Informatics Institute of Technology

In Collaboration With

University of Westminster, UK



*University of Westminster, Coat of Arms*

GATOT

Software Requirements Specification (SRS)

Mr. Nazhim Kalam

w1761265 / 2019281

Supervised by

Mr. Torin Wirasingha

September 2022

This Project Proposal is submitted in partial fulfilment of the requirements for

the BSc (Hons) Computer Science degree at

the University of Westminster.

Table of Contents

[List of Figures ii](#_Toc120468722)

[List of Tables ii](#_Toc120468723)

[1. Chapter Overview 1](#_Toc120468724)

[2. Rich Picture 1](#_Toc120468725)

[3. Stakeholder Analysis 2](#_Toc120468726)

[3.1 Stakeholder Onion Model 2](#_Toc120468727)

[3.2 Stakeholder Viewpoints & Requirement Elicitation Methodology. 3](#_Toc120468728)

[3.3 Research Question Mapping 5](#_Toc120468729)

[3.4 Roles & Requirement Elicitation Methodology 6](#_Toc120468730)

[4. Requirement Elicitation Methodologies 7](#_Toc120468731)

[5. Analysis of Data & Presentation of the outcome through Elicitation Methodologies 8](#_Toc120468732)

[5.1 Literature Review 8](#_Toc120468733)

[5.2 Survey 9](#_Toc120468734)

[5.3 Interviews 12](#_Toc120468735)

[5.4 Prototyping 14](#_Toc120468736)

[5.5 Brainstorming 15](#_Toc120468737)

[5.6 Self Evaluation 15](#_Toc120468738)

[6. Summary of findings 17](#_Toc120468739)

[7. Context Diagram 19](#_Toc120468740)

[8. Usecase Diagram 19](#_Toc120468741)

[9. Usecase Descriptions 20](#_Toc120468742)

[10. Requirements 22](#_Toc120468743)

[10.1 Functional Requirements 22](#_Toc120468744)

[10.2 Non-functional Requirements 24](#_Toc120468745)

[11. Chapter Summary 25](#_Toc120468746)

# List of Figures

[Figure 12.1 - Prototype Feature Diagram (Self-composed) 13](#_Toc117550682)

[Figure 13.1 - Gantt Chart 16](#_Toc117550683)

[Figure 13.2 - Model development flow (Self-composed) 21](#_Toc117550684)

# List of Tables

[Table 5.1 - Related work in abstractive text summarization 3](#_Toc117584436)

[Table 11.1 - Research Objectives 9](#_Toc117584437)

[Table 13.1 - Research Methodology 13](#_Toc117584438)

[Table 13.2 - Deliverables and Dates 17](#_Toc117584439)

[Table 13.3 - Risk Mitigation Plan 19](#_Toc117584440)

**Acronyms**

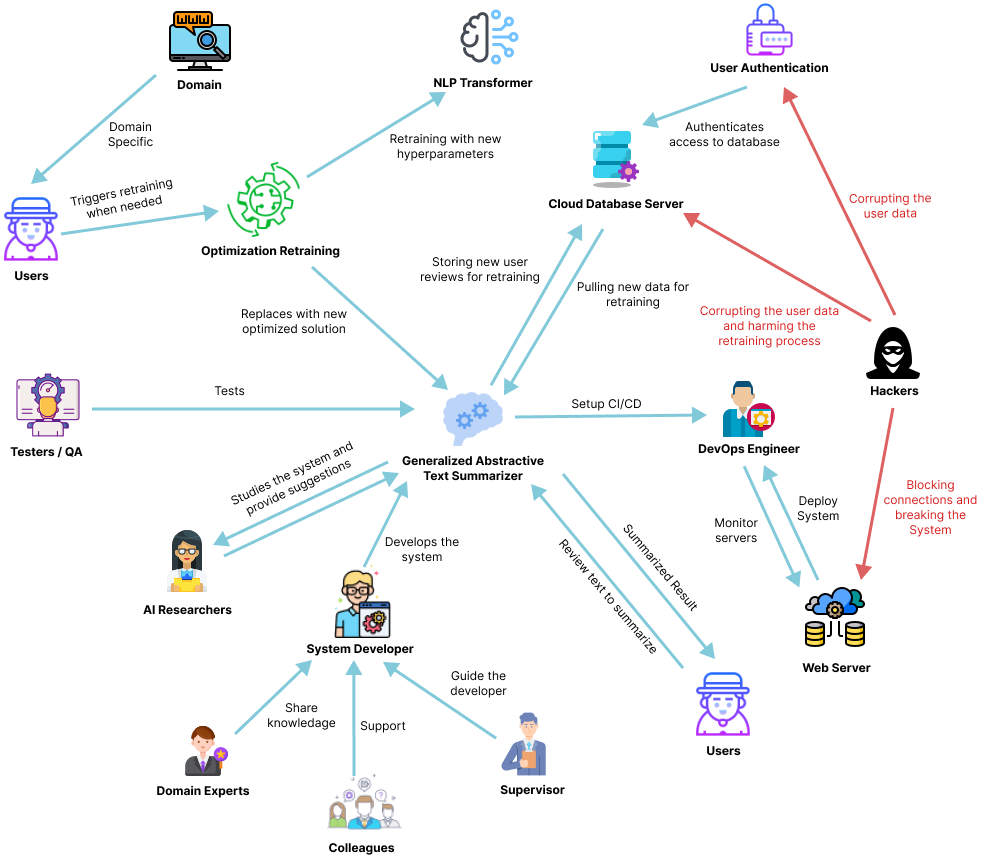
|  |  |
| --- | --- |
| AI | Artificial Intelligence. |
| DL | Deep Learning |
| GUI | Graphical User Interface |
| ML | Machine Learning |
| NLP | Natural Language Processing |
| ROUGE | Recall-Oriented Understudy for Gisting Evaluation. |
| BLEU | BiLingual Evaluation Understudy. |
| T5 | Text to Transfer Transformer. |
| BART | Bidirectional Auto-Regressive Transformers. |
| BERT | Bidirectional Encoder Representations from Transformers. |
| PEGASUS | Pre-training with Extracted Gap-sentences for Abstractive Summarization Sequence-to-sequence |
| ILP | Inductive logic programming. |
| LSTM | Long Short-Term Memory. |
| RNN | Recurrent Neural Network. |
| CNN  SEQ2SEQ | Convolutional Neural Network.  Sequence to Sequence |
| RoBERTa | Robustly Optimized BERT Pre-training Approach |
| GPT-3  REST  GPU | Third Generation Generative Pre-Trained Transformer  Representational State Transfer  Graphical Processing Unit |

# Chapter Overview

In this chapter, the author describes how to identify the essential needs and how to gather them. To carefully record the engagement of possible stakeholders, their interaction points, and their separate responsibilities, a rich picture diagram and stakeholder onion model are used. The chapter also discusses the methods used for requirement gathering and the results that were used to create functional and non-functional requirements, use case diagrams, and prototypes.

# Rich Picture

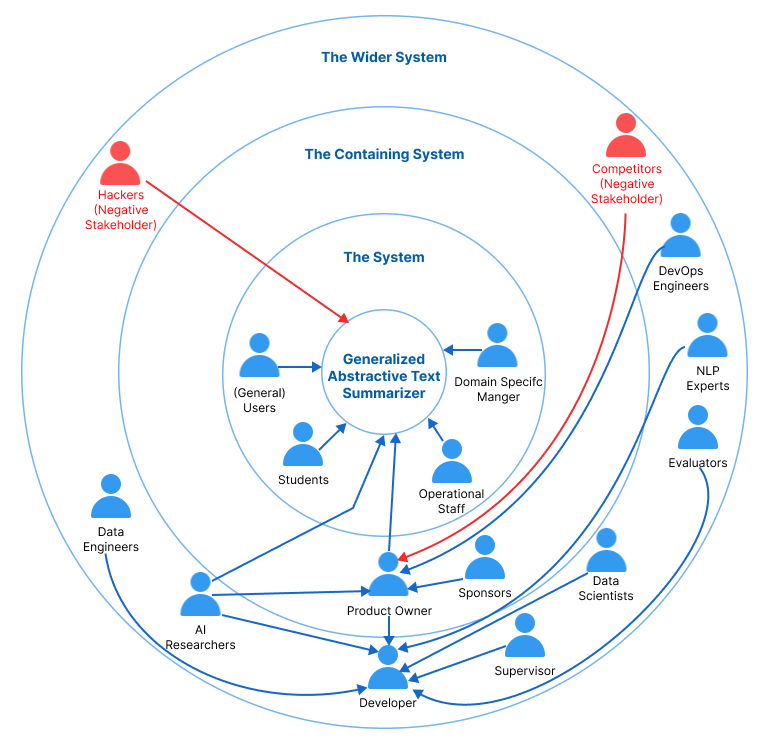
Figure 2.1 – Rich Picture Diagram (*self-composed*)

****

# Stakeholder Analysis

## **3.1 Stakeholder Onion Model**

Figure 3.1 – Stakeholder Onion Model (*self-composed*)



## 3.2 **Stakeholder Viewpoints & Requirement Elicitation Methodology.**

Table 3.1 – Stakeholder Viewpoints & Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **Stakeholder** | **Role** | **Benefits/ Role Description** | **Requirement Elicitation Methodology** |
| Developer | Financial Beneficiary | Works on developing the system | Interview |
| Sponsors | Funds to market the system and allows developers to advance the system with time. | NA |
| Data Scientists | Quality Control Regulator | Provides performance enhancements for the models and algorithms used in data science. | Interview |
| Data Engineers | Gives guidance on potential data that may be used to generate the best suggestions possible. |
| AI Researchers | Conduct research in the specified area to enhance and implement reliable text summarizing models. |
| NLP Experts | Offers specialized guidance and insights on the field  knowledge, to enhance the functionality of the system. |
| Students | Operational Beneficiary | Understanding how the generalization works along with hyperparameter retraining with respect to the domain. | Survey |
| Domain Specific Manager | Inputs the text reviews for abstractive summarization and handles the need for model retraining when need to improve performance with the previously used inputs as new data for the model. |
| General Users | General users (Not Domain Specific) will be using the general abstractive summarization model without any specific model assigned to them and hyperparameter retraining unless they want to. |
| Operational Staff | Ensures that the system is up and functioning while responding to user requests and problems. |
| DevOps Engineers | Product Deployment & Maintenance | Makes ensuring the system is up and running in the cloud and is serving users without being throttled | NA |
| Hackers | Negative Stakeholder | May manipulate the review data stored in the database which will affect the retraining process. | Self-Evaluation |
| Competitors | May build competing systems that may outperform the existing system. |
| Evaluators | Quality Inspector | Checks to see if the system is ready for production use and puts it through its paces. | NA |

## **3.3 Research Question Mapping**

Table 3.2 – Research Question Mapping List

|  |  |  |
| --- | --- | --- |
| **Question** | **Reason for asking the Question** | **RQ Mapping** |
| If you have worked with transformer in the field of NLP, which transformer designs have you found to be the best performers while working on NLP-related tasks? | In order to get the list of best performing transformer architectures. | RQ1 |
| What are the different ways of hyperparameter tuning that you know or have come across? | To get the knowledge of different ways of performing hyperparameter tuning and figure out the best | RQ2 |
| When it comes to evaluations in text/NLP related machine learning models, it's quite different from traditional evaluation methods such as creating a confusion matrix etc.… What would be the best evaluation methods for abstractive text summarization from your experience? | To figure out the best set of evaluation approaches for the field of abstractive text summarization using optimized transformers, to make sure if the optimization making any performance improvement from the existing results | RQ3 |
| Do you think creating a generalized performance solution for every domain is a good approach (let it be movies, hotels, ecommerce, tourist, restaurants etc.…) or just focusing onto a specific domain | Making sure that generalized solution is most wanted or most needed to see the importance of it | RQ4 |
| What approach do you think is the best/optimal of creating a generalized model for every domain using transformers? | To get insights of the ways of working with generalization | RQ4 |
| What are the business domains you think would benefit from the generic solution apart from the domain of movies, hotels, restaurants? | To get insights on the what are different business domains that would benefit from the solutions | RQ4 |
| What are your concerns about the data in terms of quality and necessary preprocessing techniques to be concerned? | To get insights about the concerns regrading the data preprocessing approaches taken and what else needs to be considered. | RQ2 |
| What do you think about hybrid models and integrating hybrid approach to this would be better to further increase performance? | To get an idea if its worth also attempting to create hybrid models to increase the performance further more. | RQ2 |
| What other ways do you think are possible to increase the performance of the system apart from the mentioned ways? | To get insights on other ways of increasing the performance of the system | RQ2 |

## **3.4 Roles & Requirement Elicitation Methodology**

Table 3.3 – Roles & Requirement Elicitation Methodology

|  |  |  |
| --- | --- | --- |
| **Role** | **Stakeholder** | **Requirement Elicitation Methodology** |
| Financial Beneficiary | Developer | Interview |
| Quality Control Regulator | Data Scientists, Data Engineers, AI Researchers, NLP Experts | Interview |
| Operational Beneficiary | Students, Domain Specific Manager, General Users | Survey |
| Negative Stakeholder | Hackers, Competitors | Self-Evaluation |

# Requirement Elicitation Methodologies

There were several requirement elicitation approaches used to collect needs for the creation of the research project. The approaches selected for this were literature review, survey, and prototype. The following is a discussion of the rationales behind selecting the mentioned requirement elicitation approaches.

Table 4.1 – Requirement Elicitation Methodology Description

|  |  |
| --- | --- |
| **Method** | **Description** |
| Literature Review | To determine research gaps in the chosen domain of interest and the intended topic of study at the project's outset, the author conducted a thorough literature analysis. Current systems were researched together with comparable technologies that might be applied to the existing systems that were referenced in literature in order to discover research gaps available in technologies that can be used. |
| Survey | A questionnaire was utilized as a survey instrument to obtain requirements and opinions from possible users of the suggested system. The author will benefit from this sort of poll in understanding people's thought processes and expectations for the prototype. It will also enable the author to explain whether or not the targeted users will benefit from the suggested solution. |
| Interviews | Interviews were performed to gain expert insight into domain-specific requirements and to determine the best method to address the issue at hand while adding to the body of knowledge through research. Interviews were determined to be the greatest source of information because the field is new and the technical expertise needed is very precise. Additionally, this technique allowed for the qualitative evaluation of the suggested system, allowing for the identification of any shortcomings or difficulties that could need to be resolved during prototyping. |
| Prototyping | The project was chosen to follow the Agile Software Development Life-cycle, thus prototyping would allow the author to test and evaluate the prototype while iteratively trying out several alternative implementations to find any potential areas for improvement. |
| Brainstorming | Whether you're attempting to come up with a broad subject before you start your research, you're trying to focus more specifically, or you're determining what evidence to use for a particular paragraph, brainstorming is a useful technique to produce ideas at every step of the process. In order to assess the system for personally, the author has a number of brainstorming sessions with his colleagues at various project stages. |
| Self-Evaluation | Self-evaluation is done in order to examine the currently available applications, do competitor analyses on the current systems, and get insight into how negative stakeholders, such as hackers, can breach the system and find a way around to protect the data and the system. |

# Analysis of Data & Presentation of the outcome through Elicitation Methodologies

The data analysis from the requirement elicitation methods that were selected are shown below.

## **5.1 Literature Review**

Table 5.1 – Requirement Analysis from literature review findings

|  |  |
| --- | --- |
| **Finding** | **Citation** |
| In the completion of the literature review on the existing work done, it was identified that abstractive text summarization systems for customer reviews helps users to make better and quicker decisions on their actions let it be on buying products or watching a movie, user review summarization proves to save time for customers. | (Boorugu, Ramesh and Madhavi, 2019) |
| When exploring technologies that can be applied to achieve the required outcome, it was clear that traditional machine learning and deep learning approaches were only used for abstractive text summarization in the domain of movie reviews. Leaving the usage of advanced deep learning approaches such as Transformers untouched for this domain. | (Khan et al., 2020) |
| It was identified that transformer optimization has not been looked into when working with transformers in abstractive text summarization domain in general and not specific to the movie domain. | (Gupta et al., 2021) |
| Dataset related to working with model generalized has been used previously and is recommended to be used if researchers are willing to work with the idea of generalization for the domain of abstractive text summarization. | (Kouris, Alexandridis and Stafylopatis, 2019) |

## **5.2 Survey**

Table 5.2 – Requirement Analysis from questionnaire

|  |  |
| --- | --- |
| **Question** | Have you ever realized that reading lengthy reviews takes a significant amount of time? |
| **Aim of question** | To determine whether the audience as a whole considers reading lengthy reviews to be a time-consuming activity. |
| **Findings & Conclusion**  It can be concluded that a large part of the audience (more than 90% of the audience) finds that’s reading lengthy reviews is a time-consuming hassle which also proves that they would appreciate if there would be a quicker approach for this problem, like a summarization. This also concludes to see a positive correlation from the results which was expected from the author of the project. | |
| **Question** | Do you believe that developing a generic system for all domains would be a wise course of action? |
| **Aim of question** | Ensuring that developing a generic system would be beneficial in all domains |
| **Findings & Conclusion**  It can be concluded that most of the participants (more than 90% of the audience) agrees that developing a generalized system which can adapt to the domain as they use, is beneficial and worth the effort to process with the project research. This also concludes to see a positive correlation from the results which was expected from the author of the project | |
| **Question** | Who do you think will most benefit from this system? |
| **Aim of question** | Getting to know about the thoughts of the participants about to whom the system would mostly benefit from? |
| **Findings & Conclusion**  It can be concluded that a majority of the participants (more than 60%) finds that this system will benefit the movie, restaurant, tourist, hotel, ecommerce domains (these domains were considered since they are mostly interacted with the users on a daily bases and uses customer reviews for their domain as a part of their business) as well as the general users. | |
| **Question** | How much do you think that this system would benefit you? |
| **Aim of question** | Getting to know how much the system would benefit the general participants which are NOT domain specific |
| **Findings & Conclusion**  From the statistics graph, it can be concluded that roughly 75% of the audience finds that the system would benefit them for their general work or needs given that its not domain specific to them, which is a positively correlated result from the achieved statistics. | |
| **Question** | How much do you think that this system would benefit businesses? |
| **Aim of question** | Getting to know from the participants as to how much the system would benefit businesses/domains in solving this problem. |
| **Findings & Conclusion**  From the statistics graph, it can be concluded that roughly 84% of the audience finds that the system would benefit the businesses, which is a positively correlated results from the achieved statistics and that’s what the author expected to achieve. | |
| **Question** | Before making a reservation or booking a movie or a hotel, do you read the customer reviews? |
| **Aim of question** | Getting an idea from the audience if in general they give importance to customer/user reviews to any domain before consuming their product or services. |
| **Findings & Conclusion**  It can be concluded that a majority of the participants (more than 95% of the audience) agrees that they value and read customer reviews before they consume one’s product or service. Therefore, making customer reviews a major contributing factor for business growth. | |
| **Question** | How much you think customer reviews are important with respect to any domain? |
| **Aim of question** | Getting an idea from the audience to see how much they value customer reviews. |
| **Findings & Conclusion**  From the statistics graph, it can be concluded that roughly 90% of the audience finds that customer/user reviews are very important irrelevant to the domain, which is a positively correlated results from the achieved statistics and that’s what the author expected to achieve. | |
| **Question** | Which additional features would you want to see in this system. |
| **Aim of question** | To identify the systems non-functional requirements which could potentially improve the system. |
| **Findings & Conclusion**  The majority of participant responses were concerned with classifying the review text's sentiment after it had been summarized and with managing a list of review uploads so as to add filtering for the summarized review text based on the sentiment, whether it was positive or negative, along with the sentiment score. | |

## **5.3 Interviews**

Interviews with experts and researchers in the relevant domains were performed to obtain insights on the technical domain competence. To determine the project requirements, experts and researchers in ML and abstractive text summarizing systems were chosen. 2 PhD candidate in ML and Computational Linguistics, 1 NLP Researcher, 2 Software Architects, 1 Software Engineer and 1 Lecturer with MSc completion were interviewed. The interview outcomes were processed to a **thematic analysis** based on the following themes.

Table 5.3 – Requirement Analysis from interviews

|  |  |
| --- | --- |
| **Theme** | **Analysis** |
| Data Collection & Data Preprocessing | Since this is a project connected to data science, the availability of data and the data preparation methods to be used are the main concerns. PhD candidates suggested to make use of verified and well researched datasets for the area of generalization since every domain will be using the same model initially to start off with, therefore the quality of data should be considered, it was recommended to use datasets that have already been studied and utilized by other researchers since they have done so and verified their findings. NLP researches were concerned on the language of text the project scope is into when performing text preprocessing, since text data can also contain other language characters unless the project is scoped down to only English language supportive. |
| Best performing transformer architectures | Most of the interviewees pointed out similar transformers architectures which they have used and found impressive results, which are mostly BERT, GPT-2, RoBERTa, T5 etc... where they have explored not only with text summarization but also when other NLP areas such as sentiment analysis, proving again that transformers are well known for solving NLP problems. They also stated to check up with the daily stats (most downloads and likes) about the transformer architectures from Hugging Face, this is because new better versions of the transformers are always been produced/updated. |
| Handling adaptive generalization | The Software Engineers and Architects suggested to make use of document-oriented NoSQL database management system to handling data storage for the domain specific managers, this is because its easily scalable and provider superior performance especially for the idea of adaptative generalization for this project. Such services are like MongoDB, Firebase NoSQL DB etc. |
| Research gap & scope | The technology exports and research experts find that the solution of solving this problem using optimized transformers is great but they find that creating a generalized adaptive solution would be challenging with the time frame of the project but also advised to solve for the domain of movies first and then get into the others if time permits. |
| Hyperparameter tuning | The NLP researchers and Lectures suggested several ways of using tools and libraries to help with hyperparameter tuning since doing this manually is very time consuming and unnecessary effort. |
| Looking to hybrid transformer combinations. | PhD candidates liked the idea of using hybrid transformer combination by using ensemble approaches to combine the top best two transformer architecture but it seems the scope of the project for the time frame is becoming bigger and riskier. |
| Prototype features & suggestions | The interviewees are interested to see how the generalization system for domain specific retraining is going to work together since they haven’t seen any such approach earlier from their experience. They also suggested if time permits to make use of a pretrained model to get the sentiment of the summary aswell to be displayed on the GUI. |
| Understanding which and how business would benefit. | Most of the interviewees suggested the Movie domain, Tourism, Ecommerce, Book, Researchers would find this useful in summarizing their customer reviews on their businesses. |
| Understanding the importance and evaluation ways. | The PhD candidates and NLP experts suggested the importance of evaluations when it comes to dealing with the adaptive generalization model since this can be used in any domain, therefore suggesting the author of the project to explore maximum of 3 domains when working with so its easier to compare the evaluation results else it will be confusing when demonstrating the work to anyone. |

## **5.4 Prototyping**

Several challenges and requirements emerged though the iterative prototyping. Since this project problem domain was related to the field of movies, finding the dataset with the expected metadata was challenging, however through multiple research paper reviewing the author was able to find the datasets but the amount of data present was massive (around 8million records), the author had to break it down to multiple dataset portions to work with. However, finding the dataset for model generalization wasn’t challenge but preprocessing the data was a major task since the dataset was massive and also contained noisy text data, which needed a lot of cleaning.

Since manual hyperparameter tuning was a very time-consuming task, a research for hyperparameter tuning libraries were experimented and “Optuna” worked for our needs when it comes to model automatic optimized retraining. New data entered by the domain user of the application has been designed to be stored and used for retraining their model. At least 5 top tier transformer architectures had to be explored with for choosing the best architecture.

## **5.5 Brainstorming**

The author engaged in brainstorming across various project phases. These were carried out both with the authors' colleagues and supervisors as well as through a self-analysis process.

Table 5.4 – Requirement Analysis from brainstorming

|  |
| --- |
| **Purpose** |
| Able to figure out several other research gaps/ limitations which can be fit into the current project domain in order to increase the magnitude of research effort. |
| **Findings** |
| Multiple ideas were brought up as the result of the brainstorming session. The concept of creating a performance adaptive generalization model was brought up by the authors supervisor, along with several other approaches to increase the performance of the system exponentially such like making use of the new data from the domain users for retraining and combine all data with the common domain for retraining since the data count increases with respect to the common domain user. |

## **5.6 Self Evaluation**

Comparing similar products from competitors and existing products gives the author an idea of making the project more unique and distinguish new approaches to solve the problem (**Competitor Analysis**). The author will also self-evaluate as to what data needs to be protected and how from the hackers. The table below shows few of the abstractive text summarization tools which are out there.

Table 5.5 – Requirement Analysis from Self Evaluation

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Competitor Analysis Table** | | | | | |
| **Tools**  **Feature** | Summarize Bot | Resoomer | Smmry | Text Compactor | Transformer Optimized Text Summarizer |
| Summarizing Text | **✓** | **✓** | **✓** | **✓** | **✓** |
| Domain Specific Generalization | 🗶 | 🗶 | 🗶 | 🗶 | **✓** |
| Ease of Use via GUI | 🗶 | **✓** | **✓** | **✓** | **✓** |
| Summary of sentiment (\*this is not in project scope but if time permits will do\*) | 🗶 | 🗶 | 🗶 | 🗶 | **✓** |

In the case of hackers stealing data from the database, **data encryption** can be applied therefore database will only contain the encrypted text data which will be then later decrypted from the decryption key when need, this will be most needed when performing the model retraining.

# Summary of findings

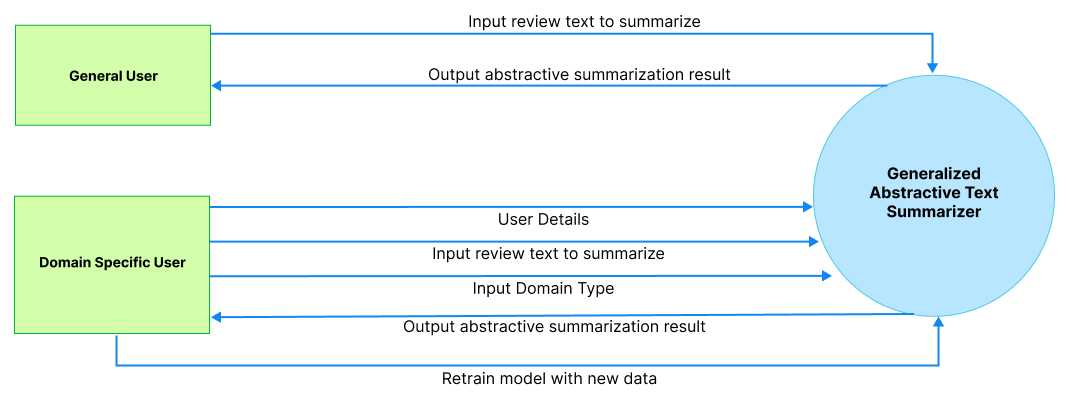
Table 5.6 – Summary of Findings

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Id** | **Finding** | **Literature Review** | **Survey** | **Self-Evaluation** | **Interview** | **Brainstorming** | **Prototyping** |
| 1 | The proposed system would benefit businesses (domain specific users) and general users (not domain specific) |  | ✓ |  |  | ✓ |  |
| 2 | For the movie domain the limit of abstractive text summarization can be further pushed using optimized transformers to increase performance this being the existing limitation | ✓ |  |  | ✓ | ✓ |  |
| 3 | Its clear that customer/user reviews are valued and reviewed mostly by a vast majority of the audience before they consume or use any product or service (applies to any domain) | ✓ | ✓ |  | ✓ | ✓ |  |
| 4 | Its clear that users spend lot of time review long reviews and they would like it being short to save time and make quicker decisions. | ✓ | ✓ |  |  | ✓ |  |
| 5 | Hyperparameter tuning is one way to increase the performance of the system and it can be done both manually or by automated tools like Raytune, Optuna etc.… | ✓ |  |  | ✓ |  | ✓ |
| 6 | Data preprocessing for the domain of Movies and Generalization is requires a lot of effort since the datasets are mostly raw data difficult to find specially in the case of movie reviews (with expected metadata) | ✓ |  |  |  |  | ✓ |
| 7 | Additional features such as sentimental and sentimental score of the review summary is mostly expected from the users. |  | ✓ |  |  |  |  |
| 8 | Creating a hybrid transformer model to further increase the performance is a suggested improved. |  |  |  | ✓ | ✓ |  |
| 9 | It’s clear on what are the suitable evaluation metrics to be used for abstractive text summarization. | ✓ |  |  | ✓ |  |  |
| 10 | It’s clear on what the top tier transformer architecture that could be explored. | ✓ |  |  | ✓ |  |  |
| 11 | Making use of larger new data for retraining for a specific domain, from companies/businesses who uses data which are of the same domain. (e.g.: - 50 different restaurants data can be combined for retraining give that the domain is “Restaurants”) |  |  |  | ✓ | ✓ |  |
| 12 | Making use of data encryption to protect the data from hackers breaking into the database to steal data. |  |  | ✓ |  | ✓ |  |

# Context Diagram

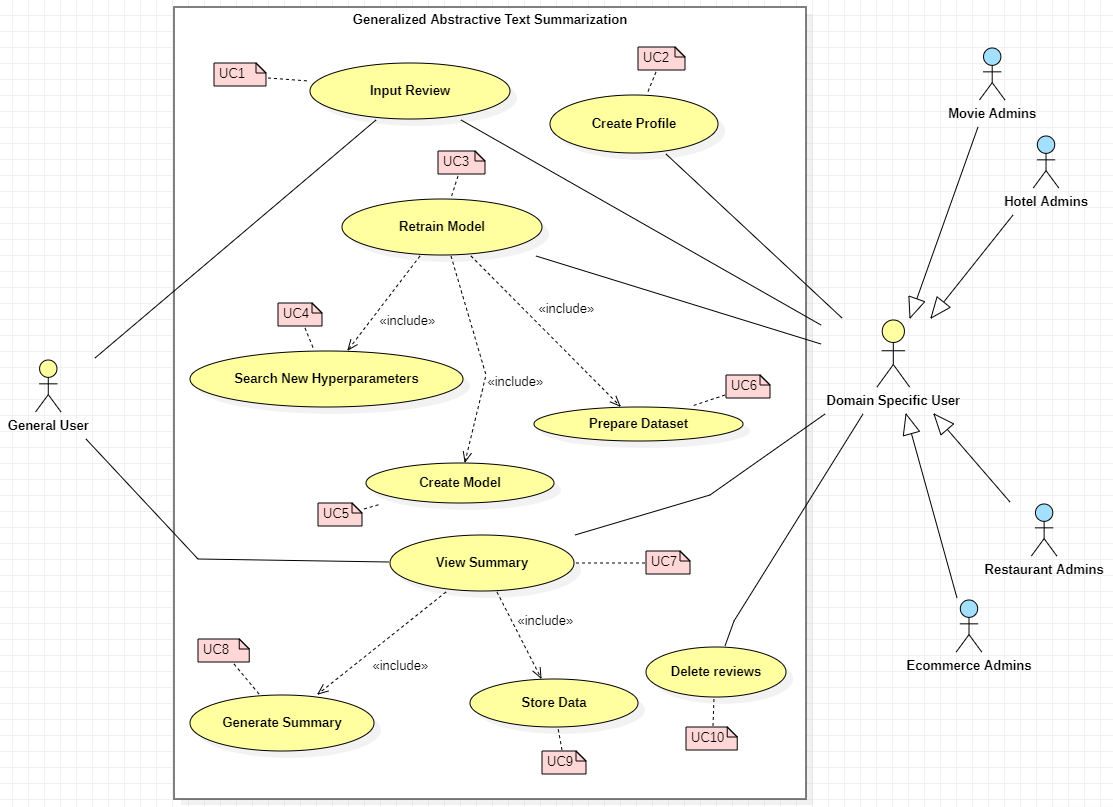
The boundaries and interactions of the system should be established before development. The graphic below shows how the system is situated.

Figure 7.1 – Context Diagram (self-composed)



# Usecase Diagram

Figure 8.1 – Use case Diagram (self-composed)



# Usecase Descriptions

Table 9.1 – Use case description UC:07

|  |  |  |
| --- | --- | --- |
| Use Case Name | View Summary | |
| Use Case Id | UC:07 | |
| Description | Displays a summarized version of the uploaded review text from the domain user’s end. | |
| Primary Actor | General User, Domain Specific User | |
| Pre-Conditions | The text review data must go through specific text preparation techniques before the summary can be produced. | |
| Extended use cases | None | |
| Included use cases | UC10, UC02 | |
| Trigger | A user selects to summarize a given customer/user review text. | |
| Main flow | **Actor** | **System** |
| 1. The user enters the review text on the text field from the GUI. 2. Clicks on “Generate Summary” from the GUI | 1. The system does the data preprocessing for the input review text. 2. Loads the generalized transformer model. 3. Generates the summary using the model. 4. **(If Domain Specific User)** stores the input review and summary into the database. 5. Returns the summary response back to the GUI |
| Alternative flows | None | |
| Expectational flows | Displays an error message if the network request fails (server is down, or internet issues from client). | |
| Post Conditions | Success end condition: The user is presented with the summarized review text. | |

Table 9.2 – Use case description UC:03

|  |  |  |
| --- | --- | --- |
| Use Case Name | Retrain Model | |
| Use Case Id | UC:03 | |
| Description | Performs model retraining with the new data from the database, to find the new best set of hyperparameters. | |
| Primary Actor | Domain Specific User | |
| Pre-Conditions | The actor should be a Domain Specific User and have an account created. | |
| Extended use cases | None | |
| Included use cases | UC05, UC06, UC07 | |
| Trigger | The Domain Specific User clicks on the “Perform model retraining” button | |
| Main flow | **Actor** | **System** |
| 1. Domain Specific logs into their account 2. Clicks on “Perform model retraining” from the GUI | 1. The system pulls all the data with respect to the user id from the database. 2. Combines data of the common domains (only if user consent is given to use their data) 3. Finds new set of hyperparameters for the model with respect to new data. 4. Trains the model using the new hyperparameters. 5. Saves the model with the user Id 6. Updates the status in the database if succeed/fails |
| Alternative flows | None | |
| Expectational flows | Displays an error message if the network request fails (server is down, or internet issues from client). | |
| Post Conditions | Success end condition: The user will be able to see the recent status of the model if the retraining is successful or failed | |

# Requirements

## **10.1 Functional Requirements**

Based on the significance of the system demands, the MoSCoW approach was utilized to identify their priority levels.

Table 10.1 – Priority Levels

|  |  |
| --- | --- |
| **Priority Level** | **Description** |
| Must have (M) | The demand at this level is the fundamental functional requirement for a prototype, and it must be carried out. |
| Should have (S) | Although not strictly required for the anticipated prototype to function, important criteria do provide a lot of value. |
| Could have (C) | Optional, non-essential desirable needs are crucial to the project's scope. |
| Will not have (W) | Requirements that the system might not meet right now and that are not given first consideration. |

Table 10.2 – Use case and Use case Mapping

|  |  |
| --- | --- |
| **Use case Id** | **Use case name** |
| UC01 | Input Review |
| UC02 | Create Profile |
| UC03 | Retrain Model |
| UC04 | Search New Hyperparameters |
| UC05 | Create Model |
| UC06 | Prepare Dataset |
| UC07 | View Summary |
| UC08 | Generate Summary |
| UC09 | Store Data |
| UC10 | Delete reviews |

Table 10.3 – Functional requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **FR ID** | **Requirement** | **Priority Level** | **Use Case** |
| FR1 | Both general and domain specific users must be able to enter a review text from the GUI considering as the starting point of the summary generation. | M | UC01 |
| FR2 | Only Domain Specific Users should be able to sign up and create an account after entering the necessary details required | S | UC02 |
| FR3 | The system could allow the ability to update the account details of the domain user after creating the account | C | UC02 |
| FR4 | The system must undergo model retraining with the new data stored in the database for the specific domain user, when its triggered from the GUI with the user’s consent. | M | UC03 |
| FR5 | The system could be able to perform model retraining automatically during off peak hours every day. | C | UC03 |
| FR6 | The system must be able to find the new set of best hyperparameters with the usage of the new data. | M | UC04 |
| FR7 | The system must be able to able to retrain the model with the new best hyperparameters and create the model | M | UC05 |
| FR8 | The system must be able to pull the new data from the database to recreate the new dataset for retraining. | M | UC06 |
| FR9 | The system should be able to combine all the data from a common group of domains when creating the dataset only given that the consent is approved to use their data | C | UC06 |
| FR10 | The system must be able to process the review text and display the summary output on the GUI | M | UC07 |
| FR11 | The system must be able to use the latest trained model to generate the summary for the review text | M | UC08 |
| FR12 | The system could also find the sentiment of the generated summary if its positive or negative and return the result. | C | UC08 |
| FR13 | The system could make use of a hybrid model for the text summarization. | C | UC08 |
| FR14 | The system must store the entered user review and generated summary to be stored in the database for retraining purposes. | M | UC09 |
| FR15 | The system should encrypt the data when saving into the database (both the review and summary) | S | UC09 |
| FR14 | The system could allow the domain users to delete the reviews from the database. | C | UC10 |

## **10.2 Non-functional Requirements**

Table 10.4 – Non-functional requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **NFR ID** | **Requirement** | **Specification** | **Priority Level** |
| NFR1 | The system needs to be simple enough for non-technical individuals to utilize without much effort. | Usability | Important |
| NFR2 | Meaningful error messages should be displayed if anything goes wrong | Usability | Desirable |
| NFR3 | Summary generation should be done within 3000ms | Performance | Important |
| NFR4 | Following coding standards and best practices | Maintainability | Important |
| NFR5 | Any domain users are able to use the application and model performance will adapt with respect to the domain | Generalization | Important |
| NFR6 | The system should protect against data corruption by attackers, and testing can ensure this. | Security | Desirable |
| NFR7 | The prototype can be used by several domains and multiple businesses under a single domain, then the system may have to support many concurrent user-requests. | Scalability | Desirable |

# Chapter Summary

In this chapter, a Rich Picture Diagram was created to show how the system interacts with society and the system stakeholders. The stakeholders were represented using Saunder's Onion model, which included the flow of influence from each stakeholder. To acquire all the necessary information and the opinions of potential system stakeholders, requirement gathering approaches were used. Last but not least, the insights gained from the requirement elicitation approaches were used to specify the system's use cases, functional requirements, and non-functional requirements.