Informatics Institute of Technology

In Collaboration With

University of Westminster, UK



*University of Westminster, Coat of Arms*

Abstractive Text Summarization for Movie Reviews Using Optimized Transformers

A Project Proposal by

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.

**1. INTRODUCTION**

In this research project, the author tries to increase the performance of abstractive text summarization for the domain of movie reviews by performing hyperparameter optimization on a set of top tiers pretrained transformer architectures, in-order to achieve an optimized architecture.

This document will discuss the research problem, research gap, research challenge, and the research approach that the author aims to address over the next months. Additionally, a review of prior research interests and the essential evidence of the issue is done. Finally, in the work plan, the expected schedule of the project’s deliverables is presented.

**2. PROBLEM DOMAIN**

**2.1. Movie Reviews**

A growing number of websites, like Amazon and the Internet Movie Database (IMBD), a website for movie reviews, allow users to publish reviews for the things they are interested in, in line with the growth of Web 2.0, which places an emphasis on user interaction. (Khan et al., 2020)

Online movie reviews are evolving into an important information source for Internet users as the amount of information on the web increases (Khan et al., 2020). Online users, however, publish tens of thousands of movie evaluations every day, making it challenging for them to manually summarize the reviews. The mining and summarizing of movie reviews is one of the difficult issues in natural language processing (Khan et al., 2020).

Text summary aids users or corporate executives in decision-making by gathering and evaluating a significant collection of online reviews (Alsaqer and Sasi, 2017).

These days, most of us check the movie ratings before choosing or watching a movie on any platform, such as Netflix or Amazon Prime, but we also come across mixed movie reviews that can be either positive or negative. This is where the issue arises because we will feel the need to read through all the reviews to determine how and what the movie is specifically about and whether it meets our needs. A summary of the countless reviews a film has gotten can make it easier for viewers (or customers) to quickly peruse the summary and decide about whether or not to watch the film. The summaries of movie reviews, on the other hand, can help streaming services like Netflix quickly discover the viewing habits or preferences of their users (Khan et al., 2020)

**2.2. Text Summarization**

Given that there is a lot of textual information available nowadays, including news articles and reviews, text summarizing allows us to rapidly discover the important points of the entire piece by reducing the amount of text (Mahajan et al., 2021).

Extractive summarization and abstractive summarization are typically the two methods of text summarization. In the case of extractive summarization, the context or article's most valuable sentences are taken out without changing them in any way. On the other hand, abstractive summarization tries to develop the sentences on its own and produce the summary; this is much better than extractive summarization, since it is more meaningful to build our own phrases inside the context than to use chosen sentences from the context without any alteration (Etemad, Abidi and Chhabra, 2021).

**2.3. Transformers**

Transformers in NLP is a novel architecture that aims to solve sequence-to-sequence tasks while handling long range dependencies with ease. It has surpassed competing neural models like CNN (Convolutional Neural Nets) and RNN (Recurrent Neural Nets) in terms of performance to appear as the dominant architecture for natural language processing (Wolf et al., 2020).

Transformers uses self-attention mechanism to target on selected areas of the input sentence followed by the encoder and decoder architecture (Etemad, Abidi and Chhabra, 2021).

**3. PROBLEM DEFINITION**

In the domain of movie review summarization, currently there are no in-depth researches done using deep learning approaches to solve this problem, mostly standard machine learning algorithms such as Naïve Bayes have been used, and that’s where the room for the usage of deep learning approaches comes into picture in order to enhance the quality/accuracy of the text summarization process for this domain area.

Although deep learning models take longer to train, they offer more accuracy because they can simultaneously automate feature extraction and classification, whereas machine learning algorithms require feature selection first. Therefore, applying deep learning techniques will help to improve the quality of text summarization and help the user in making better decisions.

* 1. **. Problem Statement**

The use of deep learning approaches to generate abstractive summaries from movie reviews has not been investigated before, which can help in increasing the quality of text summarization for this domain.

**4. RESEARCH MOTIVATION**

The problem identified in this proposal will not just to helpful for this specific domain in text summarization with movie reviews but instead it will be helpful and can be used in several other domains which requires the need to improve the quality of abstractive text summarization using the advanced approaches of deep learning.

As mentioned in the work of (Etemad, Abidi and Chhabra, 2021), syntactic and semantic issues with text summarization were the main issues that researchers were concentrating on solving. and with respect to their intense research by exploring multiple deep learning models, they have come into conclusion that Transformer based models (T5 model) outperformed in all NLP tasks, this encourages the author to go deeper into the field of transformers optimization in order to enhance the quality of text summarization and address the constraints associated with the summarizing of movie reviews.

**5. EXISTING WORK**

Table 1.5.1: Related work in Abstractive text summarization

|  |  |  |  |
| --- | --- | --- | --- |
| **Citation** | **Brief Description** | **Limitations** | **Contribution** |
| (Khan et al., 2020) | An automatic approach is desirable to summarize the lengthy movie reviews and allow users to quickly recognize the positive and negative aspects of a movie. | Making use of deep learning models to generate abstractive summaries from movie reviews. | Making using of **BoW** (Bag of words) for feature extraction and converting reviews into vector space, followed by the Naïve Bayes machine learning algorithm used for review classification (either positive or negative), then using an undirected weighted graph based ranking algorithm to rank score for reach review sentence in graph. Finally, the top ranked sentences are chosen based on highest rank scores to produce **extractive summary.** |
| (Boorugu, Ramesh and Madhavi, 2019) | Using all the customer reviews on products when making purchasing decisions to give a proper summarization of the reviews to the customer, so that he doesn’t need to go through all the reviews to figure out if the product is what he is looking for and save time. | Focused on improving the accuracy by using the latest models in the field of text summarization. By using transformers architecture, we could improve this. | Using seq2seq model for summarization along with attention mechanism for increased accuracy, also using word embedding model Concept net Number batch which is better than Glove. Finally, using a 1D convolutional layer followed by max pooling layer, LSTM layer and then at the end a fully connected layer. |
| (Mukherjee et al., 2020) | A solution for generating personalized aspect-based opinion summaries from large collections of online tourist reviews, also able to customize the attributes of the summary based on the user’s interest. | Motive for the need to create tourist review dataset for our experiments.  The need for also experimenting with the data of lesser known places (Tourist locations) | Using an Integer Linear Programming (ILP [Unsupervised method]) based extractive technique to select an informative subset of opinions around the identified aspects. Evaluate and compare the summaries using ROUGE based metrics and obtain competitive results. |
| (Gupta et al., 2021) | This research will be presenting a comprehensive comparison of a few transformer architecture based pre-trained models for text summarization. | Future work should focus on building more robust models which can further extend the algorithm to create summaries of variable length and apply for multi-document summarization. | Using the pretrained models such as Pipeline BART, BART modified, T5 and PEGASUS to work with the text summarization. Evaluation metrics we done using the ROUGE Scores. |
| (Mahajan et al., 2021) | The aim of the project is to generate a text summary along with proper grammar and no repeated words using the Encoder-Decoder model with the attention layer. | Real time training required if this is used in production, in order to train with the latest articles with time. | Developed an encoder-decoder model using Gated Recurrent Units and trained the model to generate abstractive summary from an article. |
| (Etemad, Abidi and Chhabra, 2021) | Experimenting the text summarization domain with deep learning approaches and finding which performs the best, from RNN, CNN, Transformers etc.… | NA | Experimenting with RNN based models’ architectures, working with pre-trained transformer-based model architectures. Finally, using evaluation metrics such as BLEU and ROUGE to evaluate the models. |
| (Alsaqer and Sasi, 2017) | This is an approach where the research focuses on improving the sentimental text summarization analysis for movie review using RapidMiner. | NA | The first model of summarization is built using  the Aylien Text Analysis extension. The proposed second  model is built using the Text Processing extension. For both  these methods, the sentiment analysis is done using the same  Aylien Text Analysis extension for evaluating the  summarization results. |

**6. RESEARCH GAP**

Based on previous work done related to abstractive text summarization on movie reviews, the literature doesn’t identify or seek for the need of using advanced deep learning approaches to improve the performance of text summarization for this domain over traditional machine learning approach.

This project focuses on Empirical gap in the Movie Domain, as well as Theoretical and Performance gaps in the area of transformer optimization. Transformers plays a major role in the field of deep learning especially at problems related to Natural Language Processing, by performing hyperparameter optimization on several transformer architectures we can contribute to the enhanced quality of abstractive text summarization.

1. **RESEARCH CONTRIBUTION**

The need for improving the performance of an existing solution is very common in the field of data science, as we can explore new algorithms or fine-tuning existing algorithms to meet better expectations. The contributions of this project can be classified as theoretical contributions and domain contributions.

**7.1. Technological Contribution**

There are several deep learning approaches which can be used to handle abstractive text summarization, however with the previous research applied on this domain of NLP text summarization, it is found that transformers outperform most of the other deep learning approaches as of today specific to this field but there was no more research on optimizing them for a much better performance.

This research will be focused on getting the best optimized transformer architecture from few of the top tier existing pre-trained model by fine-tuning and performing hyperparameter optimization, therefore we are able to maximize the performance of the recommended architecture. Additionally, it is believed that this study approach could be utilized in any field that utilizes abstractive text summarization transformers.

**7.2. Domain Contribution**

Neural Networks makes up the backbone of deep learning algorithms which enables them to process complex unstructured data over normal means of machine learning algorithms. It is found that, the need for using advanced deep learning approaches has not been explored in the domain of movie review summarization.

Given that transformers perform well in this field, the proposed solution for this domain will be finding the recommended architecture along with hyper-parameter optimization by fine-tuning the model, to reach its best performance. An additional contribution will be that, the proposed solution will be generalized to any other domain linked with the field of NLP text summarization.

1. **RESEARCH CHALLENGE**

The main objective of this research is to achieve the optimized transformer architectures for the field of NLP abstractive text summarization. Transformers were introduced in 2017 by a team at Google Brain and are the most used choice for NLP problems replacing RNN models, given that this architecture was introduced not much longer back brings to a point where there is a lack of research done in the area of transformer optimization for the purpose of abstractive text summarization. (Wolf et al., 2020). Therefore, finding the most recommended transformer architecture along with the optimal parameters becomes a challenge with very less resources to look up to.

Additionally, identifying suitable datasets for this domain (Movie Reviews Summarization) is challenging and necessitates a substantial amount of effort in data preprocessing where it is important since we are dealing with NLP and performance optimization related domain.

1. **RESEARCH QUESTIONS**

**RQ1:** What are the top tier transformer architectures widely used and know for NLP problems related to text summarization?

**RQ2:** How can a pretrained transformer architecture be fine-tuned to get the optimal hyper parameters?

**RQ3:** What kind of evaluations should we perform after fine-tuning to filter out the best transformer architecture?

**RQ4:** What kind of metadata is expected to be required from the dataset for this problem?

**10. RESEARCH AIM**

***The aim of this research is to propose the most optimized transformer architecture from a range of popularly used architectures by fine-tuning the model via hyperparameter optimization, therefore obtaining the recommended architecture's optimum performance for abstractive text summarization.***

To further explain the objective, a fully working system that can be utilized to execute abstractive text summarizing based on the movie review provided as input will be created by this research project. The quality of the resulting text summary or performance optimization will be the main points of emphasis. To get the best result, the usage of data preparation, data analysis, conducting hyperparameter tuning, and evaluating the models will be investigated.

To confirm or refute the selected hypothesis, the necessary information will be obtained and investigated, components will be built, and performance will be evaluated. Both a hosted server and a local browser will be able to execute the system for private or public usage. The data science models and their source code will be made accessible for future study and usage in a public repository that is simple to set up and utilize. The information gleaned from the literature review will be published in a review paper. On the basis of the study project's findings, a research paper will be published.

**11. RESEARCH OBJECTIVES**

The completion of the resulting research objectives is expected to fulfill the aims and provide answers to the research questions listed above. These goals are benchmarks that must be achieved for the research to be considered successful.

Table 1.11.1: Research Objectives

|  |  |  |
| --- | --- | --- |
| **Objective** | **Description** | **LO** |
| Literature Survey | Complete a thorough critical review of earlier related work.  **RO1:** Make a preliminary investigation on existing abstractive text summarization using deep learning approaches.  **RO2:** Make a preliminary investigation on why transformers architecture was the chosen deep learning choice for this research.  **RO3:** Analyze the top tier transformer architectures widely used.  **RO4:** Analyzing how the models can be fine-tuned via hyperparameter optimization.  **RO5:** Analyzing the different approaches used for model evaluation. | LO2,  LO4,  LO5 |
| Requirement Analysis | Defining the project's needs utilizing relevant approaches and tools in order to solve the projected research gaps and obstacles based on prior related research.  **RO1:** Gathering information related to the expected metadata required for the dataset to contain for the model training.  **RO2:** Gathering the requirements of transformer architectures for fine-tuning and understand the end to end user expectations.  **RO3:** Getting insights from domain experts to build a suitable system. | LO1, LO2, LO5, LO7 |
| Design | Considering the following when developing the suggested system:  **RO1:** Design a component to preprocess the dataset for the respective model inputs.  **RO2:** Design a component to store the top tier transformer models with their respective metadata, to use throughout.  **RO3:** Design a hyperparameter tuning component that can improve accuracy of the transformer model.  **RO4**: Design high-level architecture for the system. | LO1,  LO2 |
| Development | Setting up a mechanism capable of addressing the gaps that were intended to be covered.  **RO1**: To develop data preprocessing component.  **RO2**: To develop a component that handles and stores the top tier transformer architectures for fine-tuning.  **RO3**: To develop the hyperparameter tuning component that handles all the top tier architectures assigned.  **RO4**: To develop a component for the model evaluations for the measured hyperparameters | LO1,  LO5,  LO6 |
| Testing and Evaluation | Testing and evaluating the developed system (including the data science models with the suitable metrices)  **RO1**: Performing unit test, integration and performance testing along with a test plan created.  **RO2**: Evaluating all the transformer architectures used for fine-tune experimentations, using recommended scores such as (ROUGE, BERT SCORE). | LO4 |
| Documenting the progress | Keeping track of and documenting the study project's ongoing progress and any challenges encountered. | LO8, LO6 |
| Publish Findings | Ensure that the documentation, reports, and papers are well-structured and include a critical analysis of the research.  **RO1**: To publish a research paper on the related work done.  **RO2**: To publish the testing & evaluation results of the work done.  **RO3**: To publish the code implementation repository as public to be access by future research investigations, along with the models and datasets. | LO4,  LO8 |

**12. PROJECT SCOPE**

The aim of this project to maximize transformers optimization by hyperparameter tuning, given below are the following scope details for the project objectives to be achieved, along with the review of existing solutions and with the time period taken into consideration.

**12.1. In-Scope**

The project's scope is as follows:

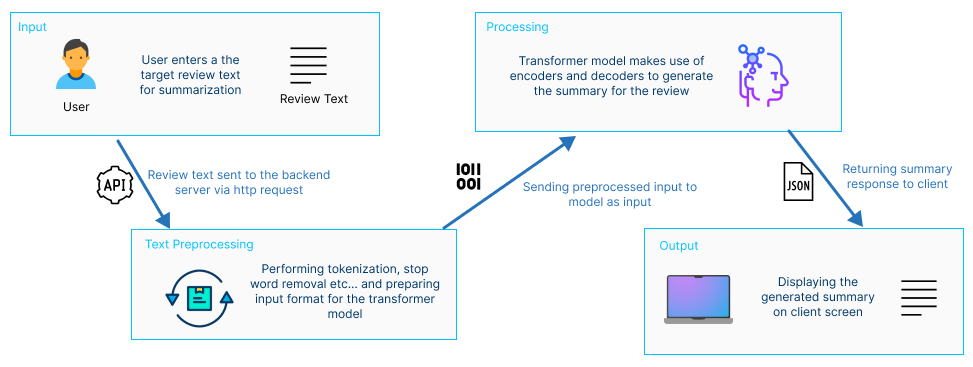
* ***Recreating a usable dataset for the project*** – Reconstructing the dataset to a format structure which can be used for model training.
* ***Model refinement on hyperparameter tuning*** – Performing hyperparameter tuning on the top tier transformer architecture models*.*
* ***Evaluating the models*** – Evaluating all the architectures using appropriate metrics to filter out the best architecture from the rest*.*
* ***API integration development*** – REST API endpoints will be created to serve/call the final chosen model for interactions.
* ***GUI development*** – A graphical user interface will be developed; therefore, the end user will be able to perform abstractive text summarization and get visual results.

**12.2. Out-Scope**

The project will not include the followings:

* ***Limited architecture explored –*** The system will only be explored with few of the top tier architectures (roughly around 3 or 5 maximum), and will not be exploring more than that.
* ***Only single model integration–*** The final model which outperforms the rest with the best set of hyperparameters will be used as the summary generation model, options to select other architectures explored with their hyperparameters aren’t included.

**12.3. Prototype Diagram**

****

*Figure 1.12.1: Prototype Feature Diagram*

**13. PROPOSED METHODOLOGY**

**13.1. Research Methodology**

When determining the quality of a project, there are a number of important factors to consider, including the cost incurred, the amount of time required, and the weight given to the project's scope. These factors must be effectively managed throughout the project's lifespan, which is when methodologies are required.

The table listed below are the chosen methodologies for the project, where Saunders Researched Onion model has been used (Saunders, Lewis, and Thornhill, 2003).

Table 1.13.1: Research Methodology

|  |  |
| --- | --- |
| Research Philosophy | The author will explore and experiment with numerous techniques as part of a combined strategy to determine which is most effective for reaching the research aim, therefore the **pragmatism** approach was chosen among the positivism, pragmatism, realism, and interpretivism approaches. |
| Research Approach | This research experiments with several approaches to figure out the best, the **deductive approach** was taken into consideration this was because the research aims at applying a combination of existing model architectures to fine-tune and get the best. As the data analysis **qualitative method** were chosen. |
| Research Strategy | This area focuses on data collection with respect to the research questions created. S**urvey and experiments** were the strategies considered to address the research questions. Both of these strategies are expected as an approach for the **quantitative result** at evaluation. |
| Research Choice | Weather the research is concerned with qualitative or quantitative aspects depends on the choice of methodology.  Even though we ultimately prioritize quantitative findings mainly, **multi-method** was taken into consideration for this study. This is partly because determining the qualitativeness of the data utilized for development is important since, in the end, it will influence the quantitative outcomes. |
| Time Horizons | **Cross-sectional** will be used since only during the requirement engineering and evaluation phase the data will be gathered and therefore not repeatedly collection over time. |
| Techniques and procedures | Here, data collecting and analysis methods are considered.  We'll utilize sources including internet news, discussions, reports, surveys, publications and organizational records. |

**13.2. Development Methodology**

**13.2.1. Life Cycle model**

The project's research development methodology of choice was the **Agile** Software Development Life Cycle. This is a result of the project's reliance on an iterative development method.

**13.2.2. Design Methodology**

Modularity for flexibility and Code Reusability for efficiency and future development continuity was considered by the author to support incremental methodology, hence **Object-Oriented Analysis and Design** was chosen as the Design Methodology for the project.

**13.2.3. Development Methodology**

**Object oriented programming** methodology will be used for the development methodology for the project, this is due to the project's ease of future developer enhancement, making it simpler.

**13.2.4. Requirement Elicitation Methodology**

Conducting Surveys via questionnaires, review more previous research done, experimenting with various transformer architectures and brainstorming will be the approaches taken in-order to communicate and gather **insights** for the projects need.

**13.2.5. Evaluation Methodology**

**Prototype testing**

A set of test cases will be created in order to test out the entire flow of the prototype with respect to the actual output and whats expected.

**Model testing**

With respect to previous research done (Steinberger and Jezek, 2009) states that for text generated/summarization evaluations using **ROUGE** score demonstrate the best performance compared to other available evaluation methods such as **BLEU**. Therefore, **ROUGE** will be used as the evaluation metric for this project.

**Benchmarking**

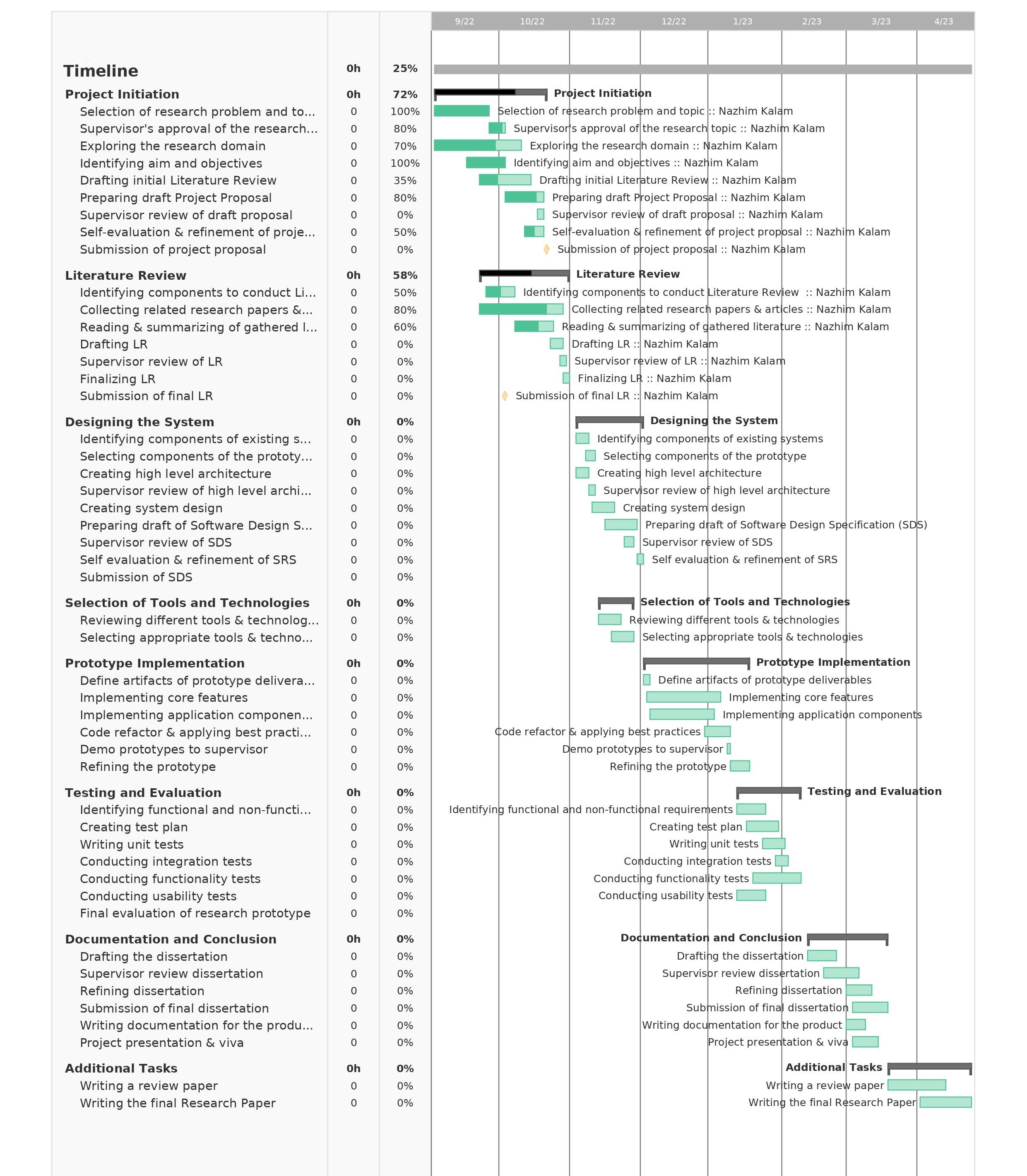
Performance benchmarking is necessary to assess the model's effectiveness on a test data set that replicates the production data, as well as its output speed and memory consumption. Benchmark testing will be done on the final model architecture, which will serve as the optimized model, meanwhile ROUGE will be used for the model evaluation (Steinberger and Jezek, 2009) .

**13.3. Project Management Methodology**

**Prince2** a controlled project management which allows the author to develop environments for different parts or section of the project and maintain, this will be the chosen project management methodology.

**13.3.1 Schedule**

**Gantt Chart**



**Deliverables**

Table 1.13.3: Deliverables and dates

|  |  |
| --- | --- |
| Deliverable | Date |
| **Literature Review Document**  Critical review of existing work and solutions. | 27th October 2022 |
| **Project Proposal Document**  Initial proposal of the project. | 3rd November 2022 |
| **Software Requirement Specification**  Documentation outlining the requirements that must be met, designed as the ultimate prototype, including data collection methods. | 24th November 2022 |
| **System Design Document**  Document outlining the system's design for text summarizing using transformers. | 2nd February 2023 |
| **Prototype**  A functional prototype with all its main features included as stated. | 3rd February 2023 |
| **Thesis**  Final report detailing the research and project decisions | 27th April 2023 |
| **Review Research Paper**  A review paper reviewing published existing systems in handling abstractive text summarization. | 2nd May 2023 |
| **Manuscript Paper**  A research paper introducing the concepts and design developed as part of this project | 10th May 2023 |
| **Final Research Paper**  A research paper about the experimentations done with the transformers hyperparameters. | 15th May 2023 |
| **Public project repository**  A publicly accessible project repository to setup and test the development | 30th April 2023 |

**13.3.2. Resource Requirements**

**Software Requirements**

* ***Operating System* –** Microsoft Windows OS will be used for the research, documentation and for the complete project implementation (end to end).
* ***Python* –** Machine learning & Deep learning model development and APIs creation to serve the models and handle logic will be implemented by using the Python language. Python is a general-purpose language that has been used most widely in data-science related projects and in backend frameworks link Flask and Django.
* ***Flask* –** Backend web framework for API development for the prototype. This will be used to access/transfer data to and from the data science models developed.
* ***TensorFlow/ Scikit learn Python packages* –** Libraries that will be used during the development of the data science models.
* ***Jupyter Notebook / Google Colab* –** Used for Machine-learning/Deep learning model development in this project, it’s an Integrated development environment for programming.
* ***TypeScript (React)* –** JavaScript framework which is used for the development of the frontend application interface of the project. Here is where the user will be able to input and view their data.
* ***Vscode* –** The project's development environment. This will be utilized while creating the codebase for the backend API and frontend development.
* ***Zotero* –** Referencing software that keeps a copy of all the articles as well as managing the references for research papers
* ***MS Office/ Google Docs/ Figma* –** Software & tools which will be used to create figures, reports and handle documentations.
* ***Google Drive/ GitHub* –** Backup platform and code management system to help keep backup of all documents and code.
* ***Git* –** Version control system which will be used to keep track of the changes made in the project code and manage code changes.

**Hardware Requirements**

* **Core i5x Processor (8th generation) or above –** Above average processing power required to perform high resource intensive tasks (such as model training).
* **Nvidia MX130 GPU or above –** To handle data science model training processes.
* **16GB RAM or above –** Sufficient amount of RAM needed to run multiple applications (client + server), model training also consumes a lot of CPU and RAM.
* **Disk space of 30GB or above –** To store project data and applications.

**Data Requirements**

* Amazon Movie review data **–** From Stanford University Education.

**Skill Requirements**

* Good understanding about machine learning and deep learning concepts.
* Good understanding about Natural Language Processing and its data preprocessing methods.
* Good understanding about transformers and how to work with hyperparameters in general along with the knowledge of its use.
* Research writing skills

**13.3.3. Risk Management**

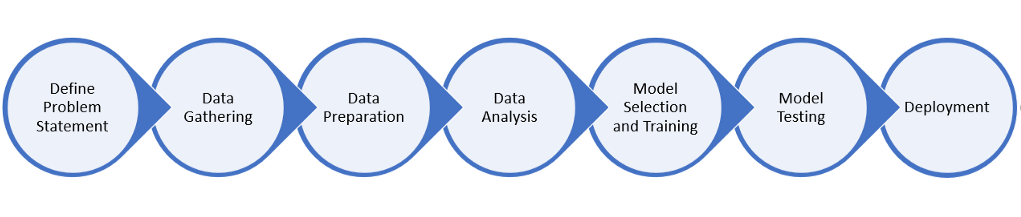
The table given below defines the possible risks which can be encountered during the process of the project development along with the possible mitigation steps.

Table 1.13.3: Risk Mitigation Plan

|  |  |  |  |
| --- | --- | --- | --- |
| Risk | Magnitude of the loss | Probability of occurrence | Mitigation Plan |
| Losing the development project codebase/repository | 5 | 2 | Using GitHub and external backup to keep a latest copy of the project codebase. |
| Project documentation corruption | 5 | 4 | Use a dedicated folder under the same GitHub repository and push all latest documentation changes & use cloud-based documentation approach |
| Unable to complete all mentioned project deliverables on time | 4 | 2 | Prioritize and create a timeline to complete the deliverables. |
| Insufficient knowledge on the project domain | 5 | 3 | Performing an intensive research on the problem domain along with the research domain. |
| Personal computing breaks down during the project timeline progress. | 5 | 4 | Upload the complete backup to GitHub and Google Drive, use University Lab service to continue project work, till personal machine recovery. |
| Any unavoidable personal health risk – Sickness | 3 | 1 | Create weekly goals to complete and keep them updated. |

**13.4. Solution Methodology**

Data collection, data preprocessing, data visualization, model training, model evaluation, and model deployment are the main phases that all machine/deep learning model developments go through. Regarding a few earlier studies on transformers, the author followed the general principles and experimented with a variety of pre-trained transformer models in order to determine which performs the best. (Gupta et al., 2021). The same process will be followed in this project, but the phase for tweaking the hyperparameters will be included as an extra step.

****

*Figure 13.4.1: Summarization model creation flow (patel, 2022)*

**13.4.1 Data Gathering**

Data gathering techniques often fall into one of two categories: primary or secondary. In contrast to secondary data collecting methods, which use data that has already been gathered, primary data refers to information that was obtained directly by the researcher (Wagh, 2022). Secondary data gathering is the method utilized in this project.

**13.4.2. Data Preparation**

This step, often referred to as data preprocessing, involves preparing the data before it is subjected to the training processing by using a series of techniques as follows:

* ***Lower casing***: Creating a common casing out of every text context.
* ***Punctuation removal***: Remove any characters that don't have significant meaning.
* ***Stopwords removal***: Elimination of terms that are used frequently but don't add much sense to the context, such "the" and "a".
* ***Contraction mapping***: Adding more detail to abbreviated words like "don't" into "do not".
* ***Stemming***: A method of reducing a word to its word stem.
* ***Lemmatization***: Grouping different inflected of words into the root form.

**13.4.3. Data Analysis**

This procedure, also known as data visualization, involves graphically depicting the data (using maps, graphs, and charts) and identifying data patterns. (Analytics Vidhya, 2021). Where necessary, data visualization will be used in this project.

**13.4.4. Model Selection and Training**

The author will experiment with tuning hyperparameters while utilizing various pretrained model transformer architectures. This transformer architecture is the optimum option since it dominates the field of natural language processing. Using pertained model, the author is able to achieve stronger performance than creating a model from scratch as a result of their extensive corpus data training. (Wolf et al., 2020)

Since they are the top tier model architectures, the pre-trained models chosen for hyperparameter tweaking will include BART, T5, PEGASUS, RoBERTa, and GPT-3.

***Hyperparameter Optimization***

This set of parameters needs to be calculated for optimal performance, so model training parameters could be set to them. Minimizing the loss and cost has the potential to stabilize the bias and variance in the model. Additionally, the chosen dataset will have an impact on the estimated set of hyperparameters.

**13.4.5. Model Evaluation and Deployment**

There are several forms of evaluation metrics for text summarization models, including BLEU and ROUGE. Since ROUGE is more credible than BLEU, it will be utilized for model evaluation and choosing the optimal model architecture to apply.(Steinberger and Jezek, 2009)

The model deployment will be done together with the backend server and hosted in cloud platforms.

**14. REFERENCES**

‌Alsaqer, A.F. and Sasi, S. (2017). Movie review summarization and sentiment analysis using rapidminer. *2017 International Conference on Networks & Advances in Computational Technologies (NetACT)*. July 2017. Thiruvanthapuram, India: IEEE, 329–335. Available from https://doi.org/10.1109/NETACT.2017.8076790 [Accessed 10 October 2022].

Boorugu, R., Ramesh, G. and Madhavi, K. (2019). Summarizing Product Reviews Using Nlp Based Text Summarization. *International Journal of Scientific & Technology Research Volume*, 8 (10), 1127–1133.

Etemad, A.G., Abidi, A.I. and Chhabra, M. (2021). A Review on Abstractive Text Summarization Using Deep Learning. *2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*. 3 September 2021. Noida, India: IEEE, 1–6. Available from https://doi.org/10.1109/ICRITO51393.2021.9596500 [Accessed 10 October 2022].

Gupta, A. et al. (2021). Automated News Summarization Using Transformers. *ArXiv*, abs/2108.01064.

Khan, A. et al. (2020). Movie Review Summarization Using Supervised Learning and Graph-Based Ranking Algorithm. *Computational Intelligence and Neuroscience*, 2020, 7526580. Available from https://doi.org/10.1155/2020/7526580.

Mahajan, R. et al. (2021). Text Summarization Using Deep Learning. *International Research Journal of Engineering and Technology (IRJET)*, 08 (05th May 2021), 1737–1740.

Mukherjee, R. et al. (2020). Read what you need: Controllable Aspect-based Opinion Summarization of Tourist Reviews. *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 25 July 2020. 1825–1828. Available from https://doi.org/10.1145/3397271.3401269 [Accessed 10 October 2022].

Steinberger, J. and Jezek, K. (2009). Evaluation Measures for Text Summarization. *Comput. Informatics*, 28 (2), 251–275.

Wolf, T. et al. (2020). Transformers: State-of-the-Art Natural Language Processing. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. 2020. Online: Association for Computational Linguistics, 38–45. Available from https://doi.org/10.18653/v1/2020.emnlp-demos.6 [Accessed 10 October 2022].

// external

patel, S. (2022). Life Cycle for Machine Learning Problem — Beginner Writes. [online] Medium. Available at: https://pub.towardsai.net/life-cycle-for-machine-learning-problem-1940014eda1 [Accessed 17 Oct. 2022].

‌Wagh, S. (2022). Research Guides: Public Health Research Guide: Primary & Secondary Data Definitions. [online] researchguides.ben.edu. Available at: https://researchguides.ben.edu/c.php?g=282050&p=4036581.

‌

Analytics Vidhya. (2021). *What is Data Visualization | Data Visualization Techniques*. [online] Available at: https://www.analyticsvidhya.com/blog/2021/06/must-known-data-visualization-techniques-for-data-science/.

‌

‌

‌