World Journal of Advanced Engineering Technology and Sciences

eISSN: 2582-8266

Cross Ref DOI: 10.30574/wjaets

Journal homepage: https://wjaets.com/

(Research Article)

Deceptive behavior analysis using deep learning

Vijayajyothi Chiluka, Pravalika Bandi *, Pranathi Enumula, Laxmi Sowjanya Korvi and Varun Teja Seelam

Department of CSE (Data Science), ACE Engineering College, Telangana, India.

World Journal of Advanced Engineering Technology and Sciences, 2025, 15(02), 647-654

Publication history: Received on 25 March 2025; revised on 02 May 2025; accepted on 04 May 2025

Article DOI: https://doi.org/10.30574/wjaets.2025.15.2.0569

Abstract

In today’s digital landscape, deceptive design patterns—such as fake urgency messages, misleading buttons, and disguised advertisements—are increasingly used to manipulate user behavior on websites. This project, titled “Deceptive Behavior Analysis Using Deep Learning,” presents an automated solution to detect such deceptive elements. It uses a Chrome extension to capture webpage screenshots and DOM content, which is then processed by a Flask backend. Text is extracted using the EAST deep learning model, vectorized using TF-IDF, and analyzed using two machine learning classifiers: Bernoulli Naive Bayes to detect the presence of deception, and Multinomial Naive Bayes to categorize the type of deceptive pattern. The results are stored in Excel and CSV for analysis. This system offers a scalable, real-time approach to identifying deceptive behaviors on websites and enhancing user protection.

Keywords: Deceptive Patterns; Deep Learning; East Text Detection; Machine Learning Classifiers; Chrome Extension; Real-Time Analysis

1. Introduction

In today's digital age, users are frequently exposed to deceptive design practices—commonly known as "dark patterns"—across websites and applications. These patterns are intentionally crafted interfaces that manipulate users into actions they may not have intended, such as subscribing to services, revealing personal information, or making unintentional purchases. The rise of such manipulative strategies poses serious ethical concerns and has prompted the need for automated detection systems that can assist in safeguarding user interests.

Traditional methods for identifying deceptive patterns often rely on manual auditing or rule-based systems, which are not scalable and cannot adapt to the rapidly evolving nature of web interfaces. These techniques also fail to analyze both textual and visual cues that play a major role in influencing user behavior. Therefore, there is a growing demand for intelligent systems that can automatically identify and classify deceptive elements by examining content from both image and DOM (Document Object Model) perspectives.

This project proposes a hybrid intelligent framework that uses Deep Learning and Machine Learning techniques to detect deceptive behavior on websites in real time. We leverage the EAST (Efficient and Accurate Scene Text Detector) model to extract text from captured website screenshots, while textual patterns extracted from the DOM are analyzed using trained Naive Bayes classifiers. The solution is integrated into a Chrome Extension, which allows users to scan any webpage with a single click and receive instant feedback on potential deceptive content.

The system is adaptable and scalable—new types of deceptive patterns can be incorporated through retraining with updated datasets. It also logs the captured data (image, DOM, prediction) into CSV and Excel files for future analysis or audit. By automating the detection process and supporting real-time interaction, this project provides an effective, user-friendly solution to address the growing concern of deceptive design practices on the internet.

* Corresponding author: Pravalika Bandi.

Beyond mere detection, the project also aims to categorize the type of deceptive behavior identified—such as hidden costs, forced continuity, or misleading navigation—offering users a clear explanation of why a specific pattern is considered deceptive. This classification is done using a separate machine learning model trained on labeled pattern strings from a curated dataset. By not only flagging suspicious elements but also providing context, the system enhances transparency and empowers users to make more informed decisions while browsing online platforms.

2. Literature review

Over the past decade, deceptive design patterns—also known as dark patterns—have gained attention due to their manipulative impact on user behavior. Initial studies highlighted how web interfaces can nudge users into unintended actions like subscriptions or data sharing. Brignull (2010) was one of the first to coin the term "dark patterns," categorizing them based on user deception tactics. Since then, multiple researchers have worked to catalog and expose these techniques, with efforts such as the Dark Patterns Tip Line and user studies indicating the psychological and ethical concerns associated with such manipulative designs.

In the field of automated detection, early approaches focused on rule-based systems and manual annotations, which lacked scalability and adaptability. More recent research introduced machine learning (ML) techniques for identifying deceptive language patterns within website text. Natural Language Processing (NLP) methods, including TF-IDF and various classifiers like SVM and Naive Bayes, have shown promise in detecting manipulative intent. However, these models often failed to consider visual cues such as misleading buttons or hidden opt-outs, leaving a gap in truly holistic detection.

To address these limitations, recent studies have explored deep learning models for visual analysis, particularly the use of the EAST (Efficient and Accurate Scene Text Detector) model for extracting text from images. These advances enable systems to analyze both textual content and visual interface elements simultaneously. Our project builds on this foundation by combining EAST for visual detection with ML classifiers trained on a labeled dataset of deceptive patterns. The integration of real-time browser interaction via a Chrome extension further enhances usability and brings the detection system closer to practical, real-world application.

3. Existing system

The current systems designed to detect deceptive patterns on websites are primarily manual or rule-based. These approaches rely heavily on human inspection or static rule-checkers to identify misleading UI elements, such as hidden checkboxes or confusing language. Tools like browser plug-ins and user reporting platforms (e.g., Dark Patterns Tip Line) help raise awareness but do not offer real-time automated solutions. Moreover, these methods typically focus only on textual content or user behavior logs, ignoring the visual layout and structural design of the interface, which are crucial in understanding how users are manipulated.

Additionally, most existing systems do not leverage deep learning or any advanced computer vision techniques to analyze the visual structure of web pages. They primarily focus on static textual analysis and fail to understand design elements like buttons, forms, overlays, or misleading colors that are meant to deceive users. This results in a major gap where subtle manipulations in modern interfaces remain undetected, leaving users unprotected and unaware.

4. Proposed system

The proposed model aims to automate the detection of deceptive design patterns (often called dark patterns) on websites by analyzing both the visual elements and the underlying DOM (Document Object Model) content of a web page. A browser extension is developed to capture a screenshot of the current webpage and extract textual content from the DOM. This data is then sent to a Flask-based API for further analysis. This allows the system to evaluate both how the content appears visually and what it semantically conveys, improving the chances of accurately identifying deceptive intent.

On the backend, the text from the DOM is processed using text vectorization techniques such as TF-IDF to convert it into a numerical format suitable for machine learning models. The project uses two different classifiers: one to detect the presence of deceptive behavior and another to categorize the type of deceptive pattern. Text extracted from the image using the EAST text detector is also considered for improved accuracy, allowing the model to see what users see — even if certain content is not in the raw HTML. The use of classification algorithms (excluding deep learning) ensures the system remains lightweight and explainable.

In addition to detection, the system stores every user interaction and prediction in both **Excel and CSV** formats for future analysis and model improvement. This helps create a growing dataset of deceptive and non-deceptive patterns. The

proposed model thus creates a **semi-automated framework** that not only identifies deceptive content in real-time but also builds a foundation for scalable future development. By combining image processing, NLP, and traditional machine learning, the system addresses current limitations while remaining accessible and transparent.

5. Methodology

The methodology outlines the structured approach adopted in designing, developing, and implementing the Deceptive Behavior Analysis using Deep Learning project. The goal of this project is to detect deceptive behaviors by analyzing multimodal data, including facial expressions and audio cues, using advanced deep learning techniques. The methodology focuses on efficient data preprocessing, feature extraction, and classification to identify whether an individual is behaving truthfully or deceptively.

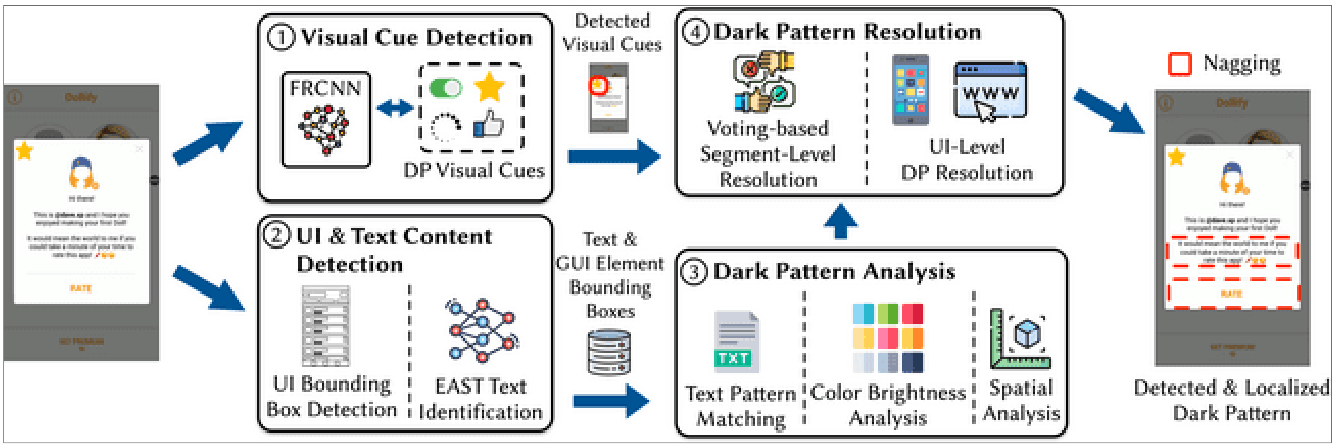

① Visual Cue Detection
FRCNN
DP Visual Cues
Detected Visual Cues
④ Dark Pattern Resolution
Voting-based Segment-Level Resolution
WWW
UI-Level DP Resolution
Nagging
② UI & Text Content Detection
UI Bounding Box Detection
EAST Text Identification
Text & GUI Element Bounding Boxes
③ Dark Pattern Analysis
TXT
Text Pattern Matching
Color Brightness Analysis
Spatial Analysis
Detected & Localized Dark Pattern

Figure 1 Methodology

5.1. System Architecture

The system follows a layered architecture where a browser extension captures user interactions and screenshots, sending them to a Flask backend. The backend preprocesses the data using tools like OpenCV and text vectorizers, then uses the EAST model for text recognition. Bernoulli and Multinomial Naive Bayes classifiers analyze behavioral and textual data. The results are then stored in local storage or a database, allowing for real-time predictions and activity logging.

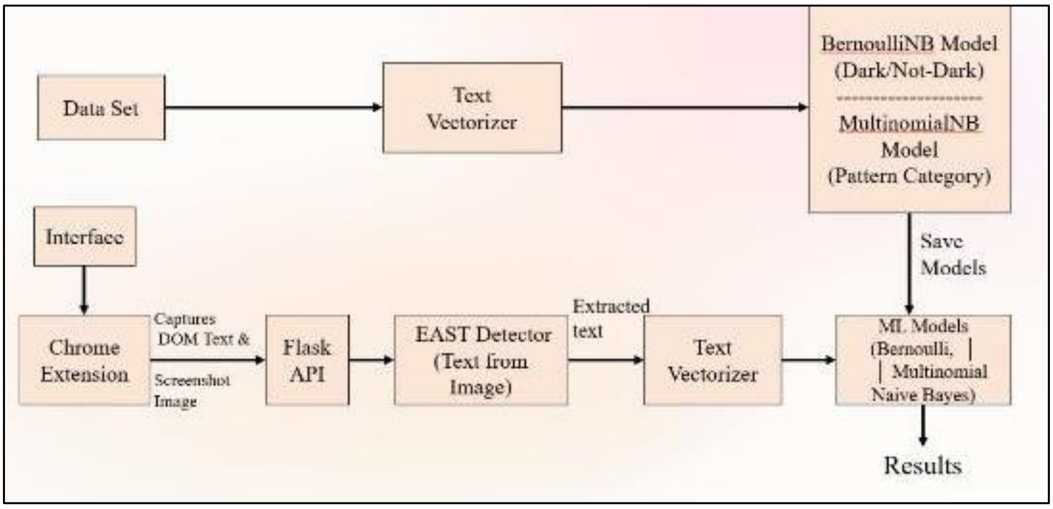

Data Set
Text Vectorizer
BernoulliNB Model (Dark/Not-Dark)
MultinomialNB Model (Pattern Category)
Save Models
Interface
Chrome Extension
Captures DOM Text & Screenshot Image
Flask API
EAST Detector (Text from Image)
Extracted text
Text Vectorizer
ML Models (Bernoulli, | | Multinomial Naive Bayes)
Results

Figure 2 System Architecture

5.1.1. Input Acquisition Layer:

- This is the entry point of the system where users provide through browser extension.
- It Captures screenshots of the current web page, Tracks user interactions (clicks, scrolls, inputs, time spent, etc.) and sends the collected data (images + behavior) to the backend API.

5.1.2. Preprocessing Layer:

Text/Image handling

- Images: Processed using OpenCV or Pillow (resize, grayscale, noise reduction).
- User Behavior/Text: Cleaned (e.g., stop word removal, tokenization) and vectorized for ML models.

5.1.3. Feature Extraction Layer:

Image/Text Features

- EAST Model extracts text from screenshots.
- Vectorizers like TF-IDF or CountVectorizer convert text and behavior data into numerical format.
- Binary vectors created for Bernoulli Naive Bayes, frequency vectors for Multinomial Naive Bayes.

5.1.4. Deep Learning and Machine Learning Layer:

- Text Recognition: EAST (Efficient and Accurate Scene Text Detector) for detecting and extracting visible text from screenshots.
- Classification Models:
  - Bernoulli Naive Bayes: Best for binary behavior features (deceptive / non- deceptive).
  - Multinomial Naive Bayes: Best for text-based classification (deceptive category).

5.1.5. Prediction Layer

The prediction layer interprets the input and returns categorized outcomes, such as identifying potentially risky behavior or content.

5.1.6. Storage Layer

Finally, the storage layer logs all interactions, predictions, and screenshots into local storage or a central database, enabling future analytics and auditability.

5.1.7. Output Layer

After both the image text and behavioral data are processed, the Flask API generates a final decision. Based on the analysis, the decision is made whether the website contains deceptive patterns or not. This decision, along with any additional insights or recommendations, is sent back to the browser extension. The extension then alerts the user about the potential deceptive nature of the site.

5.1.8. Evaluation and Testing

For model evaluation and testing, the dataset is split into training and testing sets to assess performance objectively. The Bernoulli and Multinomial Naive Bayes models are evaluated using standard classification metrics such as accuracy, precision, recall, and F1-score is applied to ensure the models generalize well across unseen data. Confusion matrices help analyze true/false positives and negatives, while ROC curves and AUC scores are used to measure the models' ability to distinguish between classes. The best-performing model is selected based on a balance of these metrics and is then integrated into the live Flask backend for real-time predictions.

The proposed methodolgy is designed with a layered architecture comprising a browser extension, a Flask backend, machine learning models, and storage. The browser extension acts as the input layer, capturing user interactions and screenshots from the web browser and sending them to the Flask backend API. In the preprocessing layer, the received data—whether textual or visual—is cleaned and formatted using tools like OpenCV for images and text vectorizers for behavioral logs. The feature extraction layer utilizes the EAST model to recognize and extract text from screenshots, while user behavior and textual inputs are transformed into numerical vectors suitable for classification. In the machine learning layer, Bernoulli Naive Bayes is applied to classify binary behavioral patterns, while Multinomial Naive Bayes is used for text-based classification. The prediction layer interprets the input and returns categorized outcomes, such as identifying potentially risky behavior or content. Finally, the storage layer logs all interactions, predictions, and screenshots into local storage or a central database, enabling future analytics and auditability. This layered setup ensures a modular, scalable, and intelligent system for real-time user behavior and content analysis.

6. Results and Discussion

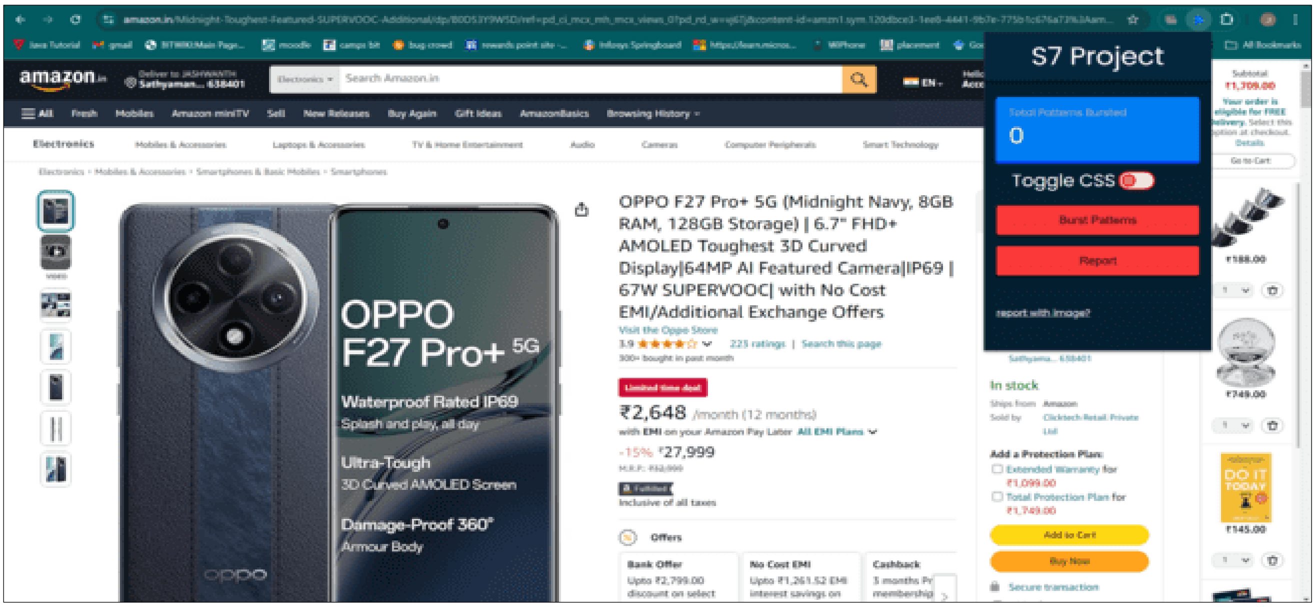

S7 Project
Total Patterns Bursted
0
Toggle CSS
Burst Patterns
Report
report with image?
OPPO F27 Pro+ 5G (Midnight Navy, 8GB RAM, 128GB Storage) | 6.7" FHD+ AMOLED Toughest 3D Curved Display|64MP AI Featured Camera|IP69 | 67W SUPERVOOC| with No Cost EMI/Additional Exchange Offers
₹2,648 /month (12 months)
-15% ₹27,999
In stock
Add to Cart
Buy Now
OPPO
F27 Pro+ 5G
Waterproof Rated IP69
Ultra-Tough
Damage-Proof 360°

Figure 3 User Interface (Before Prediction)

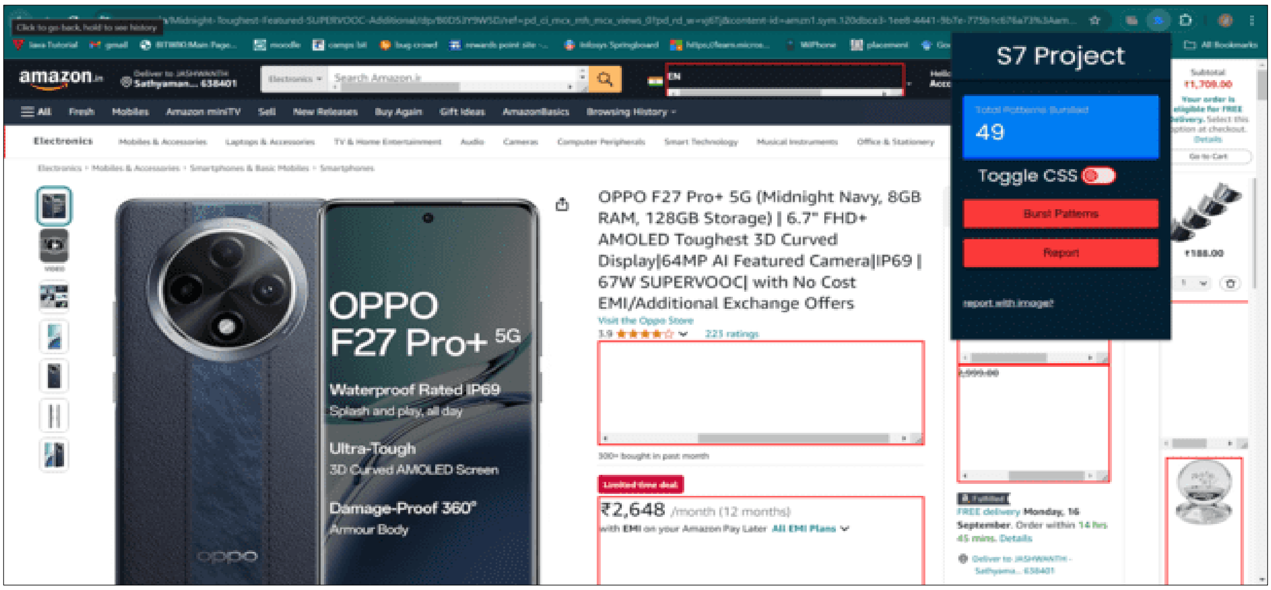
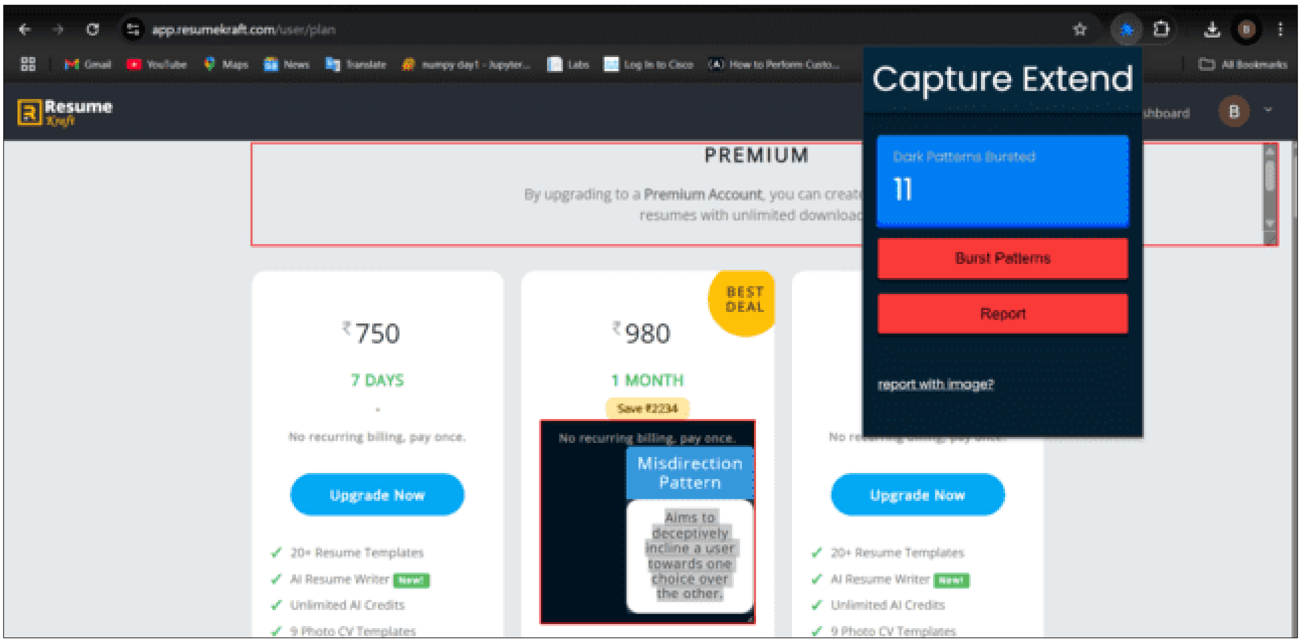

S7 Project
Total Patterns Bursted
49
Toggle CSS
Burst Patterns
Report
report with image?
OPPO F27 Pro+ 5G (Midnight Navy, 8GB RAM, 128GB Storage) | 6.7" FHD+ AMOLED Toughest 3D Curved Display|64MP AI Featured Camera|IP69 | 67W SUPERVOOC| with No Cost EMI/Additional Exchange Offers
₹2,648 /month (12 months)
OPPO
F27 Pro+ 5G
Waterproof Rated IP69
Ultra-Tough
Damage-Proof 360°

Figure 4 User Interface (After Prediction)

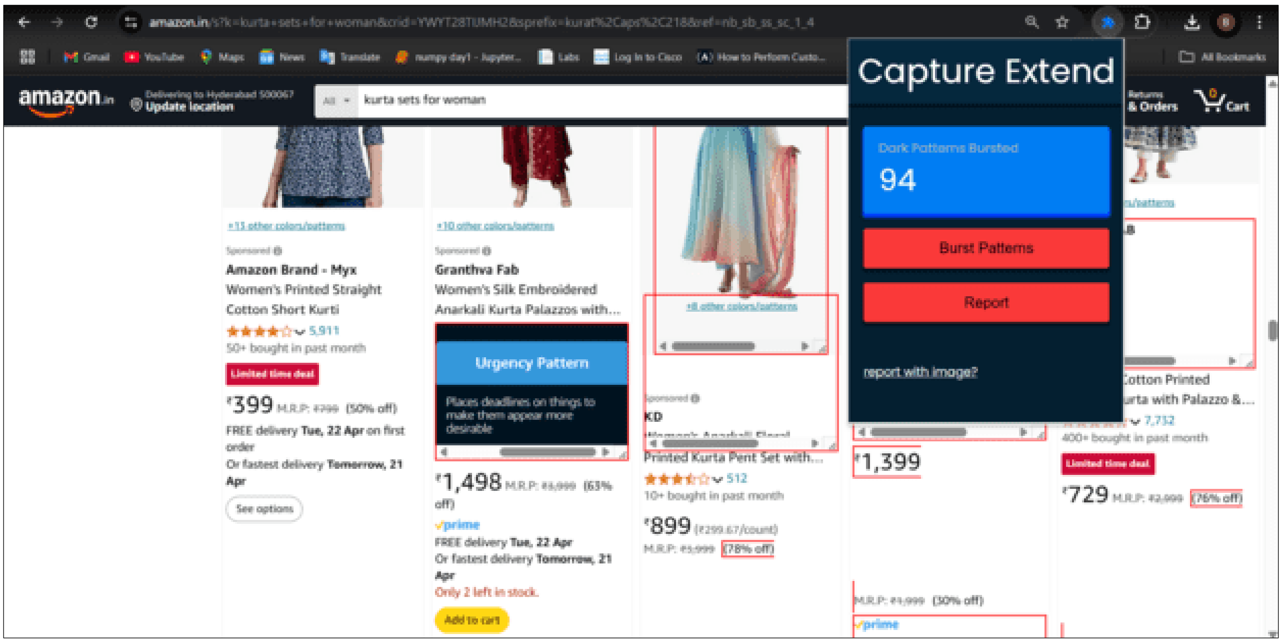


By upgrading to a Premium Account, you can creat… resumes with unlimited download…

No recurring billing, pay once.

Aims to deceptively incline a user towards one choice over the other.

Figure 5 Misdirection Pattern



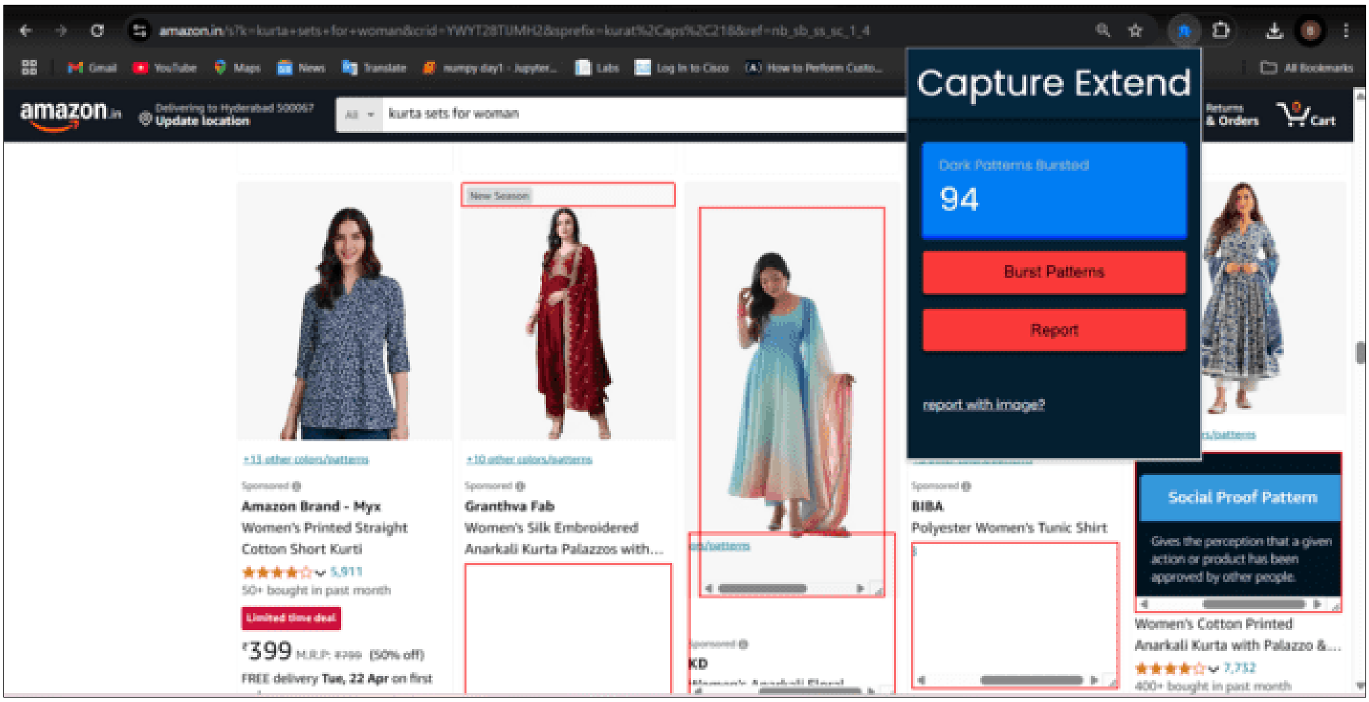
Places deadlines on things to make them appear more desirable

FREE delivery Tue, 22 Apr on first order

Or fastest delivery Tomorrow, 21 Apr

Only 2 left in stock.

Figure 6 Urgency Pattern


amazon.in/s?k=kurta+sets+for+woman&crid=YWYT28TUMH2&sprefix=kurat%2Caps%2C218&ref=nb_sb_ss_sc_1_4
Gmail
YouTube
Maps
News
Translate
numpy day1 - Jupyter...
Labs
Log In to Cisco
How to Perform Custo...
All Bookmarks
amazon.in
Delivering to Hyderabad 500067
Update location
All
kurta sets for woman
Returns & Orders
Cart
Capture Extend
Dark Patterns Bursted
94
Burst Patterns
Report
report with image?
New Season
+11 other colors/patterns
+10 other colors/patterns
Sponsored
Amazon Brand - Myx
Women's Printed Straight Cotton Short Kurti
5,911
50+ bought in past month
Limited time deal
₹399 M.R.P: ₹799 (50% off)
FREE delivery Tue, 22 Apr on first
Granthva Fab
Women's Silk Embroidered Anarkali Kurta Palazzos with...
KD
BIBA
Polyester Women's Tunic Shirt
Social Proof Pattern
Women's Cotton Printed Anarkali Kurta with Palazzo &...
7,752
400+ bought in past month

Gives the perception that a given action or product has been approved by other people.

Figure 7 Social Proof Pattern

7. Conclusion

This project successfully demonstrates the integration of visual and behavioral analysis using deep learning and machine learning models to detect deceptive UI practices. By leveraging a combination of frontend browser extension, backend Flask APIs, and robust classifiers, the system is capable of identifying potential deceptive interactions in real-time. The use of the EAST model ensures accurate text detection from UI screenshots, while Bernoulli and Multinomial Naive Bayes models help classify deceptive behavior based on user interaction and textual cues.

In conclusion, this solution offers an enhanced user experience by providing timely alerts and visual markers for potentially harmful interface elements. The modular structure of the system also supports scalability and easy integration with larger systems and monitoring platforms.

Compliance with ethical standards

Disclosure of conflict of interest

There is no conflict of interest.

References

[1] Zhou, X., Yao, C., Wen, H., Wang, Y., Zhou, S., He, W., & Liang, J. (2017). EAST: An Efficient and Accurate Scene Text Detector. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR https://arxiv.org/abs/1704.03155

[2] EAST: An Efficient and Accurate Scene Text Detector: Introduces the EAST model for real-time text detection in natural scenes: arXiv:1704.03155

[3] Smith, R. (2007). An Overview of the Tesseract OCR Engine. In Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR). https://ieeexplore.ieee.org/document/4376991

[4] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed Representations of Words and Phrases and their Compositionality. In Advances in Neural Information Processing Systems. https://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.pdf

[5] Russell, S., & Norvig, P. (2020). Artificial Intelligence: A Modern Approach (4th Edition). Pearson. https://aima.cs.berkeley.edu/

[6] Bishop, C. M. (2006). Pattern Recognition and Machine Learning. Springer. https://link.springer.com/book/10.1007/978-0-387-45528-0

[7] Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press. https://www.deeplearningbook.org/

[8] An Ontology of Dark Patterns: Foundations, Definitions, and a Structure Proposes a three-level ontology harmonizing existing taxonomies, defining 64 dark pattern types. https://colingray.me/wp-content/uploads/2024/02/2024_Grayetal_CHI_OntologyDarkPatterns.pdf

[9] Kowsari, K., Meimandi, K. J., Heidarysafa, M., Mendu, S., Barnes, L. E., & Brown, D. E. (2019). Text Classification Algorithms: A Survey. Information. https://www.mdpi.com/2078-2489/10/4/150

[10] Theorizing Deception: A Scoping Review of Theory in Research on Dark Patterns and Deceptive Design: Reviews theoretical frameworks applied in dark pattern research, identifying gaps and proposing future directions. https://arxiv.org/abs/2405.08832

Author’s short biography

Mrs. Vijayajyothi Chiluka is an Assistant Professor with an M.Tech in Computer Science and Engineering and is pursuing a Ph.D. She has 12 years of professional experience in the field of computer science. Her research interests include Deep Learning and Analysis of Algorithms.	
I am Pravalika Bandi , an undergraduate student pursuing a Bachelor's degree in Computer Science and Engineering with a specialization in Data Science. My research interest lies in Machine Learning, where I explore data-driven techniques and their applications. As an aspiring data science student , I am keen on expanding my knowledge and gaining hands-on experience in the field.	
Pranathi Enumula is an undergraduate student pursuing a Bachelor’s degree in Computer Science and Engineering with a specialization in Data Science. Her research interest lies in Artificial Intelligence, where she explores intelligent systems and their real-world applications. She is passionate about learning and applying AI techniques to solve complex problems.	
Laxmi Sowjanya Korvi is an undergraduate student pursuing a Bachelor’s degree in Computer Science and Engineering with a specialization in Data Science. Her research interest is in Data Science, where she explores data analysis techniques and their applications. She is eager to expand her understanding and contribute to advancements in the field.	
Varun Teja Seelam is an undergraduate student pursuing a Bachelor’s degree in Computer Science and Engineering with a specialization in Data Science. His research interest is in Data Science, focusing on extracting insights from data and developing data-driven solutions. He is keen on enhancing his knowledge and practical skills in this evolving field.	