook Overviews

BTC Golden Cross Model:

- Santiment BTC data time frame: 12/01/2018 - 12/01/2021 (3 years)
- FinTA 50 SMA x 200 SMA = Golden Cross
- 4 entries
- 3 exits
- Split data into training and test datasets
- Standardized data using StandardScaler
- Training data Support Vector Machine model accuracy score = 60%
- Testing data Support Vector Machine model accuracy score = **53%**
    - Trading algorithm returns **outperforms** actual returns
- Training data Logistic Regression model accuracy score = 54%
- Testing data Logistic Regression model accuracy score = **48%**
- Testing data model comparative analysis:
    - Logistic Regression model accuracy score = **48%**
    - Support Vector Machine model accuracy score = **53%**

ETH Golden Cross Model::

- Santiment ETH data time frame: 12/01/2018 - 12/01/2021 (3 years)
- FinTA 50 SMA x 200 SMA = Golden Cross
- 4 entries
- 3 exits
- Split data into training and test datasets
- Standardized data using StandardScaler
- Training data Support Vector Machine model accuracy score = 58%
- Testing data Support Vector Machine model accuracy score = **54%**
    - Trading algorithm returns **underperforms** actual returns
- Training data Logistic Regression model accuracy score = 55%
- Testing data Logistic Regression model accuracy score = **45%**
- Testing data model comparative analysis:
    - Logistic Regression model accuracy score = **45%**
    - Support Vector Machine model accuracy score = **54%**

Team Roles
Charles Panagopoulos - Project Manager
Ian Melhorn - Quant Developer
Will Pape - Quant Developer
Samirah Djachechi - AWS Bot Strategist
Benjamin Boule - AWS Bot Strategist
Eugenio Ngondji - Lead Developer
Alex Toenshoff - Machine Learning Developer

Strategy
Evaluate Risk Profile to make investment recommendations based on backtesting/sentiment. Furthermore, provide recommendations based on the users available funding and risk tolerance

Technical Indicators For Consideration
3 different trading models for a customer to choose from on a website or query from the bot

- Bollinger Bands 'BBANDS'
- Simple Moving Average 'SMA' or Exponential Moving Average 'EMA'
    - Short and long window crossover
- Relative Strength Index 'RSI'
- SuperTrend
- MACD
- CCI
- Ichimoku Cloud
- Parabolic SAR

Investment Horizon
Input field/function
Initial Investment

Action Items

1. Start with one asset to model
2. Build and test the model using one asset
3. Once proven, update model to reflect multiple assets

Helpful Tools
Create functions for everyone to call in individual jupyter files to maintain code consistency

Bot configuration

- Bot name: Trading_Bot
- Output voice: Salli
- Session timeout: 5 minutes
- Sentiment analysis: No
- COPPA: No
- Advanced options: No
- *Leave default values for all other options.*

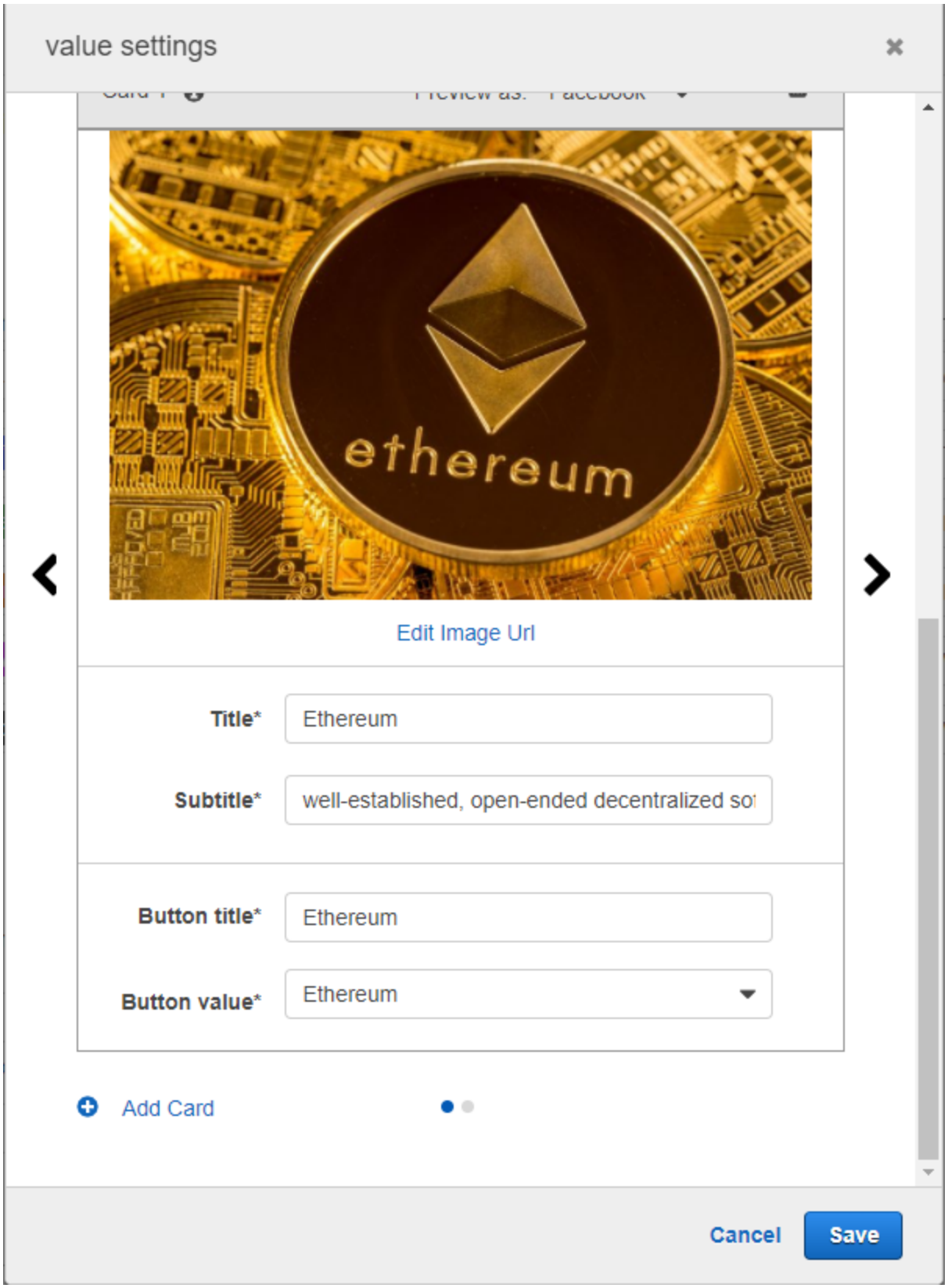
Create the Cryptotrader intent, and configure some sample utterances as follows (you can add more utterances as you wish):

- I want to invest in crypto
- I'm {age} and I would like to do day trade
- I'm {age} and I want to invest for my retirement
- I want the best crypto currency to invest in
- I do not know what crypto currency to invest in
- I want to become rich with crypto
- I would like to grow my crypto portfolio
- I want a prediction of {value} performance

This bot will use five slots, three using built-in types and one custom slot named riskLevel. Define the three initial slots as follows:

Name	Slot type	Prompt
firstName	AMAZON.US_FIRST_NAME	Thank you for trusting me to help, could you please give me your name? And email address ?
age	AMAZON.NUMBER	Hello {firstName}, how old are you?
InvestmentAmount	AMAZON.NUMBER	How much do you want to invest?
value	Cryptocurrency	Would you like to invest in Bitcoin or Ethereum?
decision	Amazon.Yesintent	Would you like to purchase this model?

value : ETC image - https://images.mktw.net/im-336321?width=700&height=467



BTC image- https://cdn.pixabay.com/photo/2017/01/25/12/31/bitcoin-2007769__480.jpg

value settings

Edit Image Url

Title* Bitcoin

Subtitle* enables a new payment system and a complet

Button title* Bitcoin

Button value* Bitcoin

Add Card

Cancel Save

Slots

Priority	Required	Name	Slot type	Version	Prompt	Settings
		e.g. Location	e.g. AMAZON.US_CITY		e.g. What city?	
1.	✓	firstName	AMAZON.US_FIRST_NAME	Built-in	Thank you for trusting me to help, could you pleas	
2.	✓	age	AMAZON.NUMBER	Built-in	Hello {firstName}, how old are you?	
3.	✓	InvestmentAmount	AMAZON.NUMBER	Built-in	How much do you want to invest?	
4.	✓	value	Cryptocurrency	1	Would you like to invest in Bitcoin or Ethereum?	
5.	✓	decision	YesIntent	1	Would you like to purchase this model?	

Test bot (Latest)

Ready. Build complete.

Hello jaja, how old are you?

5

How much do you want to invest?

5

Would you like to invest in Bitcoin or Ethereum?

Clear chat history

Chat with your bot...

Inspect response

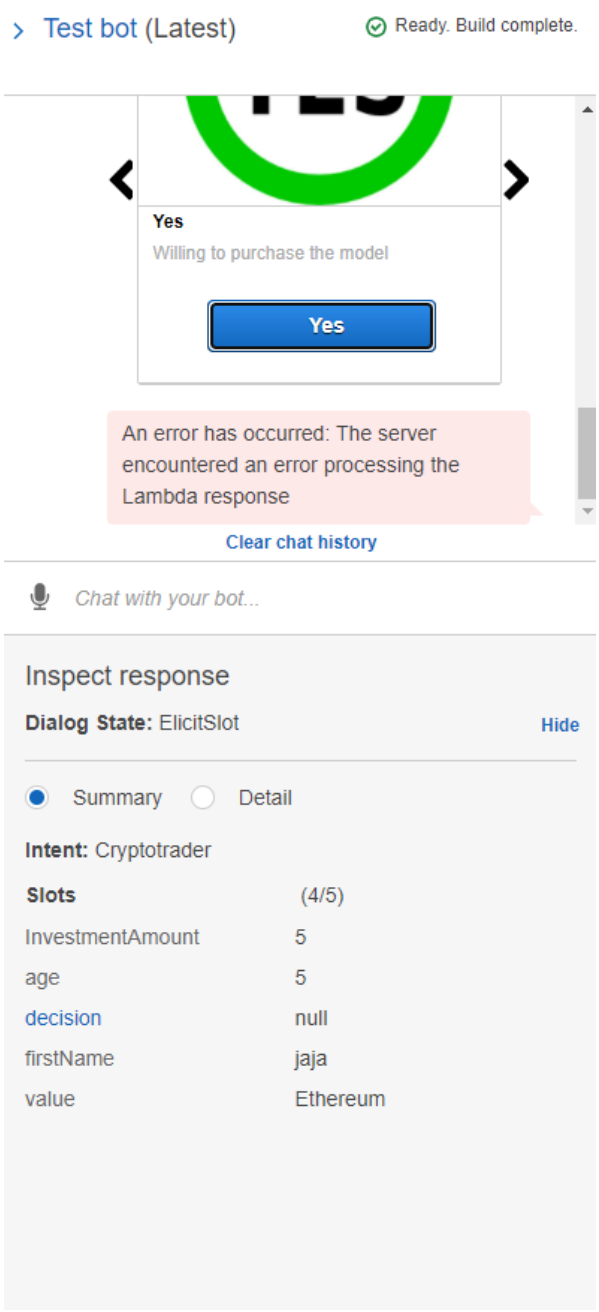
Dialog State: ElicitSlot Hide

Summary Detail

Intent: Cryptotrader

Slots	(4/5)
InvestmentAmount	5
age	5
decision	null
firstName	jaja
value	Ethereum

Error:

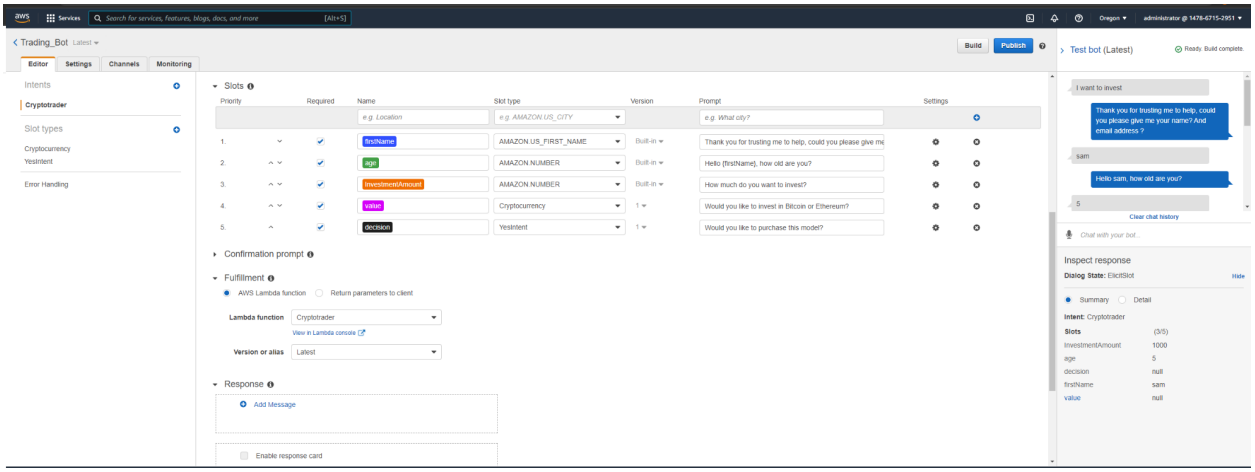

Test bot (Latest)
Ready. Build complete.
Yes
Willing to purchase the model
Yes
An error has occurred: The server encountered an error processing the Lambda response
Clear chat history
Chat with your bot...
Inspect response
Dialog State: ElicitSlot
Hide
Summary
Detail
Intent: Cryptotrader
Slots (4/5)
InvestmentAmount 5
age 5
decision null
firstName jaja
value Ethereum

Output after selection of a value : (Brief description of code and its purpose)

```
    # Get the decision

    ### Decision code starts here###
    if value == 'Yes':
        recommendation = 'Send $200 to this crypto wallet address;
1F1tAaz5x1HUXrCNLbtMDqcw6o5GNn4xqX . After the transaction is confirmed, a
copy of the Gross support vector machine model will be sent to  email
address provide above.'
    if value == 'No':
        recommendation = ''

     # Return a message with the initial recommendation based on the
value.
    return close(
        intent_request["sessionAttributes"],
        "Fulfilled",
        {
            "contentType": "PlainText",
            "content": """{} Okay;
             {}
            """.format(
                first_name, recommendation
            ),
        },
    )
```


Trading_Bot
Editor
Settings
Channels
Monitoring
Build
Publish
Intents
Cryptotrader
Slot types
Cryptocurrency
YesIntent
Error Handling
Slots
Priority
Required
Name
Slot type
Version
Prompt
Settings
1. firstName AMAZON.US_FIRST_NAME Built-in Thank you for trusting me to help, could you please give me
2. age AMAZON.NUMBER Built-in Hello {firstName}, how old are you?
3. investmentAmount AMAZON.NUMBER Built-in How much do you want to invest?
4. value Cryptocurrency 1 Would you like to invest in Bitcoin or Ethereum?
5. decision YesIntent 1 Would you like to purchase this model?
Confirmation prompt
Fulfillment
AWS Lambda function
Return parameters to client
Lambda function
Cryptotrader
View in Lambda console
Version or alias
Latest
Response
Add Message
Enable response card
Test bot (Latest)
Ready. Build complete.
I want to invest
Thank you for trusting me to help, could you please give me your name? And email address ?
sam
Hello sam, how old are you?
5
Clear chat history
Chat with your bot...
Inspect response
Dialog State: ElicitSlot
Hide
Summary
Detail
Intent: Cryptotrader
Slots (3/5)
investmentAmount 1000
age 5
decision null
firstName sam
value null

decision settings

Card 1 Preview as: Facebook

Edit Image Url

Title* Yes

Subtitle* Willing to purchase the model

Button title* Yes

Button value* Yes

Add Card

Cancel Save

decision settings

Card 2 Preview as: Facebook

Edit Image Url

Title* No

Subtitle* You are not willing to purchase this model

Button title* No

Button value* No

Add Card

Cancel Save

Technical Requirements

The technical requirements for Project 2 are as follows.

- Create a Jupyter Notebook, Google Colab Notebook, or Amazon SageMaker Notebook to prepare a training and testing dataset.
- Optionally, apply a dimensionality reduction technique to reduce the input features, or perform feature engineering to generate new features to train the model.
- Create one or more machine learning models.
- Fit the model(s) to the training data.
- Evaluate the trained model(s) using testing data. Include any calculations, metrics, or visualizations needed to evaluate the performance.
- Show the predictions using a sample of new data. Compare the predictions if more than one model is used.
- Save PNG images of your visualizations to distribute to the class and instructional team and for inclusion in your presentation and your repo's README.md file.
- Use one new machine learning library, machine learning model, or evaluation metric that hasn't been covered in class.
- Create a README.md in your repo with a write-up summarizing your project. Be sure to include any usage instructions to set up and use the model.

Presentation Guidelines

You are free to structure your presentations to your liking, but students tend to have success with the following format.

- Title Slide
    - Include the name of the project and group members.

- Motivation & Summary Slide
    - Define the core message or hypothesis of your project.
- Model Summary
    - Elaborate on the predictive model used, describing why it was the best choice for the data.
- Data Cleanup & Model Training
    - Describe the exploration and cleanup process.
    - Discuss any problems that arose with preparing the data or training the model that you didn't anticipate.
    - Discuss the overall training process and highlight anything of interest with the training process: Cloud resources used, training time required, issues with training.
- Model Evaluation
    - Discuss the techniques you used to evaluate the model performance.
- Discussion
    - Discuss your findings. Was the model sufficient for the predictive task? If not, why not? What inferences or general conclusions can you draw from your model performance?
- Postmortem
    - Discuss any difficulties that arose, and how you dealt with them.
    - Discuss any additional questions or problems that came up but you didn't have time to answer: What would you research next if you had two more weeks?
- Questions
    - Open-floor Q&A with the audience.

Presentation Requirements

The presentation requirements for the project are as follows.

Your presentation must:

- Be at least 8 to 10 minutes long (check with the instructor for the official presentation time).
- Describe the core message or hypothesis for your project.
- Describe the predictive model chose and why this model was chosen.
- Describe the data preparation and model training process.
- Describe the techniques used to evaluate the model performance.
- Summarize your conclusions and predictions. This should include a numerical summary (what data your model yielded), as well as visualizations of that summary (plots of the final model evaluation and predictions).
- Discuss the implications of your findings. This is where you get to have an open-ended discussion about what your findings mean.