

Forta Infinity - Project: BotWeiser

PRDv1

Abel Atnafu, Junhwan Lee, Taeho Kim, Kenji Kano, Vikram Elango

1. Intro

Forta is a decentralized platform, safeguarding the expansive and intricate world of Web3. Its mission is to provide security and operational monitoring for blockchain activity, acting as a frontline defense against potential threats. Forta's ecosystem revolves around a network of autonomous bots that diligently scan transactions and identify threats and anomalies in real-time, ensuring timely alerts. Forta Network is the largest network of security intel in Web3, leveraging machine learning to detect exploits, scams, and other threats. It's an open-source project where a community of developers and security researchers build ML-based threat detection bots to detect hacks and exploits.

In the context of the Forta Network, where numerous bots serve various purposes, such as scam detection, exploit identification, and operational monitoring, users face a significant challenge in locating the most suitable bot or feed for their specific needs. This challenge hinders efficient utilization of the available resources and can lead to suboptimal bot selections. To address this issue, our project aims to develop an intelligent feed and bot finder powered by Large Language Models (LLMs). This intelligent system will analyze the descriptions, source code, and sample alerts associated with each detection bot deployed on the Forta Network. It will then take the user's problem space as input and provide personalized recommendations for a

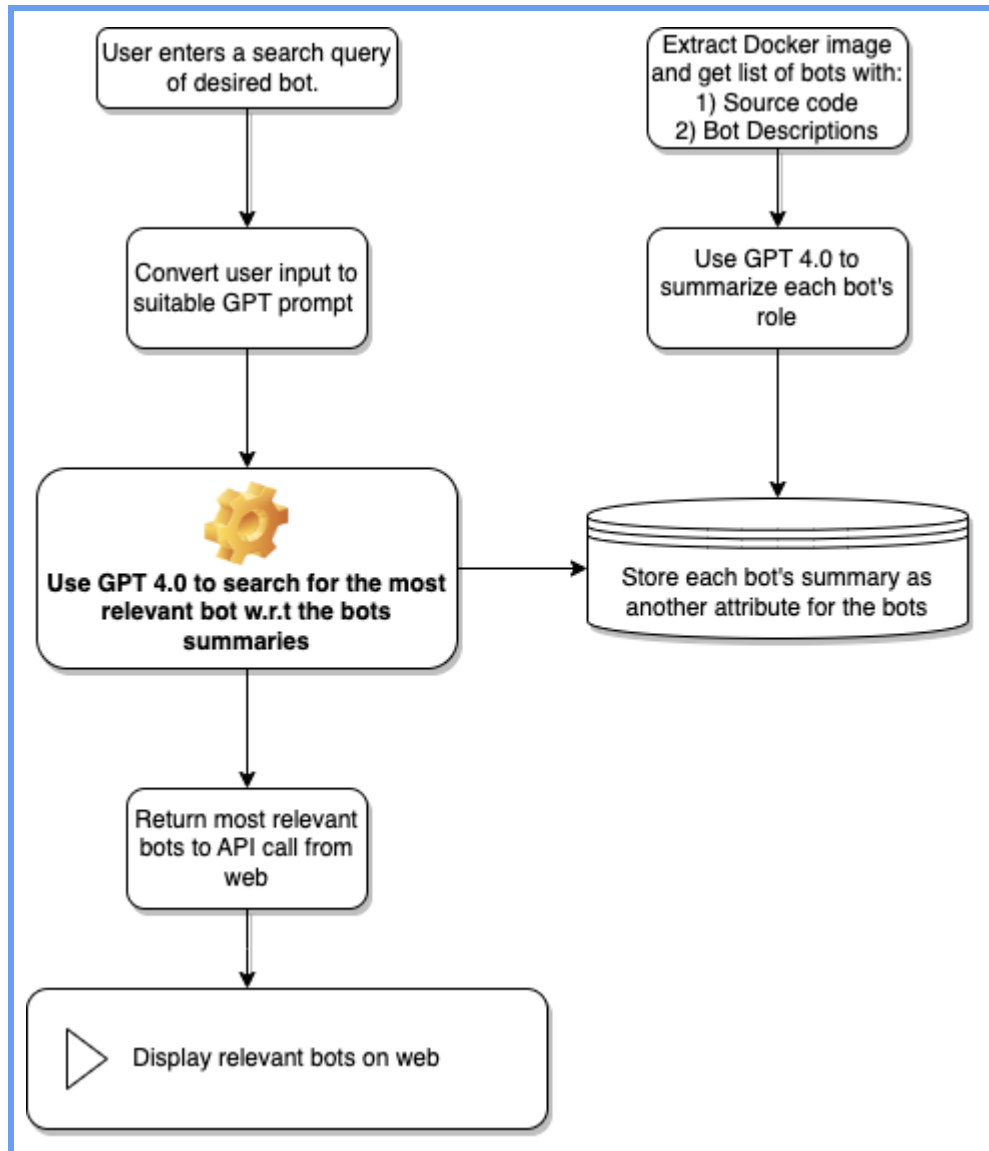
specific bot or feed, along with a rationale for the recommendation. The goal is to simplify and streamline the process of selecting and utilizing bots and feeds within the Forta Network, enhancing user experience and optimizing resource utilization. This solution should be easily accessible to Forta Network users, ensuring that they can make informed choices and efficiently address their specific needs. The search tool currently used by Forta can only search for Bot Name, Bot ID and Developer ID. This means that users need to know in advance the name of the bot they are looking for, since they cannot search for a description of the bot. We are planning to implement LLM in search tools or chatbots so that users can find the bot they are looking for by referring to the bot's description field when searching.

The current approach to bot recommendation within the Forta Network relies on keyword matching against bot titles, lacking semantic understanding and personalization. This limits its ability to accurately capture user intent and potentially overlooks relevant bots without exact keyword matches. Additionally, it doesn't provide users with explanations for recommended choices. To address these shortcomings, the development of an intelligent feed and bot finder using LLMs aims to enhance the Forta Network's usability by offering context-aware, personalized recommendations, improving the overall user experience.

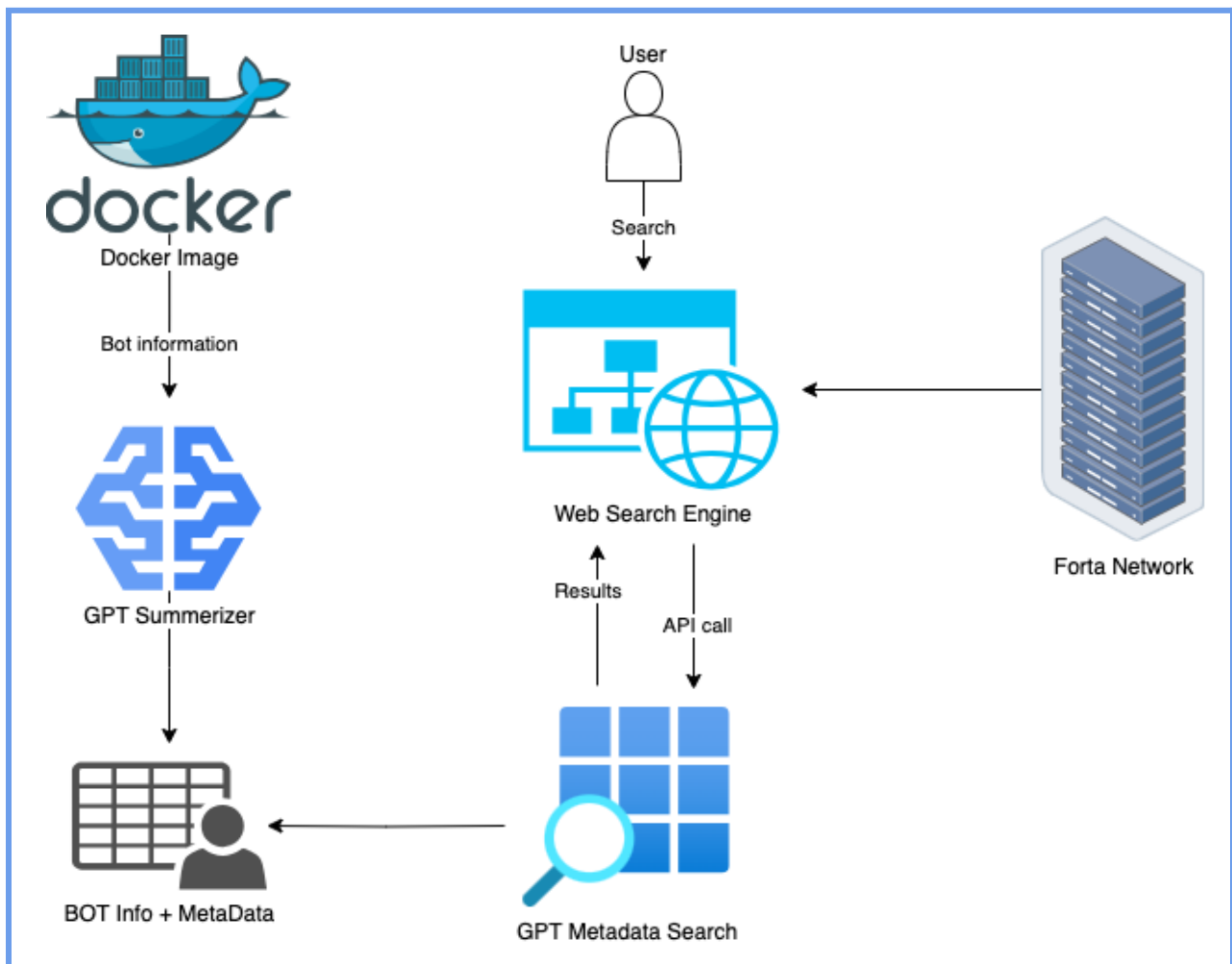
Our main goal is to design a website that's easy for people to use on Forta.org. We're aiming to create a working model of a web application that can find the right bot when users search for them. Additionally, it will explain why a particular bot was suggested and how it can be useful, giving users a clear understanding. By doing this, we want to make it simpler and more enjoyable for users to navigate Forta Network and pick the right tools for their needs.

2. System Architecture Overview

1. High Level Design Flow



2. High Level Diagram



3. User Stories

1. As a user, I can enter the descriptions of the detection bot so that I can locate the bot without having to know the name of it.

Github Issue: <https://github.com/junhwanlee2316/ForTa-BotWeiser/issues/1>

- Acceptance Criteria: Users can converse with the chatbot which will guide to the list of bots arranged by the chatbot as well as the reasons for recommending these bots and why they are the best fit.

2. As a user, I can sort the detection bots by popularity or date after the chatbot directs me to the list of bots so that I can choose a detection bot depending on my needs.

Github Issue : <https://github.com/junhwanlee2316/ForTa-BotWeiser/issues/2>

- Acceptance Criteria: After the chatbot directs the user to the list of recommended bots, the user has an option to arrange them by popularity or date.

3. As a user, I can be prompted to revise the description when I accidentally input an incorrect description or make a typo, so that I don't need to manually return to chatbot to enter the description again.

Github Issue: <https://github.com/junhwanlee2316/ForTa-BotWeiser/issues/3>

- Acceptance Criteria: When the user's entry doesn't align with any of the bot summaries, chatbot asks the user to re-enter the description instead of directing the user to the list with no/wrong bots.
- Scenario 1: The user enters a relevant description, and the chatbot directs them to the appropriate list of bots.
- Scenario 2: The user enters an irrelevant description, and the chatbot warns the user that no bots are found and asks to re-enter.

4. As a user, I want to be informed when my description doesn't match any of the available bots so that I can refine my searches instead of recommending wrong bots.

Github Issue: <https://github.com/junhwanlee2316/ForTa-BotWeiser/issues/4>

- **Acceptance Criteria:** On entering an irrelevant or unmatched description, the bot notifies the user of an unsuccessful match and provides guidance through suggesting potential corrections or prompting re-entry.
- **Scenario 1:** User Types in “blockchain tennis bot” into the search bar. Bot doesn't find any bots that match this description. Bot sends message: “Sorry, we couldn't find any bots matching “blockchain tennis bot ”, consider refining your search.

5. As a user, if I make a typo I want my typo to be corrected and the search results should be shown accordingly.

Github Issue: <https://github.com/junhwanlee2316/ForTa-BotWeiser/issues/5>

- **Acceptance Criteria:** The bot should auto-correct detected typos and display results based on corrected query. The system should inform the user of the correction made, offering an option to revert to the original query if the correction is inaccurate.
- **Scenario 1:** User types “blockchain bot”
User: User sees a notification above the results saying, “Did you mean ‘blockchain bot?’”. [Showing results for ‘blockchain bot’]. “Search for ‘blockchain bot’ instead” with an option to revert back to the original search.

6. As a user, I want a chatbot to initiate our interaction with a greeting message, allowing me to recognize it as a chatbot before I make any input.

Acceptance Criteria: Upon the user's initial interaction with the chatbot, the chatbot initiates the conversation with a greeting message, such as "Hi there! I'm [name of bot], here to assist you in finding your detection bot. Please provide me with a description of the bot you're looking for!" This ensures users understand how to utilize the chatbot.

7. As a user, I would like the chatbot to inform me when it is in the process of preparing a response using visual cues such as "..." icons, allowing me to understand that it's currently formulating the response.

Acceptance Criteria: When users input a query and the backend is in the process of generating a response, the chatbot displays a visual cue within the chat interface in order to ensure that users know the chatbot is actively formulating the response.

8. As a user, if I am not sure which bot is appropriate for my situation, I want the chatBot to recommend appropriate bots by explaining my situation.

Acceptance Criteria: User types their situation into chatBot, the chatBot generates a solution based on description of Forta bots. The system can show the bots in order of recommendation.

9. As a user, I can decide not to use the chatbot and instead navigate myself to the list of all the detection bots if I wanted to.

Acceptance Criteria: If a user prefers accessing the whole list of bots, they can choose to do so rather than being automatically directed to the chatbot.

10. As a user, I can ask about the result that the chatbot gave me and it should give me detailed information about various questions I ask about.

Acceptance Criteria: User asks about the recommended blockchain bots, the chatBot should generate a suitable and friendly response explaining the question. The response should build up from the explanation given initially when recommending the bot and should include more details answering user questions.

4. Appendix

- Backend: Python
- Frontend: React.js
- DevOps/Tools: Docker
- Artificial Intelligence Tool: GPT 4.0
- Data Formats: .csv