Technical assessment

Backend Developer at PublicSonar

1. Programing exercise

Publicsonar app collects millions of social media messages every day in bulk. Customers create “cases” where they can add their queries. After the case is set up, it starts receiving the messages that match those queries. Imagine you need to create a service that handles such matching tasks. This service receives messages as inputs and assigns them to the correct customer case.

Assumptions:

- For the purposes of the exercise assume you have a list of input messages (see example input messages)
- The service has access to a list of all cases (see example cases file)
- One message can match to zero, one or multiple cases
- The service outputs to another list the messages matched to the respective case or cases.

User queries contain terms and can handle multiple operators to combine them. Matching on terms follows these rules:

- **term1 AND term2 -** This will return only results that contain term1 and term2 in the same message, regardless of placement.
- **term1 OR term2 -** This will return messages that contain term1 or term2, or both.
- **(term1 OR term2) AND term3 -** The system allows for grouping using brackets. This will match all messages that contain term1 or term2 or both, and always term3.
- Terms are case insensitive, so **Soccer** matches **soccer**
- A term matches only if it exactly matches a whole word. So **soc** and **söccer** don't match **soccer**

Given these requirements and the example input, please create the code that outputs matches to another file or list.

2. System Design

What would be a good system design (high level) to handle incoming messages? The system would need to handle periods with high bursts of messages (10k m/s) and periods with no messages at all (during the night for instance). Messages are collected from multiple social media sources at different (random) intervals and are matched to cases. We discard the messages that don't match to any case. Different users might see the same message if they have similar queries. Messages are delivered in realtime to the frontend. Also important to mention that we have multiple data enrichment pipelines like language recognition, text summarization, sentiment analysis, etc

We expect to see a high level diagram explaining the flow of data. No need to expand too much on the data structures.

Feel free to make your assumptions, there's no single right answer. **We will value the thought process more than the end result.**

In the next meeting we will ask you questions about these exercises.

Good luck with the assignment!