AI Job Posting, draft 1.

Mozaic is an AI-powered DeFi startup that helps people efficiently make smart and safe decisions on their crypto activities. Unlike others, we envision tapping into the cutting-edge Machine Learning technology to guide our operating machines to learn quantitatively the underlying process of crypto markets.
Recently we successfully demonstrated omnichain DeFi aggregations that showed a 9% competitive edge in profitability by optimizing staking/lending strategy. Based on this exciting initial momentum, we are going to thrust our ML division with the highest priority to even get higher profitability.

You will be in charge of

- Developing your own ML models that predict the optimal staking portfolio at intervals,
- Or, collaborate with other data scientists to develop ML models.

You will be supported by

- the data engineering team

You will be supervised by

- the CEO, who has extensive experience in trading and price prediction.

Your benefits will include:

- Decent development environment, including almost unlimited use of GCP,
  True creative, inspiring workplace, working together with ambitious CEO and data scientists,
- A seamless market data collecting tool that will completely free you from missing, upside-down, or slipped price data points, in a fastest manner,
- A most flexible trading bot framework that you can configure with your ML models,
- Access to the latest blockchain and DeFi techniques, which is emerging as the next AI arena with freely available big data and huge asset circulation.

You are expected to:

- Be familiar with minimal, yet deep fundamentals of NLP and generative modeling,
- Be a creative data scientist who can handle ML tools on their own,
- Not be an if-then ML engineer or copy-and-paste programmer,

Suggested terms and conditions:

10k

You will be given technical interviews

- **with the focus on data science and time series/sequence modeling**, including but not limited to:

Basic ML skills for time series/sequential modeling

Backpropagation

- Implement the XOR gate using backpropagation, without using existing gradient tools. The number of layers/neurons should not be hard-coded.
- http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture4.pdf

Losses

- https://keras.io/api/losses/
- https://www.tensorflow.org/api_docs/python/tf/keras/losses

Optimizers

- https://keras.io/api/optimizers/
- https://www.tensorflow.org/api_docs/python/tf/keras/optimizers

Timeseries

- https://stanford.edu/~shervine/teaching/cs-230/
- cheatsheet-recurrent-neural-networks
- Implement the LSTM model with python.
- https://keras.io/examples/timeseries/timeseries_classification_transformer/

- https://keras.io/examples/timeseries/timeseries_anomaly_detection/
- https://keras.io/examples/timeseries/timeseries_weather_forecasting/

Normalization

- https://arxiv.org/pdf/1502.03167.pdf

Regularization

- https://cedar.buffalo.edu/~srihari/CSE574/Chap5/Chap5.5-Regularization.pdf
- https://www.cs.toronto.edu/~rsalakhu/papers/srivastava14a.pdf

Word embeddings

- http://home.ustc.edu.cn/~yaoyaq/Signals/Word_Embedding.pdf
- https://www.tensorflow.org/text/guide/word_embeddings
- https://www.tensorflow.org/tutorials/text/word2vec
- https://www.tensorflow.org/text/guide/word_embeddings

Attention/Transformer

- https://arxiv.org/pdf/1706.03762.pdf
- https://colab.research.google.com/github/dlmacedo/starter-academic/blob/master/
- content/courses/deeplearning/notebooks/tensorflow/transformer.ipynb
- https://keras.io/examples/nlp/neural_machine_translation_with_transformer/

BERT

- https://arxiv.org/pdf/1810.04805.pdf
- https://www.tensorflow.org/text/guide/bert_preprocessing_guide
- https://www.tensorflow.org/tfmodels/nlp
- https://www.tensorflow.org/text/guide/bert_preprocessing_guide
- https://keras.io/examples/nlp/text_extraction_with_bert/

Autoencoders

- https://cedar.buffalo.edu/~srihari/CSE676/14.1 Autoencoders.pdf
- https://arxiv.org/pdf/2003.05991.pdf
- https://arxiv.org/pdf/1906.02691.pdf
- https://www.tensorflow.org/tutorials/generative/autoencoder
- https://www.tensorflow.org/tutorials/generative/cvae

GAN

https://arxiv.org/pdf/1701.00160.pdf