



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。



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](#) [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

Option Pricing V1.0

File Help

Trade

Spot Price: 100 Repo Rate: 0.001 Strategy: Call Spread ▾

ID	L/S	C/P	Qty	Strike	Expiry	Vol	Premium	Delta	Gamma	Vega	Theta	Rho	Volga	Vanna	Charm	Color	Dual Delta	Dual Gam...
1	L	CALL	100	102	2016-09-21	0.2	2.06679	0.33137	0.03632	0.18112	-0.01999	0.07746	0.21171	29.63379			0.2959	0.34547
2	S	CALL	100	105	2016-09-21	0.35	1.46811	0.16993	0.01448	0.12632	-0.02433	0.0387	0.38899	51.15316			0.12937	0.21082

Summary

Profit

Strike

Net Premium: -60
Net Delta: -16.14412

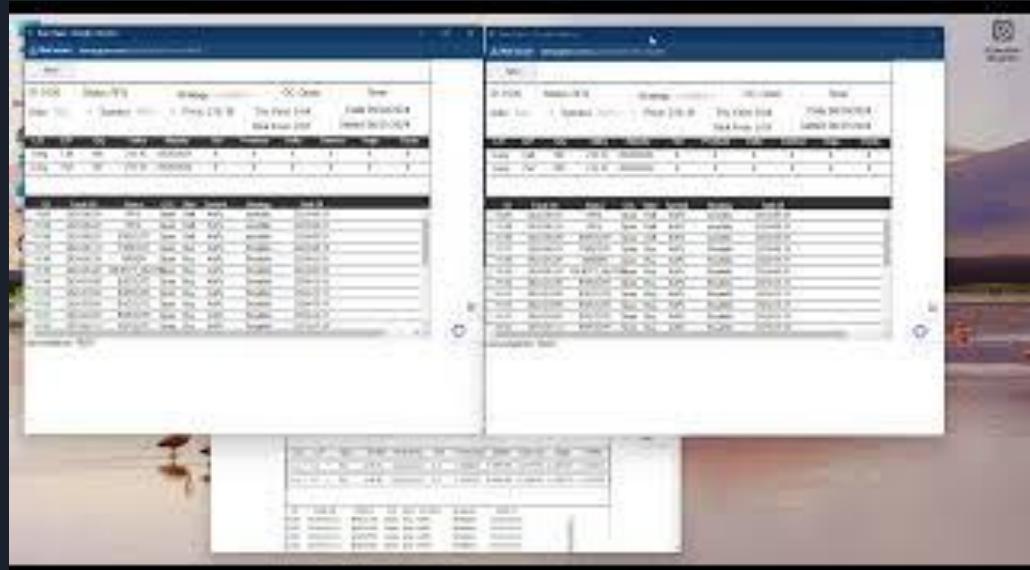
Leg-1-207 @ 2 x 100.0 Delta: -33.13664 @ 0.33137 * 100.0
Leg-2 147 @ 1 x 100.0 Delta: 16.99252 @ 0.16993 * 100.0

Leg1 Leg2 Net

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り。

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でついていたた
めだった。





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！