Dark Patterns Buster Hackathon

Naitik : Dark Pattern Buster

February 27, 2024

Group Members

- Ashish Kumar: ashisndiitr@gmail.com
- Ashutosh Srivastava : ashutosh3002@gmail.com
- Vansh Uppal: vanshuppal2002@gmail.com
- Prakhar Singh: prakharsingh0908@gmail.com
- Nehal Chechani: chechaninehal@gmail.com

Contents

1 Inspiration **3**

2 Objective **3**

3 Dark Patterns **4**
3.1 General Dark Patterns 4
3.2 Identified and Resolved Dark Patterns 4

4 Solution Implemented **5**
4.0.1 Model Architecture and implementation 6
4.1 Architecture and Improvements Made 6
4.1.1 Data Handling and Processing 7

5 Software Solutions **7**
5.1 Checkbox Unchecker 7
5.2 ClickJacking 9
5.3 Price Transparency Checker 9
5.4 Fake Reviewer 10
5.5 Detect Confirm Shaming 12
5.6 Data protection 12
5.7 Subscription trickery 14

6 Tech stack used **14**

7 UI/UX Flow **14**
7.1 False Urgency 14
7.2 Disguised Advertisement 15
7.3 Basket Sneaking 16
7.4 Confirm Shaming 16

8 Feedback Loop **18**

9 Monitoring **19**

10 References **20**

1 Inspiration

In the highly competitive world of e-commerce, retailers are found to implement dark patterns influencing user behavior in many ways. The term "dark patterns" refers to deliberate and misleading design choices that use human psychology to influence unintentional and undesirable decisions, hence adding value to the service using them.

The psychology underlying heuristics is explored in Kahneman's Thinking, Fast and Slow, where he discusses how humans employ Systems 1 and 2 to make decisions. System 1 represents the quick decision-making brain which is reflexive and intuitive. It is usually an automatic response that feels effortless and uncontrolled. System 2 is reflective and controlled side of the brain. Its the side of the brain that does all the thinking and is deductive, self- aware and rational.

System 1 leverages cognitive biases to help with these quick decisions. Dark patterns operate through the exploitation of these cognitive biases.When we encounter stimuli online, System 1 generates immediate, intuitive responses without much conscious thought. Dark patterns take advantage of this rapid decision-making process by exploiting cognitive biases and emotions. Because System 1 operates unconsciously, users are less aware of manipulative design tactics, making them more susceptible to the deceptive nature of dark patterns. This automatic and emotionally charged thinking allows these design strategies to influence user behavior effectively, as individuals may not fully recognize the subtle cues that lead them into unintended actions online.

2 Objective

Design and prototype innovative app or software-based solutions that can detect the use, type, and scale of dark patterns on e-commerce platforms. Our objective is divided in 3 phases :

- Phase 1 will go into understanding and classification of Dark Patterns. Research based approach to mitigate the effects of dark patterns based on their effect on the users is done. Bases on our data, we are implementing and designing software based strategies. This report is the product of Phase 1.
- Phase 2 will move on to the implementing of the research to make a user-friendly cross-platform web-extension to prepare, notify and protect users from dark patters, details of which can be found in the following sections.
- Phase 3 will be development of a website, with a central repository management, creating a feedback loop, and admin panel to quantify and notify the government as well as persist tha data and keep a close eye on the dark patterns.

3 Dark Patterns

3.1 General Dark Patterns

In this section, certain general dark patterns are explained, revealing the intricate methods through which user experiences undergo manipulation across diverse online platforms.

- "Trick question" is a technique used to fool a user into answering a question they didn't mean to. This is frequently accomplished by confusing copywriting or by using "opt in" and "opt out" checkboxes.
- "Misdirection" diverts the user's focus from one option, by displaying a more (visually) prominent choice.
- "Bait and switch" refers to the practice of tricking the user into selecting an option that leads to an unexpected outcome by using popular UI conventions.
- "Disguised ads" are advertisements that replicate other forms of user interface content, including buttons, graphics, or navigation, in an attempt to fool people into clicking on them and generate advertisement revenue.
- "Roach motel" a revenue model, is frequently employed in subscription services. The design of a user interface makes it easy for the user to purchase a service, but hard to cancel it.

3.2 Identified and Resolved Dark Patterns

Here is a list of dark patterns that we've spotted and successfully addressed in the further solution.

- "Privacy Zuckering" is when a user agrees to something without acknowledging the full extent of their consent. Pre-checked checkboxes on e-commerce platforms exemplify this dark pattern, subtly nudging users towards unintended actions. This manipulative design exploits users' tendency to overlook default settings, leading to inadvertent consent or unwanted purchases. Such pre-selection undermines user autonomy, creating an unethical digital environment that prioritizes business interests over user choice.
- "Urgency and False Scarcity" dark pattern pressures users to make hasty purchases by employing tactics like countdown timers and limited-time messages. Some countdown timers are deceptive, falsely signaling an expired offer that remains valid. In the realm of "Scarcity," false urgency is created through deceptive messages about low stock or high demand, manipulating perceived product scarcity to boost desirability and drive sales.
- "Astroturfing" or "Fake Reviews" is the deceptive practice of creating fake testimonials or reviews to simulate genuine customer feedback and manipulate perceptions about a product or service. These spurious reviews distort product perceptions, misguiding consumers and influencing purchasing decisions.

- "Confirmshaming" describes the practice of using guilt to convince a user to select a course of action that they otherwise might not have taken.
- "Subscription trickery" on e-commerce platforms constitutes a dark pattern, employing deceptive tactics to enroll users in subscriptions without transparent disclosure. This covert approach compromises user autonomy, leading to unintended financial commitments and emphasizing the necessity for enhanced transparency in subscription models to foster ethical consumer interactions.

4 Solution Implemented

Our solution is to create a all purpose scalable solution "Naitik" (as in ethical, derived from Hindi language) which will mainly consist of 3 microservices hosted on cloud which will communicate with each other accordingly.

- **Chrome Extension** : This is a client-side component which the users will be interacting with. It interacts with the backend API and responds accordingly in the frontend.
- **Website** : A website pertaining to the project is deployed which serves two-fold purposes
  - A monitoring platform for admin/regulation body to look into the number of dark patterns report for each e-commerce website.
  - Feedback and report mechanism for the users regarding the dark patterns left unidentified by the web extension and any new dark patterns the users would like to report to. We then process this information to make our web extention better for the edge cases.
- **Backend Server** : This is a back-end component that receives requests from the client side and processes them using respected logic and algorithms implemented for each dark pattern. It retrieves data from the database and also communicates with the 3rd party services and large language models or our respected ml/dl model hosted under the same subnet inside the cloud service, be it aws/aks etc.
- **Third Party Services** : We refrain from reinventing the wheel and creating models and services which already exist. The condition of usage is limited to the services being open-source and having some sort of Api endpoint to recieve results.
- **AI Models** : Generative AI models and DL models are used extensively for detecting Dark patterns (both text pattern matching and visual analysis).

The three tier architecture enables us to:

- Improve scalability: Each layer can be developed, modified, and tested independently, allowing for better scalability. Upon finding a solution to a dark pattern, we can always plug in the logic and algorithm as well as the required UI.

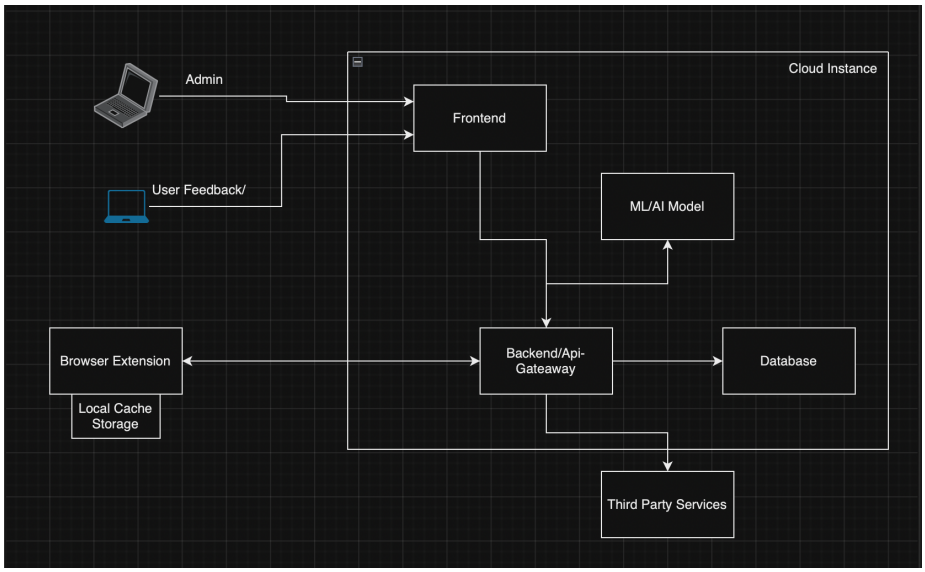


Figure 1: Architecture Implemented

- Enhanced performance: Because the layers are separate and work independently, the performance of the application is improved.
- Improved security: Users are not able to access the database directly, improving the overall security of the system.
- Improved maintainability: Changes to one layer are less likely to affect the other layers, making it easier to maintain the system.

4.0.1 Model Architecture and implementation

- Our DOM tree analyser takes a website's critical sentences containing various information and potential Dark Patterns. It then sends those collected sentences to our Model hosted on Amazon Sagemaker. This model is based on the Roberta 350 million parameter.
- We used Faster R-CNN for visual analysis to detect visual cues like stars, likes or loading animations. Then, pattern matching and colour and spatial analysis is performed on the text and non-text UI components.
- Finally, after all analysis is done, the Dark Pattern resolver generates whether or not the page contains dark patterns and classifies them. A vector Database by Weaviate is also used for clustering and retraining pipelines.

4.1 Architecture and Improvements Made

- We have finetuned Roberta on the Mathur dataset (and added more data and labels manually). It is trained such that it can perform multiclassification. It performs exceptionally well with an accuracy of 96%, and is also lightweight compared to other LLMs.

- We also use Faster CNN for UI and component detection and OCR API to segment various UI components. Further analysis is done, such as a colour histogram analysis. In this analysis process, we first calculate the grayscale histogram of the segment where we use two bins, one for the intensity values ranging from 0-127 and another one for the intensity values ranging from 128-255. If 65% of the pixels are above 128, then they are classified as a bright component or if they are below, then categorized as darker components. If it is normally distributed, then it is classified as normal.

- Similarly, spatial analysis is performed. To determine the neighbourhood area around a given segment, we apply a proximity factor to the segment boundaries. This factor, typically a small percentage (e.g., approximately 5%) of the segment size, is determined through empirical analysis. Segments whose boundaries intersect with the neighbourhood boundary are considered neighbours of the current segment under analysis.

- Once neighbours are identified, the next step involves computing each segment's relative width and height compared to those of its neighbours. This entails dividing each segment's width and height by the maximum width and height within the neighbourhood.

- In the final step, the Spatial Analysis process generates a JSON object containing information on neighbourhood coordinates, neighbouring segments, and their relative sizes (width and height).

- This data is sent in a JSON format to the DarkPattern Resolver, which uses this information to infer dark patterns. The confidence score from each analysis is taken and combined, and the final inference is drawn for the type of dark pattern.

4.1.1 Data Handling and Processing

We used the Mathur dataset and generated 200 more such examples rich with different dark patterns. We used this dataset to finetune the Roberta 350M model for multiclassification. The pipeline is such that it also accounts for new emerging dark patterns. If new dark patterns are reported, the webpage and its associated content are saved and transformed to a vector generated by the Distilbert transformer model and stored in Weaviate. This helps cluster unknown dark patterns automatically with the help of the neural networks and semantic store. These clusters are then sent to the model for further improvements in a batch. For the visual analysis, the dataset used is the ContextDP dataset, which is comprised of 175 mobile and 83 web UI screenshots containing 301 dark pattern instances.

5 Software Solutions

5.1 Checkbox Unchecker

Our extension not only takes a proactive stance against deceptive practices but also safeguards users from inadvertent agreement to unwanted terms. Through

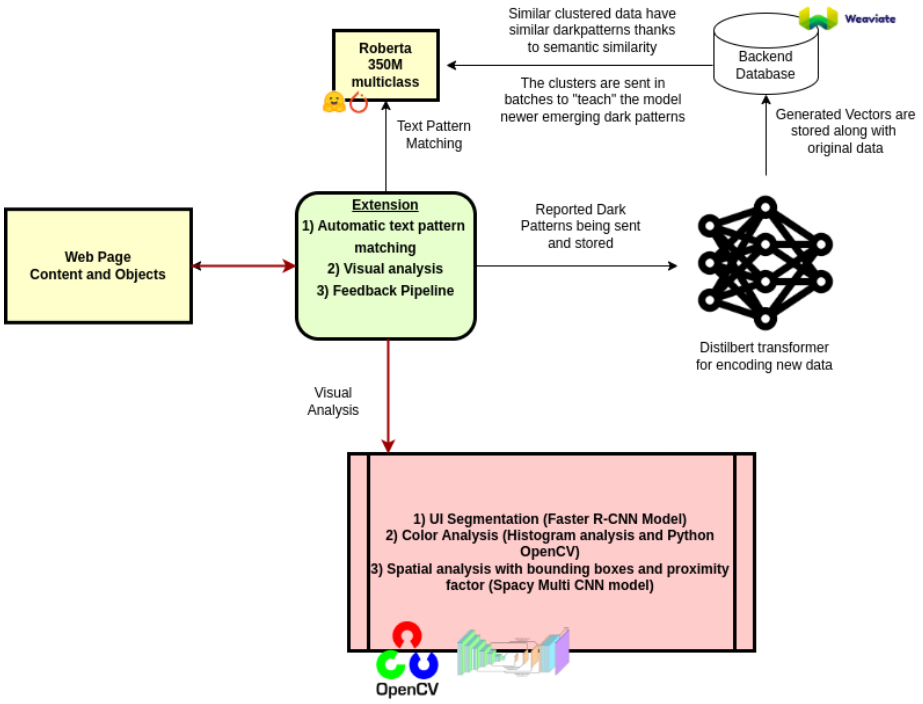


Figure 2: Generative AI Pipeline (complete)

meticulous DOM analysis, it intelligently identifies and unticks checkboxes, eliminating the risk of users unknowingly opting into undesirable agreements.

Adding an extra layer of user empowerment, the extension would promptly issue notifications whenever it intervenes to untick checkboxes. This real-time feedback ensures users remain fully cognizant of the extension's actions, fostering a transparent and trustworthy browsing experience.

Moreover, our extension would employ sophisticated algorithms to discern between checkboxes integral to genuine user interactions and those entwined with deceitful maneuvers. This meticulous approach ensures that only checkboxes aligned with users' true intentions are modified, mitigating any potential for unintended consequences.

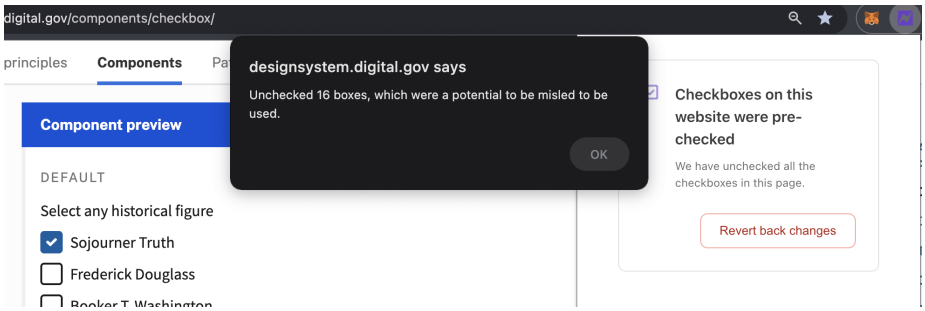


Figure 3: Checkbox Unchecker

5.2 ClickJacking

Clickjacking is an interface-based attack in which a user is tricked into clicking on actionable content on a hidden website by clicking on some other content in a decoy website. This is a relatively complex dark pattern constructed malicious adversaries trying to exfiltrate sensitive information. The technique depends upon the incorporation of an invisible, actionable web page (or multiple pages) containing a button or hidden link, say, within an iframe.

The impact of this sort of attack can be:

- Microphone or webcam activation
- Credential theft
- Unauthorized funds transfer
- Malware installation

and the list goes on...

The malicious adversaries utilise UI redressing to mislead users into clicking on illegitimate links (even if they seem legitimate) by cropping content, adding hidden overlays and rapid content replacement.

Preventing clickjacking attacks :

- Checking whether website employs X-Frame-Options which prevent websites' content being framed.
- Checking CSP policies of a website to ensure no illegitimate websites can embed them in malicious websites
- Using the feedback loop and a classifier model to predict the chances of a webpage being part of a clickjacking attack. This would also involve analysing the DOM to look for hidden buttons or actions.

Note: Additional measures can be taken by developers of the website by adding a CSRF token to all of their sensitive actions. (The extension can have a feature of notifying the owners or developers of the website about this dark-pattern/vulnerability)

5.3 Price Transparency Checker

Using a price transparency checker would enable users to effortlessly compare product prices across various websites. Using real-time web scraping techniques, the extension aggregates up-to-date pricing information, presenting it through an intuitive interface. Key components of the extension include customizable alerts, enabling users to set price thresholds for specific products and receive notifications when those thresholds are met. Additionally, the extension provides insights into historical price trends, empowering users to make informed decisions about optimal purchasing times.

Figure 5: Price checker

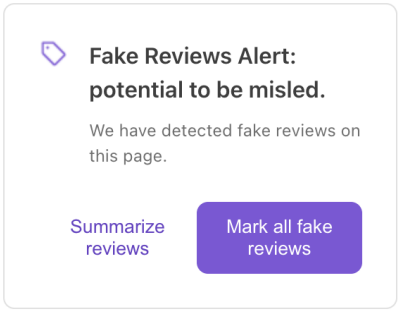


Figure 6: Fake Reviewer - 1

5.4 Fake Reviewer

Integrating a feature that verifies the authenticity of product or hotel reviews is crucial for dark pattern identification. This is important because fake reviews, often generated by AI, can mislead consumers, influence their decisions, and contribute to a distorted online reputation for businesses. Identifying and filtering out these fake reviews helps maintain the integrity of online platforms and ensures users can make informed choices based on genuine feedback.

- **Natural Language Processing** : NLP is used to identify use of AI generated reviews, which usually lack nuances and specific or personal experiences in reviews
- **AI Detection Models** : A lot of research is being done in the field of detection of LLM generated content. Many models have come up which utilise mathematical tools to classify text generated from LLMs. ConDA is

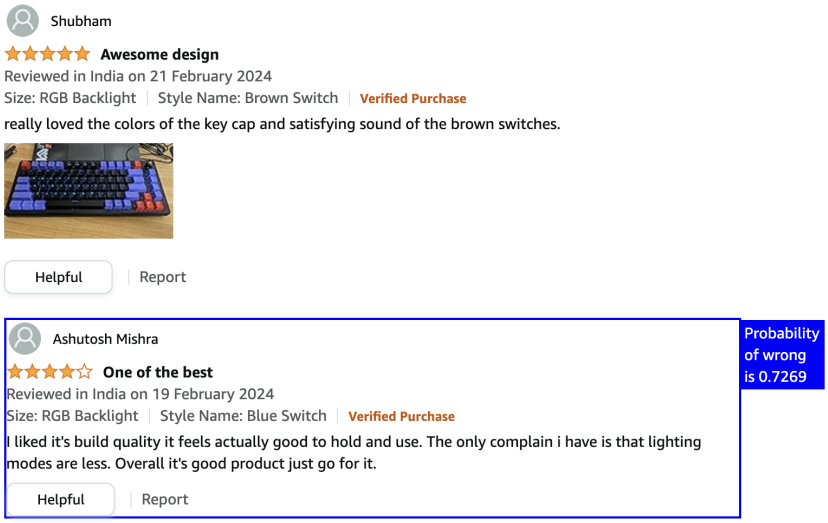


Figure 7: Fake Reviewer - 2

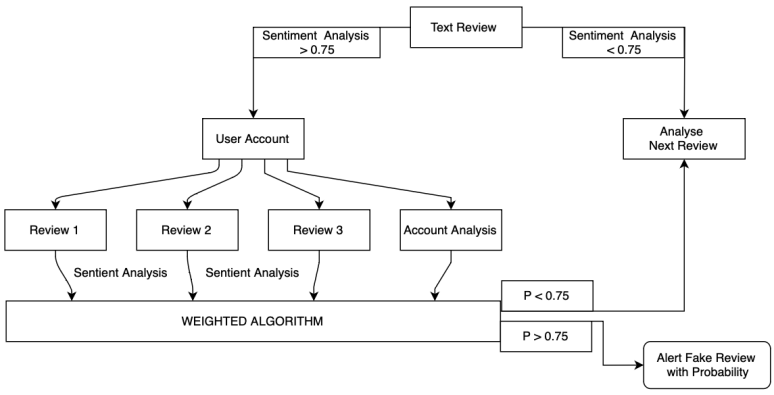


Figure 8: Fake Reviewer - Algo

one framework we came across our research. We did not use llm's because they are very generic and not having a false negative is more important than having a review marked as fake.

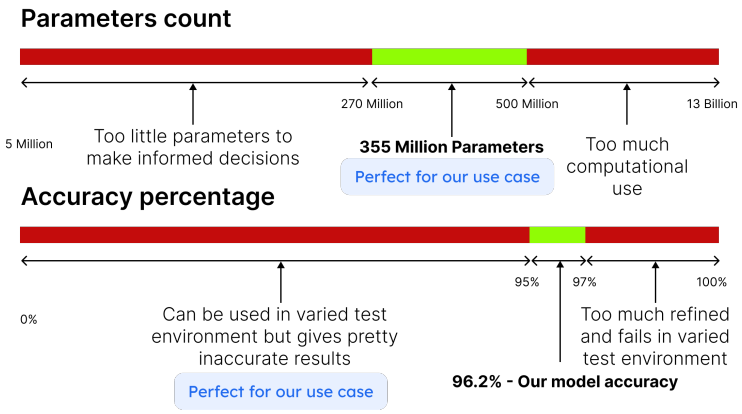


Figure 9: ML parameter counts and accuracy

- **Efficient Computation and Storage** : This feature would require the user to manually run as running this automatically on every webpage will be very computationally expensive. Also storing the classification of reviews of popular products and websites dynamically would further help in reducing computation costs.
- **Novel Algorithm** : We made a novel algorithm which gets inside user account and takes in various parameters to decide the fake review rather than directly commenting based on sentiment analysis. FLow can be seen in the picture above.

5.5 Detect Confirm Shaming

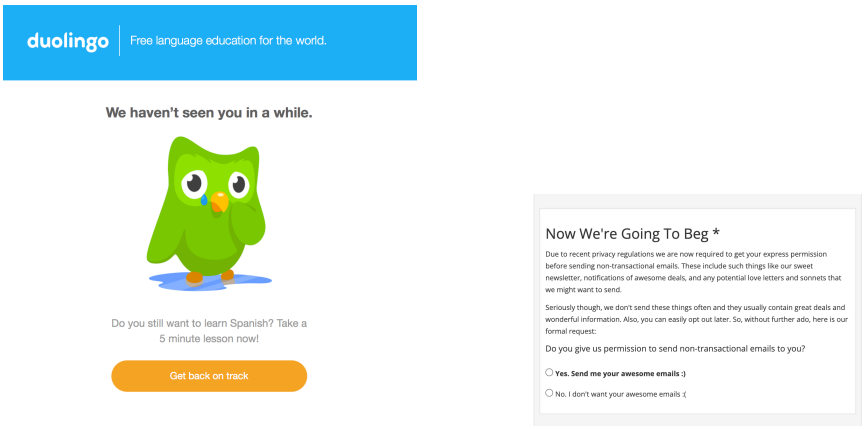
This is a language based attack in which a user is tricked using guilt shaming to either subscribe to a service or prevent them from unsubscribing.
The examples are given below

Such can be prevented using Artificial Intelligence technologies like natural language processing (NLP) to detect emotional messages. Upon this, people can be informed of the potential Confirm Shaming. We can also potentially hide with a message "**This image/email contains potential confirm shaming. Click to view.**"

5.6 Data protection

There are multiple ways to trick people into revealing more data than they originally intended to thus providing a threat to their privacy. A few ways that an enterprise might do it is.

- **Confusing Cookie Notices**: Some websites use cookie banners deliberately using misleading design and text. Deceptive patterns are especially



(a) Duolingo (b) Google form

Figure 10: Examples

common in cookie notices partially because there is no regulatory guidance on how one should be crafted. An example of this is given below. This doesn't give an obvious close popup option and specially highlights the accept cookies.

- **Confusing Language**: Sometimes the language used for popups and privacy policies is extremely confusing and ambiguous, and extremely long. On paper, it might not seem a big issue but this causes enough friction for users to agree to terms and conditions/accept cookies without giving it a read.

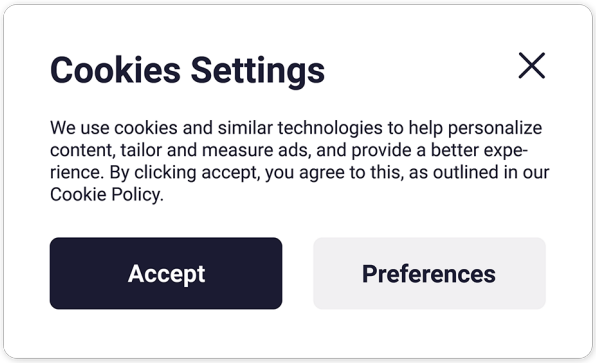


Figure 11: It just shows accept and preferences but a small close button. They are playing with the hierarchy to make it seem less obvious

The solution for this is to use a pre-existing AI tools to provide a summary of the privacy policies to inform the users of potential data leaks. Also we can provide an external close popup button. This can be implemented by detecting and getting rid of the cookies div. We can also give users an option to block all

popups and thus rejecting all potential cookie permission dialogues.

5.7 Subscription trickery

This is when OTT services use dark patterns to prevent/trick users into subscribing to their services and trick them into paying more for unnecessary services.
An example is the ongoing Amazon Prime lawsuit the Federal Trade Commission filed. Under this, several subscription-based dark patterns have come up, like having users start prime membership with a single click but have them go through pages to cancel their subscription.
The FTC also claims that Amazon tricked customers into purchasing more expensive full Prime memberships instead of the cheaper Prime Video subscriptions. Amazon's website is designed to mislead customers into choosing the costlier Prime membership instead of the cheaper Prime Video option. The complaint states that colored buttons on the website encourage and redirect customers towards signing up for full Prime membership.

The dark pattern in the cancellation process causes several users to procrastinate the cancellation of service until it's too late, and you have to continue the service for another year/month. We can use web scrappers to keep track of upcoming subscription renewal dates and keep notifying/reminding users to cancel their subscriptions.

6 Tech stack used

7 UI/UX Flow

See figure 4 for the initial proposed UI flow for the chrome extension.

7.1 False Urgency

One of the most common ways of inducing the sense of false urgency particularly in e-commerce space is through advertising the product price as a limited time offer. This is usually shown in UI as "Limited time offer" or "Special price sale".

We decided to tackle this issue through the method of keeping track of the historical price data across various platforms. This first-of-all provides a clear idea about any significant fluctuation in the price on the platform itself and secondly gives us a broader image of the historical price movement across various platforms.

Comparison across various platform is necessary because there could be a case where the native site might rise the price of the product way above the market price and then dip down the price to justify a "Special price" offer. Although it is technically a special price offer but in practicality it is just another decoy of the "False Urgency" pattern.

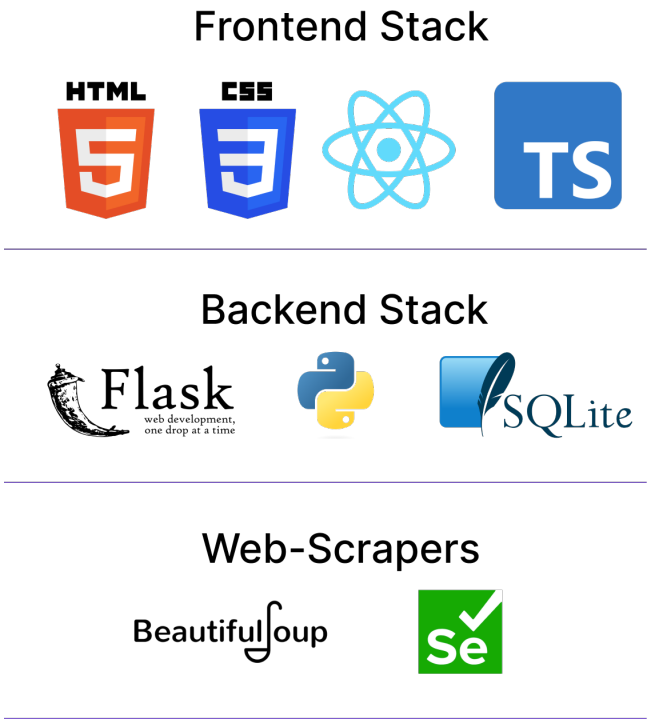


Figure 12: List of tech stack we have employed to build our solution

7.2 Disguised Advertisement

Other dark pattern often deployed in the e-commerce space is the use of disguised advertisement. This is done usually through the use of fake reviews.

Fake reviews can be identified through various ways. These reviews are identifiable and follows a particular pattern. Some of these are -

- A very high percentage of five-star reviews
- Lack of detail in reviews and vague praise
- Generic review titles like “Nice product” or simply “Awesome”
- Mentions of competing products
- Wording similar to other reviews
- Poor grammar and spelling mistakes
- Multiple reviews on specific dates (especially if there are long gaps between them)
- “Customers also bought” section contains unrelated products
- Glowing reviews with one small negative that isn’t a deal breaker

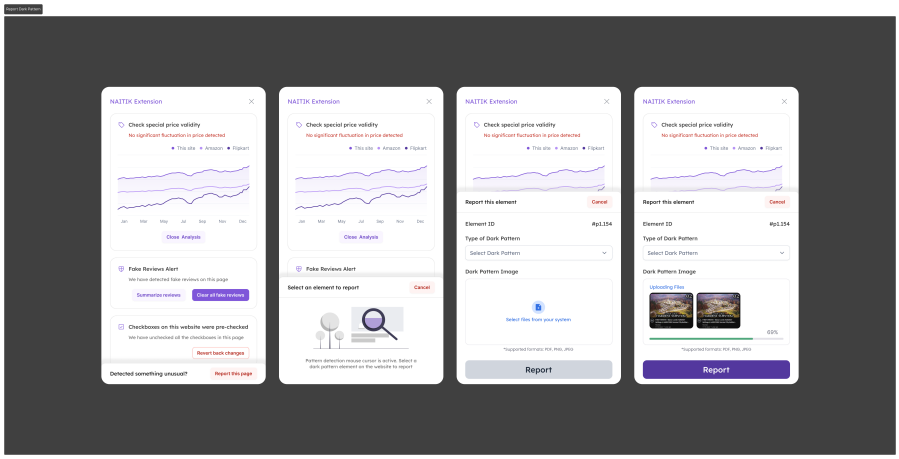


Figure 13: Extension UI Screens

We can develop a language-based model to detect this. Once we figure out that there are fake reviews oh the platform we can offer two functionalities. The first one is to remove all the fake reviews from the product page or we can offer another option of summarising all the original and valid reviews. This can be easily done by a language trained LLM model.

In case there are reviews that the user identifies as fake but are not filtered out by our model. The user can select such reviews and report them to us. If those are really fake reviews then they are flagged as fake for the future users. This user flow is explained in detailed in the upcoming section of **Feedback Loop**

7.3 Basket Sneaking

Often people are sub-consciously trapped to perform actions that they might not originally wanted to do so.

Such as paying extra for a service/product that is automatically included in the checkout list. This is often done through the usage of pre-ticked boxes in a web page.

We figured out one way to do so is to automatically untick all the pre-ticked boxes. The user don't need to perform any actions to let this happen. But the user is identified that the boxes are unchecked. If the user want to revert back the changes then he can easily do so.

7.4 Confirm Shaming

Just as the basket sneaking, our language-based algorithm automatically identifies any phrases or words that target the sensitivity of the user such that forcing them to perform an action; usually through a demeaning and a shameful way.

Once our algorithm figure out that there is confirm shaming employed in the web page we automatically remove/change it. But the user is identified that we have removed/changed such phrases and the user has the autonomy to revert back the changes.

Other side scenario could be that our algorithm misses such phrases. In that case the user can report such incident and make the experience better for future use case. This is again explained in detailed in the upcoming **Feedback Loop** section.

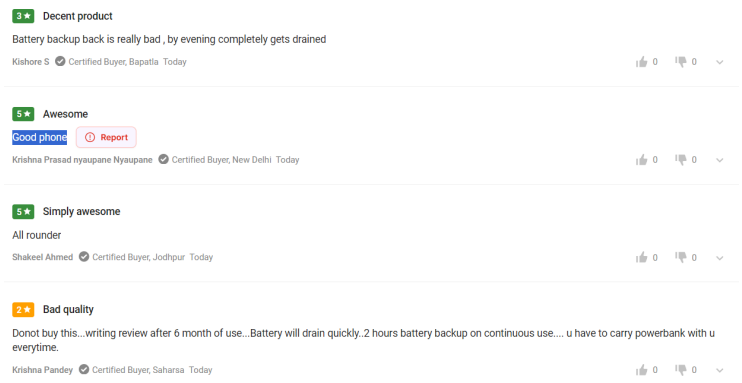


Figure 14: Pattern identified by the user

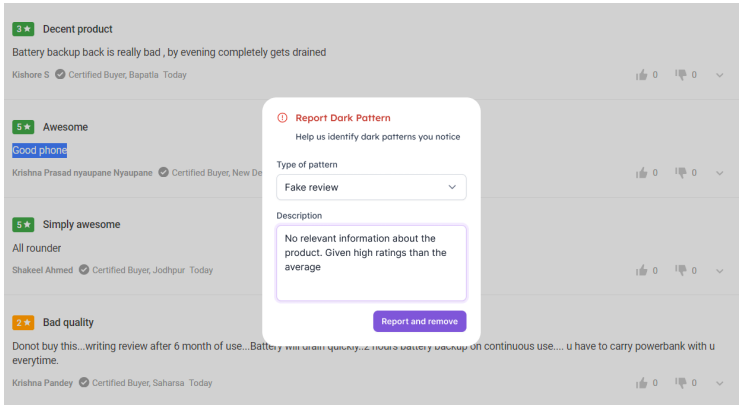


Figure 15: Pattern reported by the user

8 Feedback Loop

As shown in the figure 5 above, upon selecting a review, a report button pops up for the user, allowing him to click it, if he feels the review is misleading. Clicking it leads to a pop-up screen(figure 6), which has a drop down menu in options to select the type of dark pattern to be reported and the description along with it. Similarly we can extend this model to other dark patterns such as price fluctuation upon which user can report the dark patterns which our extension might have missed.
This helps our model to identify more dark patterns and take in user input to tweak our web extension to be more accurate.

Now improving upon this feedback model, we plan to create a website on which we will add a feedback option/form review, where users can report dark patterns observed on any e-commerce website. We will then check the validity of the the report and add the functionality if needed on our web extension.
We also plan to add an admin panel where we can see how many dark patterns are detected on which websites and their types as well. This will help the re-

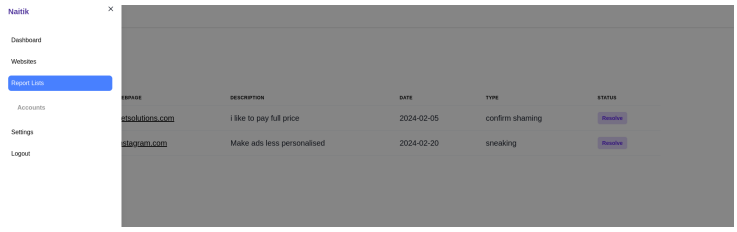


Figure 16: List out all the reports

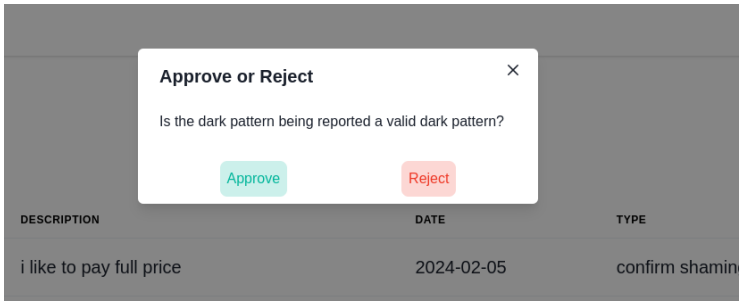


Figure 17: Approve or Reject

spective authorities to take appropriate actions if needed and also provide us with a database for the e-commerce platforms.

9 Monitoring

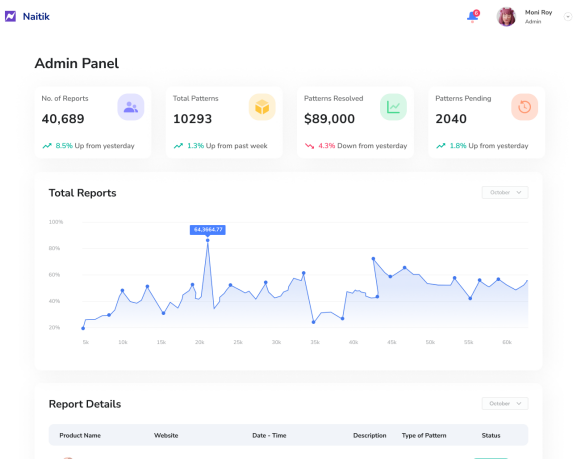


Figure 18: Admin panel home screen

We have a monitoring system to see the current situation of each and every website that contains dark patterns. We can not only see the dark patterns detected by our ML model but also the patterns that are reported by the users.

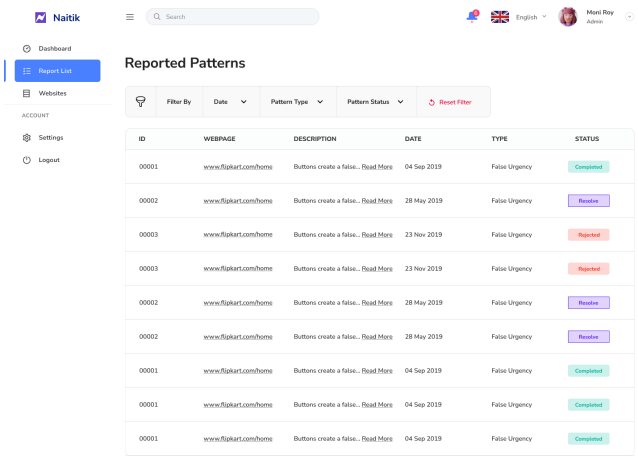


Figure 19: Reported Patterns screen on the admin panel

Without the report functionality, any potential dark pattern buster is useless. As history is evident, the moment any software is developed that can report any malicious attack, the attacker gets smarter and finds ways to avoid. Thus, most project would reach a position where it would become useless, and by the time users realise it, it might be too late. But not with Naitik. The report functionality is important in the sense that our model might miss out some dark patterns that might be novel in approach. This not only helps us to find out dark patterns to much better extent, but this also acts as a good database to train our model. Making our model more resilient and dynamic at combating the dark patterns.

As an admin, we will have the power to cross check the validity of the dark patterns. We can approve and reject the reported patterns.

10 References

- https://www.ftc.gov/legal-library/browse/cases-proceedings/1910129-1910130-amazoncom-inc-amazon-ecommerce
- https://portswigger.net/web-security/clickjacking
- https://doi.org/10.48550/arXiv.2111.15242
- https://jyx.jyu.fi/bitstream/handle/123456789/72034/URN:NBN:fi:jyu-202010066090.pdf;sequence=1
- https://blog.crobox.com/article/dark-patterns