

# みじかなところで使われているAI技術を使ってみた

【LT勉強会】みんなで語るAI活用勉強会 #2【ORION】

2025/12/4 Ken Shimizu

# 自己紹介


自己紹介: 大学は法学部出身のエンジニアで、現在は証券会社でエクセルとVBAで取引結果の帳簿への自動記録や、便利なツール類を作っています。何回か転職を繰り返していますが、以前はWeb系のJavaエンジニアでした。

趣味: カヤックフィッシング



# Web Application開発期

こんな感じで転職のために自宅にPCサーバーを立ち上げてインターネットに自由につながる環境で個人開発を始めました。昔から日本語の文字化けが嫌いだったので英語でつくりました。



Home  
Products  
Option Quote  
Option Calculator  
Option Strategy  
Macro Date Input  
Business  
Greeks  
Delta Hedge  
Gamma Hedge  
Technologies  
becho\_server.c  
opengamma.html  
svn.new.project  
Calendar Panel  
svn  
Raspberry Pi X ESP32  
Cam  
Face Detect  
Object Detect  
Object Face Detect  
Hobby  
Fishing  
Sake  
Robot1  
Robot2

This site introduces you a model of application to demonstrate new technologies, C++, Java, Javascript, Perl, Python, etc. The demonstration means not only just introducing new technologies but also how we organize project and development process with new technologies.

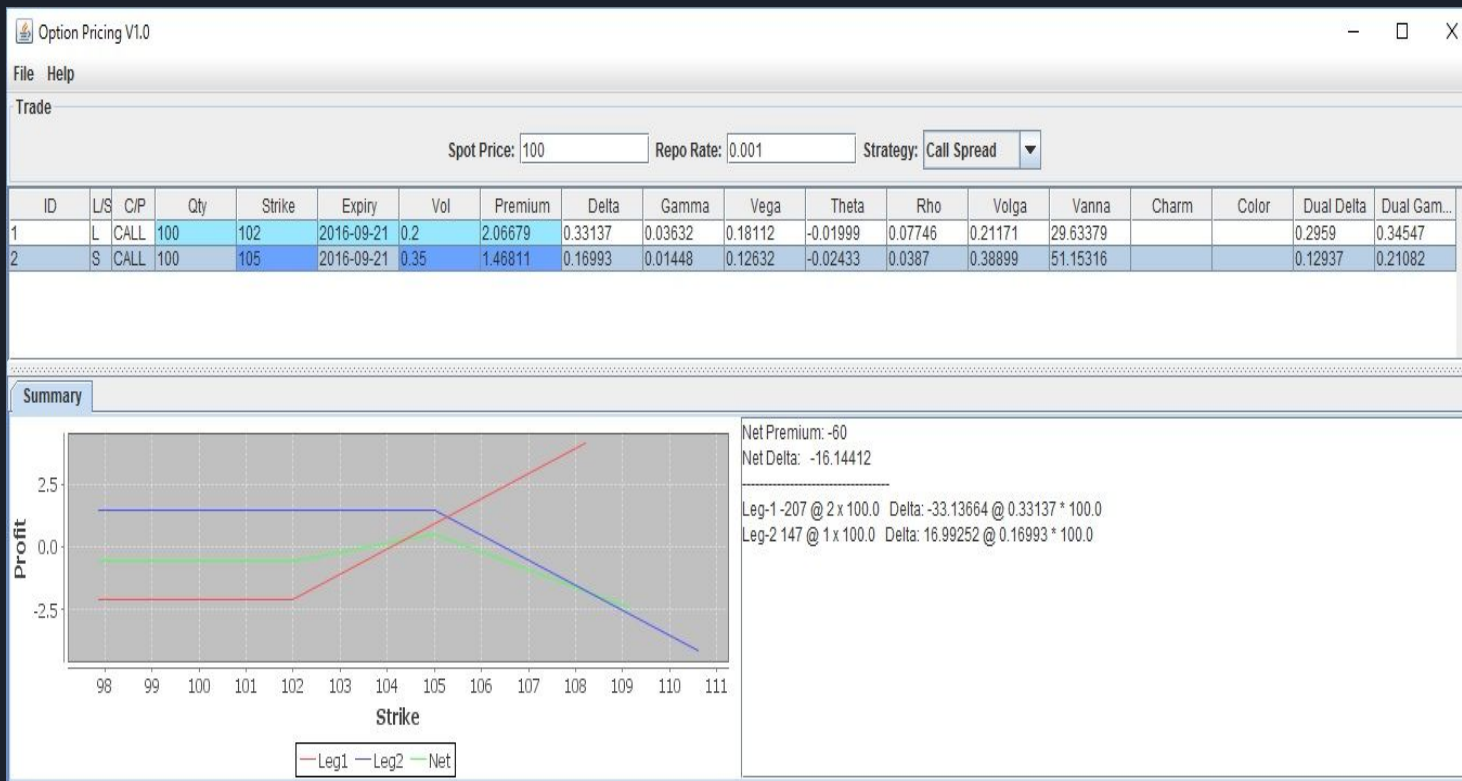
**Client Broker Option Quoting System**  
**Key Technology:** [React.js](#), [Redux](#), [Redux Thunk middleware](#) [MariaDB](#), [libwebsockets](#), [QuantLib](#) [RapidJSON](#)

This sample product introduces a model of high speed Internet application for financial industry through developing a prototype of server-client application over the Internet. The client is single page application (SPA) with React.js and the server is developed from scratch based on C++. The server-client communication is WebSocket. Web Socket and C++ application enables to handle high volume transaction over the Internet.

**Option Calculator**  
**Key Technology:** [React.js](#), [Redux](#), [libwebsockets](#), [QuantLib](#) [RapidJSON](#)

This sample product introduces a very simple model of high speed Internet application for financial industry through developing a prototype of server-client application over the Internet. The client is single page application (SPA) with React.js and the server is developed from scratch based on C++ and integrated with QuantLib which is Quant library. The server-client communication is WebSocket. Web Socket and C++ application enables to handle high volume transaction over the Internet.

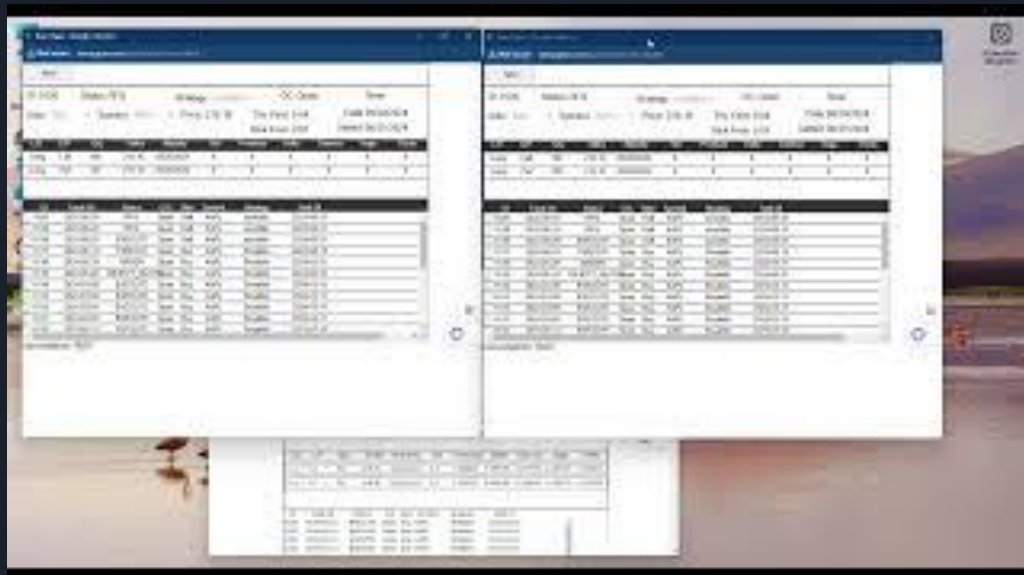
# Java SwingでRMIとRESTを使ってTomcatのバックエンドと通信するアプリケーション



# クライアントはReactJSで、サーバーサイドはC++でTest Driven開発

クライアントサーバー間は  
WebSocketで、メッセージは  
JSONを使っています。

サーバーサイドは、C++を使って、  
Test Drivenで品質をあげ  
ることができるか確認してみたかっ  
た。C++を使った理由は、  
TCP/IP、Thread、プロセスを直  
接使ってみたかったからです。



# Raspberry PIとの出会い

Raspberry PIと出会い、GPIOでサーボモーターやモータードライブを動かすことができることを知って、こんなもの作ってみました。コントローラーは、Raspberry PI側でApacheを動かしてWiFiを通じて、ReactJSを使ってコントローラーを作ってみた。

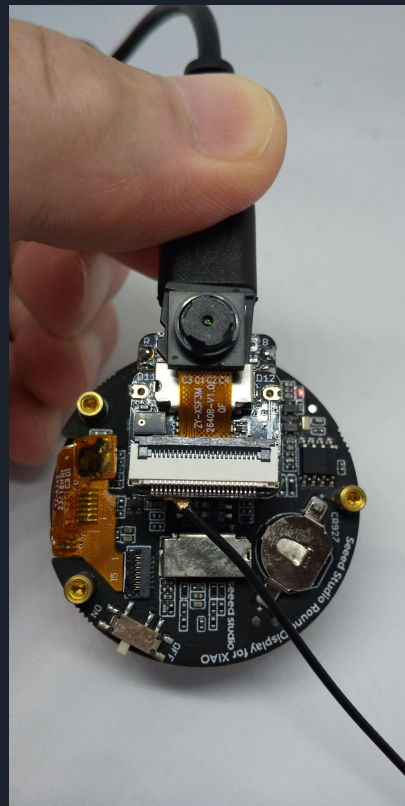


# ESP32でAI動かしてみた

マイコンでもTensorflowが動くことを知って、とりあえずESP32でPerson Detectionを動かしてみたら、遅いけど動いたじゃん！

ちょっと感動して、マイコンとAIの可能性をもっと知りたくなる。

ちなみにこれは、スマートウォッチの開発キット。






# Edge ImpulseとSpresenseでObject Detection

Edge ImpulseとSpresenseでObject DetectionをESP32の時と同じモデルを動かしてみると、推論に1.1秒かかっている。


調べると、使っていたESP32-S3ではMCUにTensilicaを使っていて、NPUがSoCでついていたのであった。








みなさま、忘れないでください！  
生成AIだけがAIではありません。




WKSは、Keyword Spotting Wake  
Word Detectionと言うやつです。  
あの、Hey SiriとかAlexaとかです。




Edge Impulseというサイトで簡  
単に作れるということで、

<https://studio.edgeimpulse.com/studio/615974>



反応が悪い理由は、私の発音が悪かったということです。ネーティブスピーカーのデータを使って教育とテストをしたためでした。



安易にネットのデータは使わず、  
きちんと自分でデータは用意を  
しましょう。



ニュース速報です



# ニュース速報

昨日、ブチ切れて退職の意志を伝えてしまいました。私の開発は、今日の日からアクセラレートしてぶっ飛ぶことになるでしょう。

来年のLT、乞うご期待！