Designing an Alert Correlation Engine using Mutual Information values

# Project Progress – Dated: Nov 14th, 2021

Below is how we are progressing towards our tasks in the project.

1. Which tasks have been completed?

|  |  |  |
| --- | --- | --- |
| **Tasks** | **%Complete** | **Notes** |
| Implement scoring function | 100% | We have finalized our scoring function using mutual information and it has been coded for consumption for other modules. Code is uploaded [here](https://github.com/swatinanda/CS410-Fall2021-TeamAlpha/tree/main/SourceCode/alarm_correlation_mutual_information).  High level notes on how do we calculate the mutual information is presented [here](https://github.com/swatinanda/CS410-Fall2021-TeamAlpha/blob/main/SourceCode/alarm_correlation_mutual_information/how_to_calculate_mi.png) |
| Data cleanup / preparation  (Dataset) | 100% | We have completed this task. It is being built with real alert monitoring scenarios based on App, Infra and Network monitoring. Some examples are   * High Average response time, * High CPU usage on the Host, * Router not reachable, * Kafka Lag breached a configured threshold etc.   Our utility will replay the dataset message located [here](https://github.com/swatinanda/CS410-Fall2021-TeamAlpha/tree/main/SourceCode/alarm_correlation_mutual_information/data) to generate real time alert template data. These alarms along with their mutual information will be populated in a Graph obj and will be sent to downstream Graph database (Neo4j). This additional alarm information will help us to enrich our “Knowledge Graph” |
| Create a bag of alerts | 100% | We have a corpus of Alert messages, Hostname and Source. This will be used to generate random alerts. Each alarm will have this 3 metadata information selected randomly from the corpus. We can create any number of dummy alarm templates through this. Associated code is [here](https://github.com/swatinanda/CS410-Fall2021-TeamAlpha/blob/main/SourceCode/alarm_correlation_mutual_information/src/main/java/com/illinois/cs410/project/AlarmFactory.java) |

1. Which tasks are pending?

|  |  |  |
| --- | --- | --- |
| **Tasks** | **%Complete** | **Notes** |
| Build Knowledge graph with limited datasets | 10% | We have completed the research related to various available frameworks for building Knowledge graph. We plan to use Neo4j for storing the graph. Chabbot will be converting the text query into Neo4j based cypher query to extract any relevant information.  Plan to wrap it up by Nov 18th |
| Develop REST API for frontend interaction  Implement user interface | <Swati> | Framework to train data for chatbot in progress.  Will club the two tasks to present data into the chatbot.  Plan to wrap it up by Nov 18th |
| Implement Feedback loop |  | Need to evaluate if we’ll be able to accomplish it optimally. |
| Integration, Testing & Evaluation |  | Data for evaluation is being created as part of Dataset preparation. We will add some rules to baseline this data, against which we can evaluate/verify our algorithm output.  Will start the Integration and Testing post that (tentatively by Nov 24th or 25th) |

1. Are you facing any challenges?

We had to do our dataset from scratch, so that is taking some time.