**Anti Manipulation Brigade - Dark Pattern Detector**

**Table of Contents**

1. Introduction

1.1 Purpose

1.2 Features

1.3 Technologies Used

1. Installation

2.1 Prerequisites

2.2 Installation Steps

1. Usage

3.1 Browser Extension

3.2 Detection Process

1. Architecture

4.1 Overview

4.2 Components

1. Data Source

5.1 Kaggle Dataset

5.2 Joblib Data

1. Future Enhancements

6.1 User Feedback

6.2 Additional Patterns

6.3 Performance Improvements

1. Acknowledgments
2. Contact Information

**1. Introduction**

**1.1 Purpose**

Anti Manipulation Brigade is a web extension developed for detecting dark patterns on ecommerce websites. Dark patterns are deceptive user interface design elements that manipulate users into taking actions they might not otherwise perform, such as making unintended purchases or subscribing to services unknowingly. The goal of this project is to empower users to identify and avoid such manipulative practices.

**1.2 Features**

* **Real-time Detection:** Anti Manipulation Brigade scans ecommerce websites in real-time to identify and highlight potential dark patterns.
* **User-Friendly Interface:** The browser extension provides an intuitive interface to users, making it easy to understand the detected manipulative elements.
* **Configurability:** Users can customize the extension settings to adjust the sensitivity of the detection algorithm and personalize their browsing experience.

**1.3 Technologies Used**

* **Python:** The backend of the extension is built using Python, leveraging the Flask framework for web development.
* **Flask:** Flask is used for creating a lightweight and modular web application that interfaces with the browser extension.
* **Database:** The dataset used for training and testing the dark pattern detection algorithm is sourced from Kaggle, and additional data is acquired from Joblib.

**2. Installation**

**2.1 Prerequisites**

* Ensure you have Python installed on your machine (version 3.x recommended).
* Web browser compatible with the extension (e.g., Google Chrome, Mozilla Firefox).
* Internet connection for real-time detection.

**2.2 Installation Steps**

1. Clone the repository: **git clone** [**https://github.com/TeamcodeARC/Anti-Manipulation-Brigrade.git**](https://github.com/TeamcodeARC/Anti-Manipulation-Brigrade.git)
2. Change directory to api: **cd api**
3. Install required Python packages: **pip install -r requirements.txt**
4. Run the Flask application: **python app.py**
5. Install the browser extension by loading it as an unpacked extension in your browser.

**3. Usage**

**3.1 Browser Extension**

* Once installed, the Anti Manipulation Brigade extension icon will appear in your browser toolbar.
* Click on the icon to open the extension interface and access settings.

**3.2 Detection Process**

* Visit an ecommerce website.
* The extension will automatically scan the webpage for dark patterns.
* Detected patterns will be highlighted, and relevant information will be displayed to the user.

**4. Architecture**

**4.1 Overview**

The architecture consists of a backend server (Flask), a browser extension frontend, and a dark pattern detection algorithm.

**4.2 Components**

* **Backend (Flask):** Handles communication between the browser extension and the dark pattern detection algorithm.
* **Browser Extension:** Provides a user interface for interacting with the extension and displaying detected patterns.
* **Dark Pattern Detection Algorithm:** Utilizes machine learning models trained on the Kaggle dataset to identify manipulative design elements.

**5. Data Source**

**5.1 Kaggle Dataset**

The Kaggle dataset provides a diverse set of examples for training the dark pattern detection algorithm. It includes labelled instances of websites with and without dark patterns.

**5.2 Joblib Data**

Additional data from Joblib is used to enhance the accuracy of the detection algorithm. This dataset includes specific patterns related to job-related manipulations.

**6. Future Enhancements**

**6.1 User Feedback**

Collect user feedback to improve the detection algorithm and address false positives/negatives.

**6.2 Additional Patterns**

Expand the dark pattern detection algorithm to identify new and evolving manipulative practices.

**6.3 Performance Improvements**

Optimize the extension for speed and efficiency, ensuring a seamless browsing experience.

**7. Acknowledgments**

We would like to express our gratitude to the Kaggle community for providing the valuable dataset and to Joblib for contributing additional data to enhance the capabilities of Anti Manipulation Brigade.

**8. Contact Information**

For inquiries, feedback, or support, please contact

Email: [codearcteam@gmail.com](mailto:codearcteam@gmail.com)

Github: <https://github.com/TeamcodeARC>

LinkedIn: <https://www.linkedin.com/company/teamcodearc>

Instagram: <https://www.instagram.com/teamcodearc>

Devfolio: <https://devfolio.co/@teamcodearc>