**Feature requests to analyze – Ambiguity Detection in Feature Requests**

**Task Description and Tips:**

Below you will see 5 different feature requests that are pulled from Mastodon’s GitHub issues section. A feature request consists of a request number, title of the request, and a description of the feature request. Your task is to analyze the feature requests and give your critique to your understanding of whether the feature requests are ambiguous or not. If they are ambiguous, please state what type of ambiguity exists in the text. You do not need to suggest recommendation for the ambiguity (only detect and classify the ambiguity). When completed please save the file with featureRequests\_[Your Name] and send it to discord. For reference 2 links are attached in this document too (See details in **Reference Links**).

**Reference Links:**

**Dan Berry Handbook about ambiguity (Also check GitHub pinned message)**   
- You can find the definitions for ambiguity and the types of ambiguity in this handbook. We will be following this book for discussions.

<https://cs.uwaterloo.ca/~dberry/handbook/ambiguityHandbook.pdf>

**A sample format from ChatGPT (\*This is only a reference on how the format can look like\*)**- This is only a sample of how the critique can look like. Please feel free to change the format and write in your own way. Make sure to capture as much of the ambiguity there exists in the request.

<https://chat.openai.com/share/2f7629ca-556e-4ff0-bde4-c81869114efe>

**Feature requests to analyze:**

1. **Request No: 70  
   Title**: Reloading gets you to latest toot, losing hability to read oldest unread first  
   **Description**: Maybe this is a feature, not a bug of the app, but certainly it is what keeps me using Fedilab for now. With Fedilab, the loading of new toots keeps my current position on the timeline, so I can read things in their natural order. In official Mastodon app, as it takes you to the top, final toot, it forces you to scroll back down to whatever was the last toot you saw.

\*Lexical Ambiguities:

"toot" -> may not be universally understood, especially by those unfamiliar with Mastodon or similar social platforms. (Could be confusing to those unfamiliar with the terminology.). **\*As Mastodon developer it is fine.**

Semantic Ambiguities:

"losing hability to read oldest unread first" -> It could mean the user wants to start reading from the oldest unread toot or that the current behavior prevents them from doing so efficiently. (Need for the developer)

Pragmatic Ambiguities:

(Logic) new feature should mimic Fedilab's behavior, but this is not explicitly stated -> How? Needs detail for the developer.

Syntactic Ambiguities:

Comparison between Fedilab and the official Mastodon app could be more explicitly broken down to avoid misinterpretation.

1. **Request No: 160  
   Title**: Feature: Group notifications by toot  
   **Description**: In the notifications tab, the app tells you that “@realpixelcodeboosted your post” and displays a preview of the toot beneath. Now, when a toot becomes popular and gets quite a few favourites and boosts, it quickly becomes a mess.Instead, my proposal is to group favourite and boost notifications by the respective toot, so the notification would instead look like this:Your post has received 14 boosts from:user1user2user3...[toot preview]

Lexical Ambiguities:

\*Similar to the previous request -> “toot” (No issues for the developer – nothing new)

Semantic Ambiguities:

"group notifications by toot" -> Does it mean grouping all types of notifications (favourites, boosts, replies) together per toot, or just grouping favourites and boosts separately? \*\* It is in description

Pragmatic Ambiguities:

The user's expectation is that grouping notifications by toot will simplify and declutter the notifications tab -> it is not explicitly stated how this should be handled when multiple types of interactions (favourites, boosts, replies) are present for a single toot. (Logic error)

Syntactic Ambiguities:

N/a

1. **Request No: 72  
   Title**: Unable to access hashtags with talkback  
   **Description**: When using talkback there is no way to access the hashtag links in a post Edit 1: i stand corrected. In a post with only one hashtag it appears possible to access it by long pressing the post, however this is unintuitive. In posts with more than one hashtag that trick doesnt work either Edit 2: i feel tusky has a good way of fixing this by adding links and hashtags to the talkback local context menu

Lexical Ambiguities:

"TalkBack" -> clear for android developer -> Android devices

Semantic Ambiguities:

"access the hashtag links in a post" -> might mean being able to navigate to or interact with the hashtags using TalkBack, or it could mean being able to activate the hashtags to see related posts.  
"local context menu" in "adding links and hashtags to the TalkBack local context menu" might not be clear to all developers. It needs a specific definition within the context of TalkBack.

Pragmatic Ambiguities:

The user mentions that the current method of accessing a single hashtag by long pressing the post is "unintuitive." The specific expectations for what would be considered "intuitive" are not clearly defined.

The reference to how "Tusky" handles this issue provides some direction but might require the developers to look into Tusky's implementation to fully understand the user's preference.

Syntactic Ambiguities:

N/a

1. **Request No: 115  
   Title**: Feature: Optional indicator for bots  
   **Description**: An optional setting to show who is marked as a bot on Mastodon

**Lexical Ambiguities:**

The term "bots" (marking an account as a bot might vary.)

**Semantic Ambiguities:**

"optional setting to show who is marked as a bot"-> Does it mean a visual indicator next to the bot's name, a separate list of bot accounts, or something else?

**Pragmatic Ambiguities:**

The purpose of marking bots is not explicitly stated. Is it for transparency, user awareness, or some other reason? Understanding the user's intention can help design the feature more effectively.

**Syntactic Ambiguities:**

Needs details

1. **Request No:** 50 **Title**: Feature: Display profile notes  
   **Description**: In the Web UI one can add notes to other profiles. It should be possible to view and edit them in the app too.

**Lexical Ambiguities:**

"profile notes" -> needs to be explicitly understood as personal annotations a user can add to other profiles.

**Semantic Ambiguities:**

"view and edit them in the app too" -> It might mean viewing notes in a specific section of the app, having the notes appear on profile pages, or accessing them through another interface within the app.

**Pragmatic Ambiguities:**

The user's implicit expectation is that the notes feature in the app should work similarly to the Web UI. However, details on how the user interacts with these notes in the Web UI and how they envision the same functionality in the app are not provided.

**Syntactic Ambiguities:**

N/a