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

<https://github.com/Pragyan10/WebScrapperForGithubIssues>

**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:**

* **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.

\*\*\*Only analyzed the description, but I can see few other problems with the title\*\*\*

\*\*\*It helps trying to convert the sentence to temporal logic\*\*\*

Note: The main question is whether the following linguistic ambiguities would affect the design and implementation of the requested feature. In this case, I would say the detected ambiguities would not affect the design and implementation. However, it would be useful to summarize the request and verify the summary from the user. The summary then can be helpful in translating the NL text to LTL.

Note: We will add the terms and jargon that are domain-related, such as toot and Fedilab, and we list them as pragmatic ambiguity because they are context-related.

Lexical:

* + “Reloading” can mean different things. Either close the app and re-open or refresh. What do you mean by reloading? Can you specify the action for reloading? -> Impact the replication, design, and implementation
* Process-vs.-product: the loading of new toots keeps my current position on the timeline,
* Toot: multiple meanings, for resolution, we need domain knowledge

Pragmatic:

* “Reloading” [] missing a word (ellipses), Either close the app and re-open or refresh. What do you mean by reloading? Can you specify the action for reloading? -> Impact the replication, design, and implementation
* Deictic ambiguity: “Current position” on the timeline -> It will be important to replicate and understand what the user means by current position.
* What is Fedilab? Domain knowledge
* Deictic ambiguity: In official Mastodon app, as it takes you to the top -> It will be important to replicate and understand what the user means
* Elipsis: In official Mastodon app, as it takes you to the top -> It will be important to replicate and understand what the user means
* Fedilab: Domain knowledge
* Pragmatic ambiguity: The request requires knowledge about Felilab, which is a multi-accounts client for Mastodon, Pleroma, Friendica and Pixelfed. However, the analyst or the developer should have such knowledge.
* Deictic ambiguity: Final toot, and latest toot (what does it mean?) they are related to the time and location of the toot. -> It will be important to replicate and understand what the user means

Semantic:

* Referential: In official Mastodon app, as it takes you to the top
  + Referential: , it forces you to scroll back down to whatever was the last toot you saw.

Vagueness:

* Things
* **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:

* + Becomes a “mess” [what does it mean?]
  + Boost : Domain knowledge
  + Favorites: Domain knowledge

Pragmatic Ambiguity:

* + Elipsis and Deictic : “displays a preview of the toot beneath” [missing]
  + “my proposal is to group favourite and boost notifications by the respective toot”, does it actually mean the position “by” the actucal toot rather than beneath the toot? The pragmatic ambiguity will affect the design and implementation of the feature request.

Vagueness:

* + “a toot becomes popular and gets quite a few favourites and boosts” how would we evaluate a few and popular? The pragmatic ambiguity will affect the design and implementation of the feature request.

Semantic Ambiguity:

* + gets quite a few favourites and boosts
  + Referential: it quickly becomes a mess [What is it referring to?]
  + Elipsis: my proposal is to “group favourite and boost notifications” [favorite what?]
  + The semantic ambiguity in this example is important for us to clarify since it affects the design and development. It is also related to incompleteness.
* **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

Note: To design and implement this feature, the requester should provide additional information. At this point, it is not clear how this feature should be implemented. Therefore, this request is incomplete.

Pragmatic ambiguity: understanding the last part requires additional context about Tusky & talkback, which is a lightweight client for Mastodon, a free and open-source social network server. It supports all Mastodon features, like photos, videos, lists, ect. This is the domain knowledge.

Vagueness: “Long” pressing the post. How long is long? This feature cannot be evaluated. This vagueness will not affect the design and implementation of the requested feature.

Semantic Ambiguity: The phrase "access the hashtag links in a post" could be interpreted in different ways. It's not entirely clear what "access" entails—whether it means clicking on the links, navigating to them, or something else. Additionally, it's ambiguous what "hashtag links" refers to—whether it means links to hashtags or links within hashtags.

Syntactic Ambiguity: The phrase "In a post with only one hashtag it appears possible to access it by long pressing the post" is somewhat ambiguous. It's not entirely clear whether "it" refers to the post, the hashtag, or the action of accessing the hashtag.

Referential Ambiguity: The pronoun "this" in "however this is unintuitive" lacks a clear antecedent. It's not immediately clear what "this" refers to without more context.

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

Note: The linguistic ambiguities identified in this feature request will not affect its design and implementation. What is important here is the incompleteness of the requested feature. How does the requester want to show the mark on the profile?

Semantic Ambiguity: The phrase "who is marked as a bot" could be interpreted in different ways. It's not entirely clear whether "marked as a bot" refers to users self-identifying as bots, users being identified by others as bots, or users being automatically identified by the platform as bots based on certain criteria. However, this ambiguity will not affect the design and implementation of the requested feature since the developer or analyst knows who is referring to the profile based on the context of the app.

Syntactic Ambiguity: The phrase "An optional setting to show" is somewhat ambiguous. It's not clear whether the setting itself is optional or whether it controls something optional. Additionally, it's unclear what exactly is being shown—is it the fact that a user is marked as a bot, or some information about their bot status? However, this syntactic ambiguity will not affect the design and implementation of the app.

Referential Ambiguity: The pronoun "who" in "who is marked as a bot" lacks a clear antecedent. It's not immediately clear what "who" refers to without more context. It could refer to individual users, profiles, or some other entity. However, this ambiguity will not affect the design and implementation of the requested feature since the developer or analyst knows who is referring to the profile based on the context of the app.

* **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.

Note: When the requester provides a reference to the mobile or web version of the same app, or if the requester is referring to another app as a reference, the chatbot should ask the requester to explain the details of how the action is performed. This might be seen as part of the completeness of the feature request.

Syntactic Ambiguity: The phrase "It should be possible to view and edit them in the app too" can be parsed in multiple ways. It's unclear whether "them" refers to the notes, the profiles, or some other element. Additionally, it's ambiguous whether "view and edit" applies to both actions (viewing and editing) collectively or to each action individually. However, this will not affect the design and implementation of the software.

Referential Ambiguity: The pronoun "it" in "It should be possible" lacks a clear antecedent. It's not immediately clear what "it" refers to without more context. It could refer to a feature, capability, or action within the app. However, this will not affect the design and implementation of the software.