作成日：2025年06月01日

スキルシート

基本情報

|  |  |  |  |
| --- | --- | --- | --- |
| ふりがな | こう　しゅんけつ | 学校名 | 東京情報クリエイター工学院専門学校 |
| 氏名 | 黄　俊傑 | コース | AIシステムデータサイエンスコース |

スキル一覧

|  |  |  |  |
| --- | --- | --- | --- |
| 種類 | | 使用ツール | 学習内容／スキル |
| OS | Windows | Windows 11 | 基本的な操作が可能です。 |
| Linux | Tera Term 5 | ファイルの作成・編集、コマンドラインを使ったディレクトリ操作、簡単なシェルスクリプトの実行を習得しています。 |
| 言語 | Python | Python 3.11 Visual Studio Code Google Colab Amazon SageMaker Jupyter Notebook | データ分析、機械学習（分類・回帰）・ディープラーニングモデルの構築・可視化・検証・デプロイの実施経験あり。 SageMakerを用いて、分類・回帰・画像認識・チャットボット開発などのプロジェクトに取り組みました。 HTMLと連携させて簡易的な Web アプリケーションの作成も可能です。 |
| ライブラリ： Pandas、NumPy、Matplotlib、Seaborn、Scikit-learn、XGBoost、OpenCV、TensorFlow。 |
| Java | Visual Studio Code | 基礎的な文法やオブジェクト指向プログラミングを学びました。 |
| C/ C++ | Dev-C++ Visual Studio Code | 大学の履修科目。簡単な実装が可能です。 |
| HTML/ CSS | HTML 5 / CSS 3 Visual Studio Code Adobe Dreamweaver | Webサイトの基本構造とスタイル設定を理解しています。 スマートフォン対応のレスポンシブデザインや、Djangoと連携した簡易的なCRUD機能付きWebアプリの画面実装を経験しました。 |
| フレーム  ワーク | Django | Django 5.1.3 Visual Studio Code | データベースと連携したWebアプリケーションを開発可能です。 |
| クラウド | AWS | EC2 & IAM & S3 Amazon SageMaker Amazon Q Amazon Lex & Lambda AWS Rekognition Aurora & RDS | SageMakerによる機械学習モデル構築・デプロイを実践。 Amazon LexとLambdaを用いたチャットボット開発、Rekognitionを用いた顔検出プログラムの開発経験あり。 IAMによる権限管理、S3によるデータ保存、EC2・RDSを用いた基本的なインフラ構築も学習済みです。 |
| DB | SQL | MySQL 8.0.21 dokoQL | 基本文法やデータ操作（SELECT・INSERT・UPDATE等）に加え、集計関数・グループ化を利用したデータ分析も実践しました。簡単なJOIN文を用いたリレーション操作も理解しています。 |
| Office  ソフト | Word | Microsoft 365 | ビジネス文書、レポート、報告書が作成可能です。 数式・関数・グラフを利用した集計が可能です。 Pythonによるデータ処理結果との比較・確認にも利用しました。 |
| Excel |
| その他 | グループ ウェア | Microsoft Teams Slack | 在学中の学校の授業で日頃から利用しており、使い慣れています。 |

制作実績

|  |  |  |
| --- | --- | --- |
| 制作課題 | 使用ツール | 内容 |
| 幸福スコアの予測モデル（回帰） | Python 3.11.9 Google Colab | 世界幸福度ランキングデータを用いて、各国の幸福スコア（Happiness Score）を予測する回帰モデルを構築し、主要な特徴量の影響を評価することです。 |
| 睡眠と健康の関係分析（探索的データ分析） | Python 3.11.9 Google Colab | 睡眠時間や生活習慣に関するデータを用いて、睡眠障害の有無と関連する特徴量（年齢、BMI、運動習慣など）を可視化・分析した。カテゴリごとの傾向をグラフで示し、特徴量の相関を探索的に調査しました。 |
| ペンギン分類モデルのクラウド構築（バイナリ分類） | Python 3.11.9 Google Colab Amazon SageMaker | ペンギンデータを用いて、分類モデルを構築。前処理・特徴量エンコーディング後、訓練／検証／テストに分割し、AWS S3に保存。XGBoostを使ってSageMakerで学習・デプロイし、推論精度を評価しました。 |
| 顔画像の一致確認（画像認識・クラウドAI） | Python 3.11.9 AWS Rekognition Pillow | AWSのRekognitionサービスを用いて、2枚の人物画像の中で一致する顔を検出。類似度や信頼度を出力し、検出した顔部分を赤枠で可視化・保存した。クラウド画像認識APIの活用を実践しました。 |
| Lexチャットボットによる歯科予約管理 | Python 3.11.9 AWS Lambda Amazon Lex | AWS Lexと連携するチャットボットのバックエンド処理をLambda関数で実装。スロットの入力確認・バリデーション・候補提示・予約確定などを段階的に処理し、自然な対話で歯科予約を可能にしました。 |
| 表情認識システム（画像分類） | Python 3.11.9 Google Colab OpenCV TensorFlow | FER-2013データセットを用いてCNNモデルを構築し、顔画像から7種類の感情（怒り、嫌悪、恐れ、幸福、悲しみ、驚き、無表情）を分類する表情認識システムを制作しました。画像をアップロードして推論を実行し、その結果をCSVファイルに保存します。複数顔や複数画像の対応、可視化機能、Colab動作、モジュール化など、実用的な構成に仕上げています。 |

その他・アピールポイント

「幸福スコアの予測モデル」というテーマで機械学習分析を行いました。使用したデータは「World Happiness Report」で、GDPや健康、自由度など7つの要因を特徴量として、幸福スコアを回帰モデルで予測しています。データ前処理では、外れ値をIQR法で除去し、線形回帰モデルには標準化を施しました。線形回帰、ランダムフォレスト、XGBoostの3種類のモデルを比較し、交差検証も行った結果、ランダムフォレストが最も安定して良い精度を出しました。また、ランダムフォレストについては木の本数をハイパーパラメータとして調整し、300本が最適であることが分かりました。今回の分析では、各国の幸福に影響を与える特徴をモデルを通じて定量的に把握でき、予測精度も非常に高い結果となりました。

「睡眠と健康の関係分析」というテーマで、睡眠時間や生活習慣と睡眠障害との関係を探索的に分析しました。使用データは「Sleep health and lifestyle dataset」で、年齢、BMI、運動習慣、職業などの特徴量と睡眠障害の有無との関係を、棒グラフや箱ひげ図を用いて可視化しました。特に、運動習慣や職業、年齢層によって睡眠障害の発生傾向に違いが見られ、健康と生活習慣の関連性を視覚的に明らかにすることができました。

ペンギンの種別（二値分類）を予測する機械学習モデルを構築しました。特徴量には体長・体重・くちばしの長さなどを使用し、XGBoostを用いてバイナリ分類モデルを作成しました。前処理ではカテゴリ変数のエンコーディングを行い、データを学習・検証・テストに分割。モデルはAWS SageMaker上で学習させ、推論用エンドポイントをデプロイ。AUCや混同行列で評価を行い、高い分類精度を確認しました。クラウドを用いた一連の機械学習プロセスを実践的に学ぶことができました。

「顔画像の一致確認」というテーマで、AWSのRekognitionサービスを利用した画像認識プログラムを作成しました。2枚の人物画像を比較し、一致する顔を検出する処理を実装しました。boto3を用いてクラウドAPIと連携し、一致度（類似度スコア）や信頼度、バウンディングボックス情報を取得し、検出結果を赤枠で可視化しました。クラウドAIサービスを活用した画像処理の流れを実践的に学ぶことができました。

「歯科予約管理」というテーマで、Amazon LexとLambdaを活用した会話型チャットボットを実装しました。ユーザーの入力に応じて予約内容（種類・日付・時間）を順次確認し、条件に合った予約枠を提示・確定するロジックを構築。スロットのバリデーションや、日付・時間の自動候補生成、可用性管理なども実装し、自然な対話フローで予約処理を実現しました。クラウド上でのAI対話処理と業務ロジックの連携を実践できたプロジェクトです。

「表情認識システム」というテーマで、画像処理とディープラーニングを組み合わせた感情分類モデルを構築しました。使用したデータは顔表情に特化した「FER-2013」データセットで、怒り・嫌悪・恐れ・幸福・悲しみ・驚き・無表情の7種類の感情をCNNモデルにより分類しています。モデル構築には TensorFlow/Keras を用い、学習済みモデルを .keras 形式で保存。推論では、OpenCV を利用してHaar Cascadeによる顔検出を行い、検出された顔部分をグレースケール化・リサイズ・正規化したうえでモデルに入力します。システムはColab上でもローカルでも動作するように整備されており、アップロードされた画像から感情を判定し、その結果を画像上に表示するとともに、CSVファイルに保存可能です。また、顔検出や前処理などは .py モジュールに分離し、機能を関数化・再利用可能な構造としました。今回の開発を通じて、画像分類タスクにおける前処理・モデル推論・結果の可視化や保存処理を一連のワークフローとして構築する経験が得られました。特に、複数画像・複数顔の処理、精度の高い推論処理、およびユーザー向け出力形式の整備に注力しました。

以上