SecHack365

MagicDelta

開発駆動コース
仲山ゼミ 秋穂 正斗

自作VPLとレンダリングエンジンを用いた
AIセキュリティについて学習できるWebアプリ

X:Tebasaki_lab

MagicDeltaとは？

課題

AIが難しくて挫折...

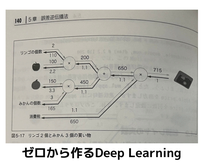
①AIセキュリティの認知度が低い！
②AI自体がめちゃくちゃ難しい！
③敷居が高いので誰もやらない！

数学が難しい...

$$\frac{\partial L}{\partial W} = \begin{pmatrix} \sum_{n=1}^{N}\sum_{n=1}^{N} x_{n,1} & \sum_{n=1}^{N}\sum_{n=1}^{N} x_{n,2} & \cdots & \sum_{n=1}^{N}\sum_{n=1}^{N} x_{n,1} \\ \sum_{n=1}^{N}\sum_{n=1}^{N} x_{n,2} & \sum_{n=1}^{N}\sum_{n=1}^{N} x_{n,2} & \cdots & \sum_{n=1}^{N}\sum_{n=1}^{N} x_{n,2} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{n=1}^{N}\sum_{n=1}^{N} x_{n,D} & \sum_{n=1}^{N}\sum_{n=1}^{N} x_{n,D} & \cdots & \sum_{n=1}^{N}\sum_{n=1}^{N} x_{n,D} \end{pmatrix} \begin{pmatrix} \sum_{n=1}^{N}\sum_{n=1}^{N} \frac{\partial L}{\partial y_{n,1}} & \sum_{n=1}^{N}\sum_{n=1}^{N} \frac{\partial L}{\partial y_{n,2}} & \cdots & \sum_{n=1}^{N}\sum_{n=1}^{N} \frac{\partial L}{\partial y_{n,H}} \end{pmatrix}$$

$$\frac{\partial L}{\partial w_{1,1}} = \frac{\partial L}{\partial y_{1,1}}\frac{\partial y_{1,1}}{\partial w_{1,1}} + \frac{\partial L}{\partial y_{2,1}}\frac{\partial y_{2,1}}{\partial w_{1,1}} + \cdots = \sum_{h=1}^{H} \frac{\partial L}{\partial y_{n,1}}\frac{\partial y_{n,1}}{\partial w_{1,1}}$$

動画や本の内容は動かせない...



ゼロから作るDeep Learning

観測がしにくい...

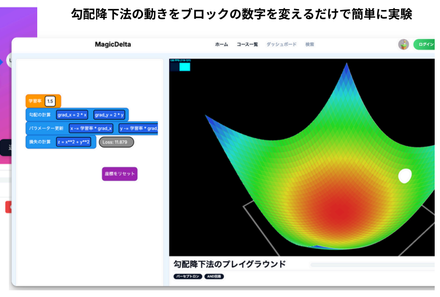
```
class SimpleCNN(nn.Module):
    def __init__(self):
        super(SimpleCNN, self).__init__()
        self.layer1 = nn.Sequential(
            nn.Conv2d(1, 32, kernel_size=3, padding=1),
            nn.ReLU(),
            nn.MaxPool2d(kernel_size=2, stride=2)
        )
        self.layer2 = nn.Sequential(
            nn.Conv2d(32, 64, kernel_size=3),
            nn.ReLU(),
            nn.MaxPool2d(2)
        )
        self.fc1 = nn.Linear(64 * 6 * 6, 1000)
        self.fc2 = nn.Linear(1000, 10)

    def forward(self, x):
        out = self.layer1(x)
        out = self.layer2(out)
        out = out.view(out.size(0), -1)
        out = self.fc1(out)
```

解決策

自作VPLとAnimationでAIセキュリティについて
効率よく学習できるコンテンツの開発

ブロック型のVPLでLLMのプロンプトインジェクションを体験

勾配降下法の動きをブロックの数字を変えるだけで簡単に実験

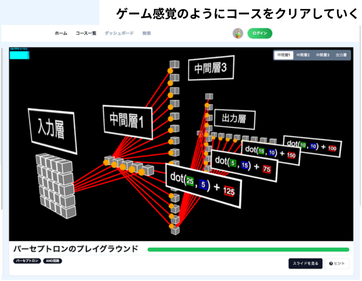


価値

①AIセキュリティの認知度が増える！
②AIを直感的に学習できる！
③ゲーム感覚でみんなが取り組む！

MagicDelta独自の競技システム

ゲーム感覚のようにコースをクリアしていく



AIセキュリティとは

AIセキュリティについて

- AIの脆弱性について定義する
  - 脆弱性＝AIへの不正な入力, AIによる間違った出力

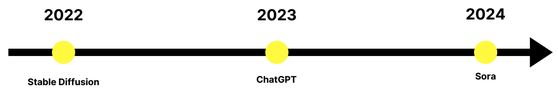
AIの脆弱性というものはかなり抽象的で、人によって意見が変わるため、今回のプロジェクトではこのように脆弱性を定義した。不正な入力というのは、ノイズや不正なデータが入った入力などを指し、間違った出力というのは、画像の認識率が極端におかしい、ハルシネーション、正解ラベルとは違うものを出力するなどが該当する。これを元に、どのようにしたら脆弱性を潰すことができるのか、脆弱性の仕組みについて学べるのかを考え、コンテンツ制作に取りくんだ。

AIへの攻撃の例

- 敵対的攻撃（サンプル）
  - 入力画像に人間が気づかないようなノイズを注入することにより、AIに間違った判定をさせてしまうもの
- 代表的な攻撃先
  - 自動運転
  - 顔認証
  - 文字認識

[2] Ian J. Goodfellow, Jonathon Shlens, Christian Szegedy. "Explaining and Harnessing Adversarial Examples." arXiv:1412.6572

AIはつい最近の出来事



ChatGPTやStable Diffusionなどは5年前ぐらいに感じている人が多いが、実際にはたった3年前のことになる。ChatGPTに関しては2年前であり、かなり最近だ。そして、AIセキュリティに関してはもっと最近のことであり、まだ情報が少ない。このSecHack365でAIセキュリティについて1年間取り組んだことは、かなり大きなアドバンテージとなった。

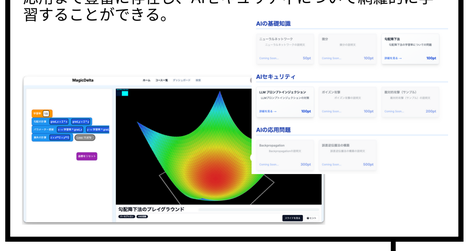
MagicDeltaの特徴

① 最高の学習体験

MagicDeltaの最大の特徴は、その学習体験にある。3Dで自由にAIモデルを観測できたり、AIに特化した自作VPLを搭載しているため、書籍に比べて圧倒的に分かりやすく、効率的にAIセキュリティの仕組みについて学習できる。

② コンテンツ

勾配降下法や敵対的攻撃などを学ぶために、それぞれコンテンツ（教材）が用意されている。このコンテンツは基礎から応用まで豊富に存在し、AIセキュリティについて網羅的に学習することができる。



③ 競技システム

これらを用いて構築されたのがMagicDelta独自の競技システムだ。ユーザーはCTF形式でAIセキュリティを学ぶことができる。例えば、うまく動かないAIモデルを修正したり、脆弱性のあるAIモデルに攻撃をし、間違った推論家かにするなど、既存のCTFにかなり近い形式をとっている。今後、この独自の競技システムは今後有名なものになる。

テーマの歴史

第1回イベント	6月15日(土)	テーマ：AI脆弱性の可視化
第2回イベント	7月19日(金)～21日(日)	テーマ：敵対的サンプル等の可視化
第3回イベント	9月27日(金)～29日(日)	
第4回イベント	11月15日(金)～17日(日)	テーマをAIセキュリティの学習に変更
第5回イベント	2025年 2月1日(土)・2日(日)	

最初に応募した時は、「AIの脆弱性の可視化」だったが、イベント回で様々な方からアドバイスをいただき、AIセキュリティを学習するために技術を使った方が良いと判断し、最終的に「AIセキュリティの学習」というテーマに変更された。開発駆動はとにかく手を動かして開発してみるのがモットーのため、テーマ変更に柔軟さを感じた。

VPLの思想

私は、Scratch（ビジュアルプログラミング言語）が初めて触った言語であり、つまり親である。もう8年間もScratchをやっているが、そこで得られた経験は素晴らしいものだった。環境構築の必要すらなく、キーボードが上手く使えない小学生でも、ブロックをマウスで繋げて複雑なプログラミングができたおかげで、中学生にもなる前に遺伝的アルゴリズムや自作RPGゲームエンジンを作るなど、とても楽しくプログラミングできた。もしPythonやJSから入っていたら、タイピングの時点で詰まり、英単語の意味やテキスト言語特有の構文エラー、複雑な構文でプログラミングが大嫌いになっていただろう。

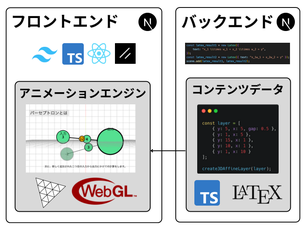
VPLには構文エラーもなく、構造やスコープも非常に分かりやすく表現され、初学者や特定ドメインに関してはテキスト言語よりも圧倒的に優秀だ。しかし、Twitterで「ビジュアルプログラミング」がトレンドに入るたび、エンジニアは総じて「VPLは開発に全く向いていない、差分管理もできなければテキストのように補完も効かない」などと批判ばかりが相次ぐ。極端な話、私の家族が批判されているのと同義だ。そんな世の中を根本から変えるべく、VPL開発を民主化し、VPLの良さが世界中に伝わるプロダクトを作り上げたいと思い、本プロジェクトを立ち上げた。

そして、SecHack365でAIについて学べるWebサイトを作るにあたり、私はVPLの強みを活かした新しい学習環境を構築することを決意した。AIは複雑な数式や高度なプログラミングが必要とされる分野だが、VPLを用いることで直感的に理解できる仕組みを作ることができる。AIモデルの学習過程やニューラルネットワークの動作を視覚的に表現し、誰でも試行錯誤できる環境を提供することで、初心者でもAIの本質を理解し、実践的なスキルを身につけることが可能になる。
今やAIはVPLの追い風になりつつある。DifyをはじめとしたAI技術の発展により、VPLの表現力や拡張性が大きく向上している。従来の「VPLは簡単なことしかできない」という固定観念を覆し、複雑なAI技術を誰もが手軽に扱える時代を切り開く。それこそが、SecHack365での私の挑戦であり、この分野で世界を変えられるのは私しかいないと確信している。

開発技術 / こだわりなど

技術構成図

今回のWebアプリは、フロントエンドとバックエンドに分かれており、主にフロントエンドが大きな役割を担っている。全体の技術要素としては、Next.js (App Router), React, TypeScript, Tailwind CSS, Three.js, shadcn, Canvas APIなどを用いた。

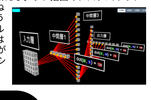


自作VPL

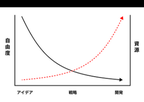
LLMのプログラムや勾配降下法などのコンテンツに存在するVPLは、全て自作であり、ライブラリを使っていない。レンダリングエンジンから動作ロジックまで全て自作することで、今までにないAIに特化したVPLを開発することができたのはそのためだ。描画にはBlocklyと同じくSVGを用いた。初期の頃はCanvas APIやThree.jsを使っていたが、レイアウトやStyle調整が難しいことが多いためSVGに乗り換えた。また、Reactで宣言的にVPLを記述し、ロジックは独自エンジンに任せるという「Headless VPL」という新しいアーキテクチャシステムを発明し、未踏ITに出す予定だ。ReactのStateは複雑なものには向いているが、OOPから脱却しているため、ゲーム技術のような大規模な状態管理には向いていないことがはっきりと分かった。

自作エンジン

2D/3Dで表示されているアニメーション、これらは全て自作エンジンによって描画されている。描画にはThree.jsを使ったが、それ以外は自作である。アニメーションを生成するロジックや自作のアニメーション記述言語を作るなど、良い映像を作れるように工夫した。よく、Webフレームワークやパーサーの作り方はネットに転がっているが、ManimやPlay Canvasのようなアニメーションエンジンの作り方は一切情報がなかったため、タイムラインの実装やアニメーションの操作、描画システムをうまくOOPを使いながら実装した。多分、日本でこのようなアニメーションエンジンを作っている人は数名もいないだろう。SecHack365でのコンテンツを実現するために、新たなグラフ描画のシステムやリファクタリングを行うなど、改良を重ねた。特に、アニメーションをどのような順番で処理するか、どのようなアルゴリズムで処理していくかの最適解はまだ見つかっていないが、SecHackが終わった後、TypeScriptの強力な型システムを用いて改良を続けていく。



開発でのこだわり



- リソース配分
  - コンセプト設計に時間を割き、開発が目的にならないようにした
- 動的なテーマ変更
  - 当初はAI脆弱性の可視化だったが、AIセキュリティの学習というテーマに変更した

開発駆動のため、開発が目的になりがちな部分がある。しかし、良いものを作るためにはアイデア、つまりコンセプト設計に時間をかける必要がある。グラフにある通り、多くの開発者はアイデアにリソースを使わず、開発にリソースをさく構造が多い。(右から左に時間が流れる)

SecHack365の開発ではこれを意識して開発を行うことで、柔軟かつ未踏的なプロダクトを作れた。

苦労した点

自作VPLとアニメーションエンジンに関しては私の専門なので根気よく進めればよかったが、AIセキュリティに関してはまだ情報が少なく、コンテンツを作るのにかなり難航してしまった。

実際、敵対的サンプルのプログラムやアルゴリズムについて勉強したが、初学者に適したコンテンツを作ることができなかった。しかし、今回で終わりではない。今回のプロダクトは今後も継続開発を行い、敵対的サンプルやハルシネーションの仕組みを分かりやすく伝えられるようなコンテンツ作りをしていく。

外部での発表

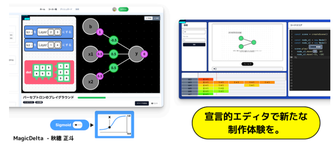
SECCON 13 Contest of Contest 入賞

2024年12月、こちらのコンテストに応募した。以前のオフラインイベント回でトレーナーに応募してみたら？と誘われたのがきっかけだった。結果入賞し、3/1の電脳会議でワークショップを行うことに。

SECCON 13 電脳会議
2025.3.1 (Sat) - 2(Sun) in 浅草橋ヒューリックホール&カンファレンス

福岡未踏 2024

経済産業省 福岡未踏に2024年度に採択。これでは、実は同じWebサイトの開発を行っていたが、ちゃんと棲み分けをしている。福岡未踏では自作VPLやアニメーションエンジンの基盤技術を開発し、SecHack365ではそれを用いてコンテンツ開発や足りない機能や新しい設計を導入した。つまり、福岡未踏でFWを作り、それを元にWebサイトをSecHack365で作っていたと表現するのが分かりやすい。実際、双方向で見つかった課題を並行して解決したことも多々あった。



今後について

社会実装

- toB
  - 大学の授業 - 10件
  - 企業の研修 - 5件
- toC
  - ユーザー数 5万人
  - MAU 1000人

機能開発

- ログイン
  - Google, emailログイン
- 学習ログ
  - 日々の勉強内容やクリア数など
- コンテンツ開発
  - +50コース
- OSS化
  - VPL
  - アニメーションエンジン

3年以内に実現

この1年で学んだこと

SecHack365の1年間で、私は沢山のことを学んだ。もともと、AIの基盤技術となる線形代数、誤差逆伝播、RNNやCNNなどはすでに知っていたが、AIセキュリティや、そもそもセキュリティについては素人だった。去年は100冊ほど専門書を購入したが、そのうち、SecHack365での活動を後押ししてくれた本たちを紹介する。基本的に機械学習、AI周りの本がもちろん多いが、SecHackのメンバーがOAuthをやっていたり、Web系のセキュリティについて詳しいのに憧れたため、認証認可やWebセキュリティについても個人で勉強した。SecHack365に参加する人はこれらのセキュリティについて詳しい人が多いが、私のようにAIに詳しい人はあまりおらず、AI × セキュアなシステムを作れるようになったのは強みになると実感した。

謝辞

本活動を進めるにあたり、多くの方々に支えられました。まず、ゼミの指導教員である仲山先生に、貴重なご指導を賜り、心より感謝申し上げます。また、メンバー、トレーナー、アシスタント、外部講師の皆様にも、多大なるご支援をいただきました。
さらに、SecHack365に取り組む環境を支え続けてくれた家族に深く感謝いたします。皆様のご協力と応援のおかげで、ここまで成長することができました。心より御礼申し上げます。

NICT National Institute of Information and Communications Technology