End term T12 - Project Phase II

on

Sentiment Analysis and Summarization of Product Reviews using Deep Learning Techniques

Submitted by

Project Members

Swaroop Nayak 1032180202
Kartik Bhutada 1032180229
Amey Bhide 1032180301
Divyang Bagla 1032180739

Under the Internal Guidance of

Prof. Rashmi Rane

School of Computer Engineering and Technology
MIT World Peace University, Kothrud,
Pune 411038, Maharashtra - India

2021-2022

SCHOOL OF COMPUTER ENGINEERING AND TECHNOLOGY

C E R T I F I C A T E

This is to certify that,

Swaroop Nayak 1032180202
Kartik Bhutada 1032180229
Amey Bhide 1032180301
Divyang Bagla 1032180739

of BTech.(Computer Science & Engineering) have completed their project titled "**Sentiment Analysis and Summarization of Product Reviews using Deep Learning Techniques**" and have submitted this Capstone Project Report towards fulfillment of the requirement for the Degree-Bachelor of Computer Science & Engineering (BTech-CSE) for the academic year 2021-2022.

[Prof. Rashmi Rane]
Project Guide
School of CET
MIT World Peace University, Pune

[Dr. Vrushali Kulkarni]
Head of School
School of CET
MIT World Peace University, Pune

Internal Examiner: ______________________

External Examiner: ______________________

Date:

Acknowledgement

An endeavour over a long period can be advice and support of many well wishers. We take this opportunity to express our gratitude and appreciation to all of them. We owe our tributes to Dr. Vaishali Kulkarni, Head of the Department, Computer Science & Engineering for her valuable support and guidance during the period of project implementation. We wish to express our sincere thanks and gratitude to our project guide Mrs. Rashmi Rane, department of computer science and engineering. Project meetings were highly informative. We express our sincere thanks for the encouragement, untiring guidance and the confidence they had shown in us. We are immensely indebted for their valuable guidance throughout our project. We also thank all the staff members of the CSE department for their valuable advice. We also thank the Principal and supporting staff for providing resources as and when required.

Name of the Students

Swaroop Nayak

Kartik Bhutada

Amey Bhide

Divyang Bagla

Abstract

Online Customer reviews is a crucial part of the decision making process of buying products from ecommerce. Each product has hundreds of reviews, making it difficult for a consumer to make a decision due to mixed feedback. It takes a long time to read all reviews, and occasionally you end up making decisions based on just a couple of reviews. As a solution, we are proposing automatic review summarization, which will be used to analyze product reviews and transform them into a user-readable, more concise and precise format, allowing consumers to save time and effort by not having to browse through all of the reviews. In order to achieve the above goal, we have implemented a LSTM model for the sentiment analysis, followed by clustering and weak-reference extraction for the processing of the summarization, and finally for summarization we have used BERT model for extractive summarization

Keywords— *Review Summarization, Multi-document Summarization, Extractive Summarization, LSTM, BERT, Clustering, Sentiment Analysis*

List of Figures

List of Tables

Contents

Chapter 1

Introduction

1.1 Project statement

Sentiment Analysis and Summarization of Product Reviews using Deep Learning Techniques

1.2 Area

1. Natural Language Processing (NLP)
2. Machine Learning (ML)

1.3 Project Introduction and Aim

- As potential customers, people usually seek help from the online portals to gain knowledge on a particular product, and finally, decide if the purchase should be made or not.
- It takes several hours to read all the reviews, sometimes even leading to missing out the important ones, thus ending up making the 36wrong decision on purchasing the product.
- Each product has thousands of reviews each, and it is tedious for the customer to make a decision based on the varying user reviews.
- A more well defined and concise product review is proposed such that the user need not skim through all the reviews, thus saving their time and effort.
- As a solution to the ongoing problem that the customer experiences daily, automatic review summarization will be used to analyze the product reviews and convert them into a user-readable and in a more concise and precise format.

Chapter 2

Literature Survey

As mentioned, our project is implemented in two phases: Classification and then Summarization of the reviews. Recent works in Product Reviews Classification has been done in [1][2][6] where they have used various machine learning models such as decision tree, logistic regression, multinomial NB, SVM, BERT and LSTM and compared then and analyzed that the BERT model has achieved highest accuracy of around 98.51%.

The use of a Convolutional Neural Network, BOW, to execute Word Vector with 97 percent accuracy and higher performance in all areas was discussed by Li Rong et al [3]. The only obstacle being here is the dataset used here is a Chinese Character Dataset in which case the model performance may give different results for reviews in English.

Aishwarya et al [4] has talked about Summarization and Prioritization of Amazon Reviews based on multi-level credibility attributes using LSTM, NLTK, TF-IDF. This paper also discusses how we can find the credibility of reviews and summarizes the next reviews based on the positive and negative keywords. The average accuracy achieved via LSTM was around 90.28%.

P. Porntrakoon et al., [5] used Simplified Sentiment Analysis, multidimensional lexicon to get the summarized Thai food reviews. The average accuracy in analyzing the negative, positive and all sentiments is 31.96, 90.26 and 85.05 respectively.

For text summarization, various resources have been cited for proposed implementation. The paper regarding extractive text summarization for multi-document summarization has been discussed by Tohalino et al [12]. using multi-layer networks and pagerank algorithm.

For abstractive text summarization, we have referred to Shapira et al., [8] which uses clustering and weak reference extraction and Fast Abstractive Summarization (FAS) for abstractive massive-multi document summarization (MMDS). We have taken this as our base paper to implement review clustering and weak reference extraction as part of our implementation. Similarly, J. Shah et al., [7] has discussed NLP based abstractive Text Summarization of reviews using NLP based Seq2seq encoders with attention layers but the dataset to which model is applied is of one category only and model to implement for various categories of products has not been discussed.

There are also works which explore end to end models for simultaneous execution of classification as well as review summarization [9,11] but for proposed implementation becomes computation heavy and a more well refined encoder-decoder transformer like BERT is more suitable for our purpose.

TABLE I. ACCURACY VALUES FOR PRODUCT SENTIMENT ANALYSIS

Model	Precision	Recall	F1 Score
LSTM [1]	0.97	0.97	0.97
Bidirectional Encoder Representations from Transformers (BERT) [2]	0.98	0.98	0.98
Fully Connected Neural Network (FCNN) [3]	0.81	0.97	0.92
TF-IDF + Chi2 [6]	0.82	0.88	0.86

The Table I shows accuracy parameter values such as Precision, Recall and F1 Score for the various references referred for Product Reviews Sentiment and it can be concluded that the maximum accuracy for product reviews sentiment analysis is for [2] which uses LSTM model for Sentiment Analysis and forms the base for our 1st phase.

For the second phase of this project we have found that for Multi-Document Summarization, we have performed clustering of reviews to implement weak reference extraction techniques[8] to find the summary of the clusters using Fast Abstractive Summary(FAS) technique. We form this as our base paper for the Summarization phase of our implementation.

Other works related to review summarization include the use of a shared text encoder[9, 11] where joint work of text summarization and classification is done.

As for the implementation, more work has been done in the extractive summarization of reviews, we implement the BERT model for summarization by performing weak reference extraction as it is more refined and researched on.

Chapter 3
Problem Statement

3.1 Project Scope

- Our opinion and purchasing decision-making are affected by the experience of others and their feedback about products. We always ask others about their opinion to get the benefit from their experience; hence, the importance of reviews has grown.
- Nowadays, a company can easily collect reviews from users via e-commerce platforms and recommender systems, but it is difficult to read through all the wordy user reviews. Therefore, distilling salient information from user reviews is necessary.
- So there is a need to process the maximum amount of information in the least amount of time.
- It is critical to analyze such feedback due to the volume and redundancy. This work investigates an efficient way to analyze such feedback and solve the problems related to the classification and summarization of app reviews.

3.2 Project Assumptions

- All the reviews should be in the English language, as the model will be trained on the dataset for reviews mentioned in English only.
- The reviews must contain the alphabetical data and should not consist of any special characters like emojis etc.
- Reviews that will be used for model training will only be from trusted users such as the "verified purchase" tag in amazon.

3.3 Project Limitations

- Multilanguage Reviews summarization is not compatible with our proposed system.
- LSTM is computationally expensive and it takes a lot of time to train the dataset.
- Currently, we are testing our model only for e-commerce products, the project has to be extended for other reviews as well such as movie reviews, book reviews, travel reviews and other types of reviews.

3.4 Project Objectives

- To provide a general overview of a product, having tons of reviews available so as to take a quick glance with both pros and cons of any product.
- To provide a brief abstractive summary of a product with a large number of reviews available in order to take a quick glance at both the pros and cons of any product.
- We will tackle the above problem using deep learning techniques like LSTM-attention mechanism and/or BERT.
- We will have 2 phases: classification and summarization
- Processing techniques will be the same for summarization, we only have to perform processing once.

Chapter 4
Project Requirements

4.1 Resources

4.1.1 Reusable Software Components

NLTK Packages

Tensorflow Packages

Lemmatization Function

Vectorization

4.1.2 Software & hardware requirements

Python

Jupyter Notebook

Google Colab

BERT

RAM: 8 GB

CPU: AMD Ryzen 7 4800H

4.2 Requirements Rationale

TABLE II. REQUIREMENTS RATIONALE

Requirement	**Rationale**
Results should be displayed in a reasonable amount of time (depending on the input data)	Issue: how long does it take before the user gets impatient Position 1: result is displayed in a reasonable amount of time Position 2: result is taking longer than expected time to generate result Argument: We don't want the user to get impatient Assumption: Any product which loses the attention of the user tends to go bust Decision 1: this requirement is mandatory

	and nonnegotiable Decision 2: generate results in a reasonable amount of time sacrificing.

4.3 Risk Management

TABLE III. RISK MANAGEMENT

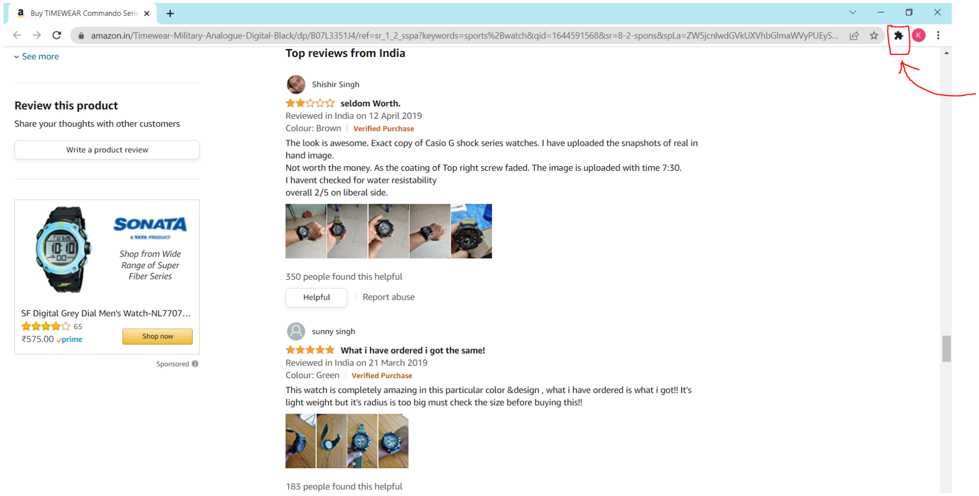
Probability	**Risk**	**Mitigation**
High	Time for result generation is high	Develop the product further focusing on the optimization of the algorithms used
Medium	Insufficient Data Input	Perform Data Augmentation to artificially increase the size of data input
Low	Similar product exists	Change the approach and methodology of solving the problem statement

4.4 Functional Specifications:

4.4.1 Interfaces

4.4.1.1 External interfaces required

The external interface will be an icon in the browser. While using the browser, a user's click will trigger a panel containing a button.



4.4.1.2 Internal interfaces required

- Sentiment Analysis Model
- Text Summarization Model
- Server which will take reviews from the client and pass on to Models.

4.4.1.3 Communication interfaces

The extension and the server are the only two points of contact. JQueryAJAX will be implemented. It's used to send and receive requests. The protocol used will be HTTP.

4.4.1.4 Graphical User interface

Chrome Extension's Graphical User Interface

4.4.2 Interactions

- User clicks the "View Summary" button.
- System takes the web page and sends it to the cloud server.
- Web page is parsed, and the summarization algorithm takes its body part as input.
- The algorithm then sends the summary sentences to the extension. The extension shows the summary of the reviews in pros and cons form.

Chapter 5

System Analysis Proposed Architecture/ high level design of the project

5.1 Design Consideration

- Functionality
- Information Design
- Data Science - Extracting knowledge data and communicating it in a way that is not misleading.

5.2 System Architecture

5.2.1 High Level Design

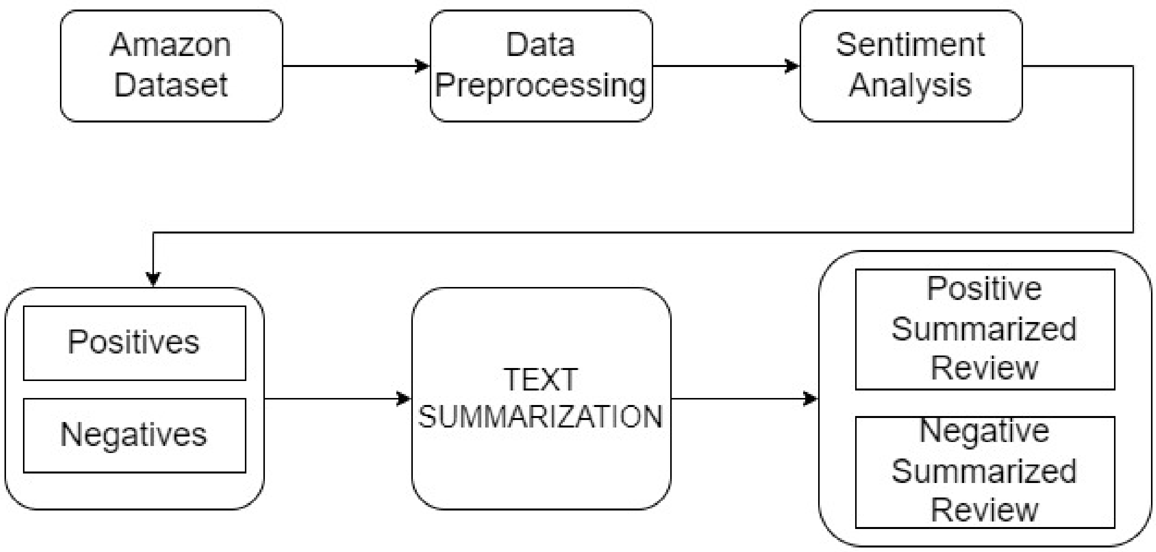


Fig. 1 High Level Diagram

5.3 Low level Design

5.3.1 Low Level - Pre Processing

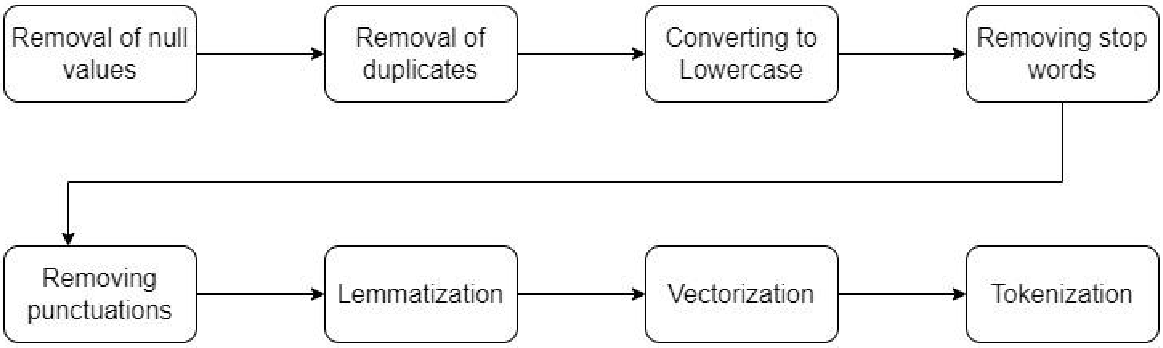


Fig. 2 Low Level Diagram - PreProcessing

5.3.2 Low Level - Text Summarization

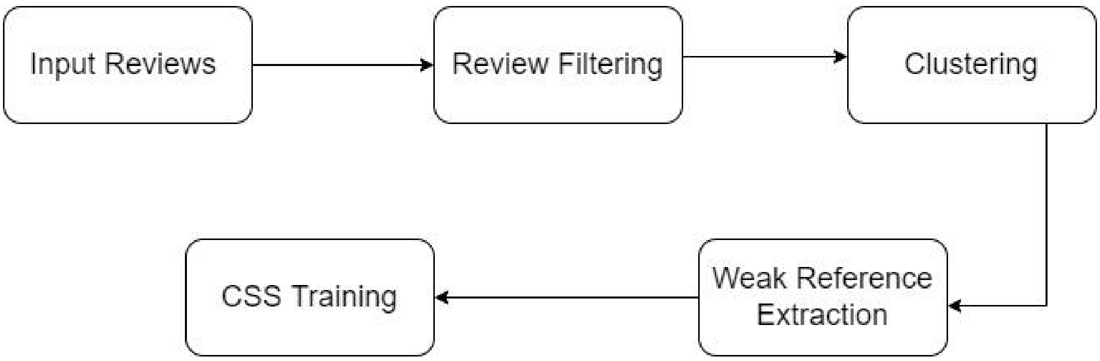


Fig. 3 Low Level Diagram - Text Summarization

5.4 UML Diagrams/Agile Framework

5.4.1 Use Case Diagram

This viewpoint focuses on how different users interact with the system.

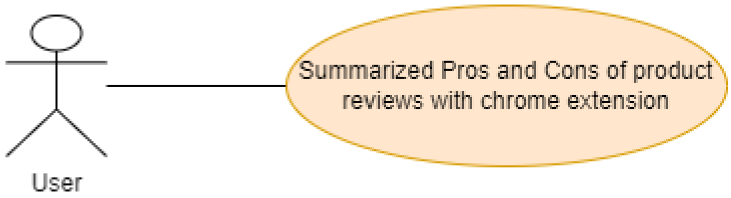


Fig. 4 Use Case Diagram - 1

TABLE IV. USE CASE - 1

Use case Id	1
Use case	Summarize Product Reviews of a web page using Chrome Extension
Description	Getting summary of a ecommerce review webpage and showing it in the form of pros and cons without any selections on the webpage
Actor	User
Trigger	User clicks the “Summarize Reviews” button in the extension popup.
Primary Scenario	- User clicks the “View Summary” button. - System takes the web page and sends it to the cloud server. - Web page is parsed, and the summarization algorithm takes its body part as input. - The algorithm then sends the

	summary sentences to the extension. The extension shows the summary of the website.

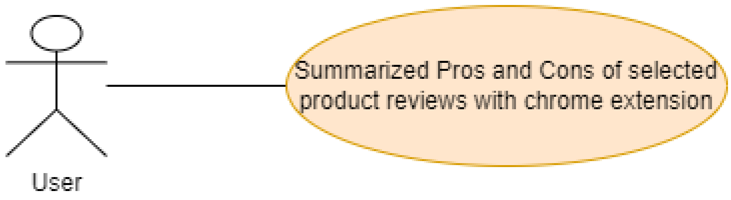


Fig. 5 Use Case Diagram - 2

TABLE V. USE CASE - 2

Use case Id	2
Use case	Summarize Product Reviews of selected reviews using Chrome Extension
Description	Getting summary of a ecommerce review webpage and showing it in the form of pros and cons of selected reviews on the webpage
Actor	User
Trigger	- User selects some parts of the text on the webpage, by using the cursor - User clicks the "Summarize Reviews" button in the extension popup.
Primary Scenario	- User clicks the "View Summary" button. - System takes the sentences that the user selected. - These sentences are submitted to the

	cloud server via the extension. - These sentences are fed into a cloud-based algorithm, which generates summary sentences. - Summary sentences are then sent back to the extension

5.4.2 Deployment Diagram

This viewpoint focuses on the structure of the system and provides a top level view of the entire system from the perspective of each component. For this aim, the UML Deployment Diagram is provided that can be seen as below.

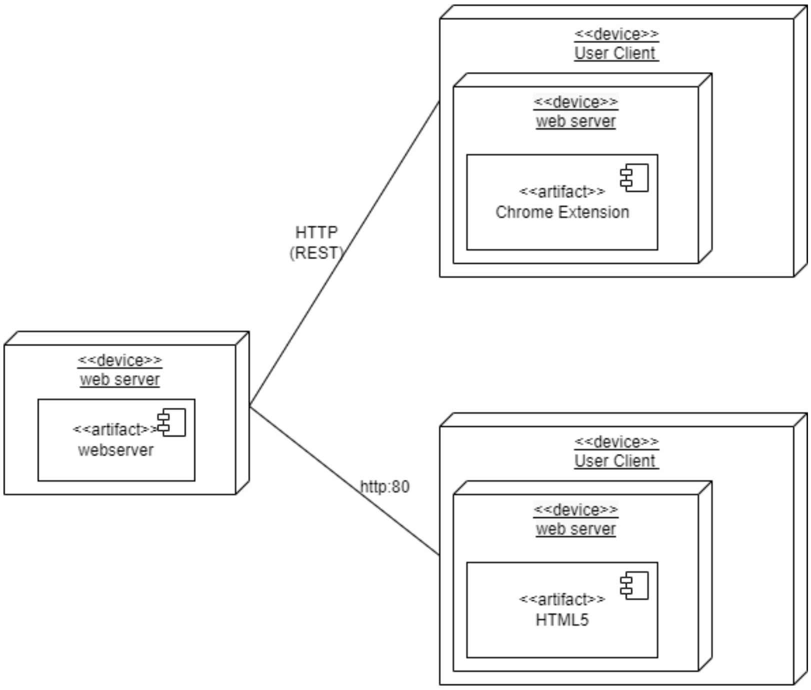


Fig. 6 Deployment Diagram

5.4.3 Sequence Diagram

In this view point the interaction among entities of the system will be visualized. UML Sequence Diagrams are used in order to provide representation of the interaction.

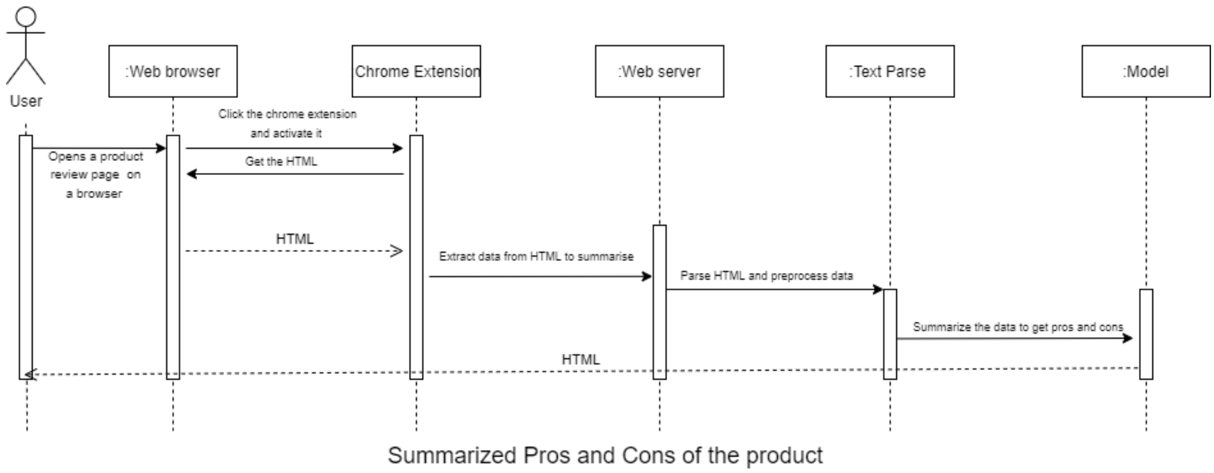


Fig. 7 Sequence Diagram

Chapter 6

Project Plan

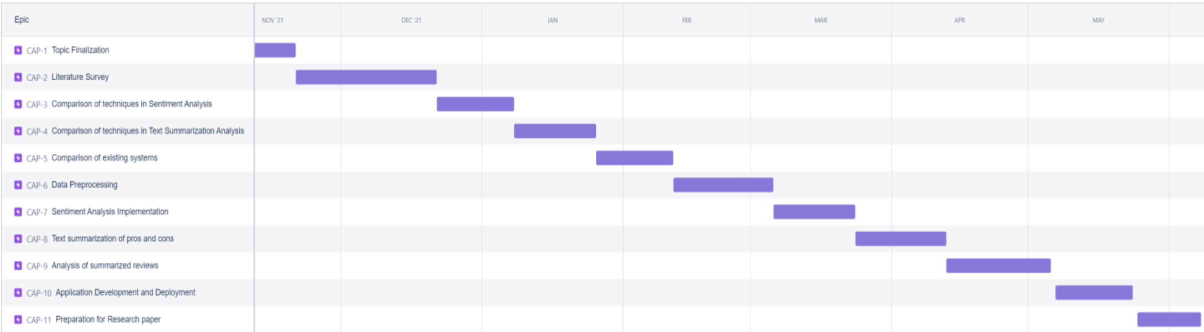


Fig. 8 Gantt Chart

Link : [Timeline]

Chapter 7
Implementation

7.1 Methodology

As we have seen that deep learning models provide higher accuracy for Sentiment Analysis and Text Classification, for the implementation of the proposed work we have used LSTM model for classification to apply binary classification of reviews as positive and negative reviews and later, we have used BERT model for Review Classification by using clustering and weak reference extraction methods for the classified reviews separately for the purpose of data generation and for the generation of output.

7.1.1 Dataset

We used the open source Amazon review dataset [13]. The dataset includes user-generated reviews (ratings, text, and helpfulness votes), product metadata (descriptions, category information, price, brand, and picture attributes), and linkages (also viewed/also bought graphs) from the Amazon e-commerce website.

This dataset includes reviews metadata like ratings, text, helpfulness, votes. Product metadata like descriptions,category information, price, brand, etc. The format of this dataset is one review per line in JSON. This dataset contains reviews of each category of products like books, electronics, movies, sports etc. This dataset contains 34 gb of data and contains 233.1 million reviews.

As the original data is too large for our needs and out of our computational limits to run through all the data, we have taken a sample of the k-cores dataset where k=5. These data have been reduced to extract the k-core, resulting in k reviews for each of the remaining people and objects. We selected a random sample of 2000 reviews from each product category. There are 29 different categories to choose from.

```
{
  "reviewerID": "A2SUAM1J3GNN3B",
  "asin": "0000013714",
  "reviewerName": "J. McDonald",
  "vote": 5,
  "style": {
    "Format:": "Hardcover"
  },
  "reviewText": "I bought this for my husband who plays the piano.
He is having a wonderful time playing these old hymns.  The music  is
at times hard to read because we think the book was published for
singing from more than playing from.  Great purchase though!",
  "overall": 5.0,
  "summary": "Heavenly Highway Hymns",
  "unixReviewTime": 1252800000,
  "reviewTime": "09 13, 2009"
}
```


Fig. 9 Dataset Snapshot

7.1.2 Preprocessing

In Fig. 1, we have discussed the steps performed for the preprocessing of the reviews. First, we have converted all the text to lowercase, removed punctuations and digits, as a form of normalization. Next, we have removed duplicates and empty tokens as they are not relevant to the training of the model during sentiment analysis. Empty tokens were formed as by-products of the above steps. Next, we handle and add the Parts of Speech tag text so as to identify the verbs, adjectives, nouns, and other parts of speech, which helps in the lemmatization process as it helps differentiate between the vocabulary and the unique words such as places, names, etc. After this step, as mentioned before, we have performed lemmatization. The process of grouping together the various inflected forms of a word so that they can be analyzed as a single item is known as lemmatization. Due to the previous few steps, there are some words formed which just have a length of 1. As these were not relevant in the training phase, they were removed. Finally, the text is converted to a matrix of integers using the vectorization, which will be fed into the sentiment analysis model.

7.1.3 Sentiment Analysis

For the first phase, we have performed sentiment analysis on the incoming data which would later be subjected to text summarization in order to obtain the final results. For the sentiment analysis, we have split the data in a 70:30 ratio of training and test set respectively. For the model itself, we have used a LSTM neural network.

The model itself is a 4-layer, 511,194 trainable parameters LSTM neural network architecture. The first layer is the embedding layer with an input dimension of 2000, and an output dimension of 128. The next layer is a spatial dropout 1D with a rate of 0.4. The third layer is the LSTM layer of size of 196 and a dropout of 0.2. The final layer is dense with 2 outputs and the optimizer used is the 'adam' optimizer. The metrics used is accuracy and the loss function used is the categorical cross entropy.the accuracy obtained was 0.9485.

After the data is run through the model, it is categorical in different locations based on its output i.e. the positive and the negative reviews are separated and stored separately for further processes.

7.1.4 Clustering and Weak Reference Extraction

As we were using data dealing with different reviews, we have followed the approach of massive multi-document summarization (MMDS) [8]. In MMDS, the data itself cannot be directly fed to the summarizer as the summarizer input is not large enough to accept all the data and either lead to errors or loss of data or both.

In order to train/test the summarizer, we need to have the right data for it. As there is no available data for this purpose, we have to create and restructure from existing data and work forward with it. The method we have used is the clustering and weak-reference extraction technique as present in [8].

Clustering: It is a collection of data which are resembling each other the closest. A form of pivot clustering has been used. It means to select a randomly assigned pivot point in the dataset. We then find the ROUGE-1 F1 score of all the data in the dataset and the data points having the highest score are added in the cluster along with the pivot data.

For the pre-processing step, we remove all the reviews having less than 15 tokens as they are not relevant based on assumptions. tokens, assuming their helpfulness is negligible. We initialize the unclustered review set, U, to the set R. Then, while U is not empty, we randomly choose a pivot review p and build a singleton cluster Cp = {p}. We then compute the ROUGE-1 F1 scores between p and all other reviews, and repeatedly add reviews to Cp, starting from the top-scoring review and moving down the scores, until Cp contains min-rev reviews, and then continue to add reviews while the accumulated text length, is below a predefined threshold max-token-len, where the text length is measured in tokens. In this project, we have taken max-token-len as 512 and min-rev as 3.

Weak-reference extraction: In a given cluster, we find the similarity function of every data point in the cluster with each other. The similarity function uses ROUGE-1 F1 score. The data point which has the highest average ROUGE-1 F1 score will be selected as the weak reference extraction.

7.1.5 Review Summarization

Now we can collect the data required to train/test the summarizer. For this phase, we have chosen the BERT extractive summarizer. As this is a pre-trained model, there is minimal need to train the model. We just needed to tune in the hyperparameters. After testing with several hyperparameters, we came to a conclusion that the default parameters gave the best results as per the metrics used. In the end, the results are displayed in Table II.

TABLE VI. ROUGE SCORES OF THE BERT EXTRACTIVE SUMMARIZER

	Recall	Precision	F1 score
ROUGE-1	0.50146	0.19342	0.27379
ROUGE-2	0.24515	0.08510	0.12297
ROUGE-L	0.47792	0.18467	0.26125

The dataflow for data generation and the summarization is a bit different and has been shown in Figure.3.

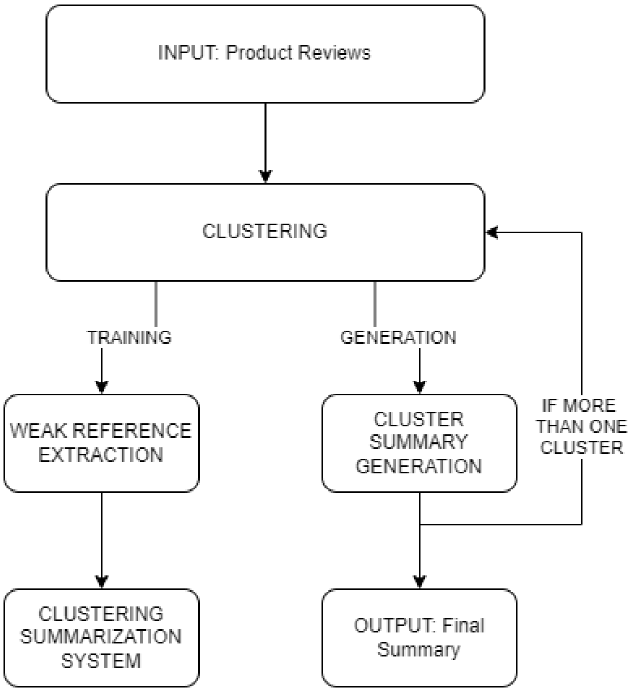


Fig. 10 Low level Diagram of the summarizer training and summary generation

7.1.6 Summary Generation

The process of summary generation starts with the clustering step. In this step, we do not take any min-rev parameters. About the max-token-len, it stays the same i.e. 512 as it is the maximum number of tokens which a BERT model can take in at a single time. We create the clusters based on the above parameters and then proceed to find the summary of the said cluster. Next the summaries are clustered together within the constraints of the parameters provided. Then we find the summaries of the said clusters to produce second-level summaries. This procedure is recursively applied until a final summary emerges.

7.2 Algorithms

7.2.1 Preprocessing Techniques:-

1. Lemmatization : This is done to get the terms back to their root form so that they can be deemed the same if they are used in different ways. For example, doing, done, and doing.
2. Vectorization : Because machine learning algorithms cannot function with text form data, a method for converting textual input into a numerical form (binary, int, float) is required. This is done using vectorization.
3. Tokenization : Tokenization is the process of breaking down a phrase, sentence, paragraph, or even an entire text document into smaller components like individual words or phrases. Tokens are the names given to each of these smaller units. 'Natural language Processing', for example, => ['Natural','language','processing'].

7.2.2 Deep Learning Models Used

1. **BERT Architecture**

- Bidirectional Encoder Representations from Transformers is a pre-trained deep learning NLP model and it has two variations, namely, a base model and a large model.
- We have built a sentiment classifier using the base model, which is a 12-layer, 768-hidden, 12-heads, 110 million parameter neural network architecture.
- The pre-processed reviews were fed to the model using ‘Adam’ as the optimizer, ‘SparseCategoricalAccuracy’ as the accuracy metric, and ‘Categorical_Crossentropy’ as the loss function.
- For our dataset, we have fine-tuned the model up to 2 epochs, which gives us a validation accuracy of 98.51%.

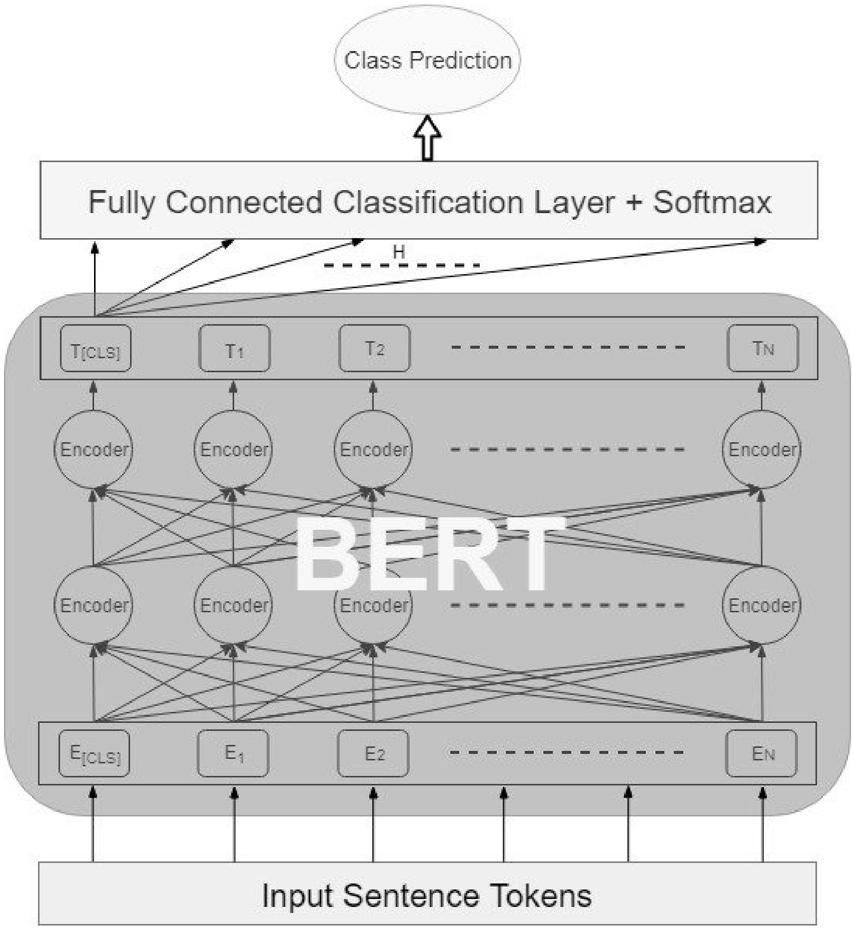


Fig. 11 BERT Architecture

2. LSTM Architecture

- To overcome the vanishing gradient problem, LSTM is an improved form of Recurrent Neural Network. An explanation of the LSTM architecture can be found below. It has a memory cell on top that aids in the efficient transfer of information from one time instance to the next.
- As a result, as compared to RNN, it is capable of remembering a large amount of information from earlier states and avoids the vanishing gradient problem. With the use of valves, information can be added to or withdrawn from a memory cell.
- We have built our model on top of the Sequential class of keras, where we have used the LSTM layer as a bidirectional layer, which is added along with the Embedding and Dense layers on top of the Sequential class.
- The pre-processed reviews were fed to the model using ‘Adam’ as the optimizer, and ‘binary_crossentropy’ as the loss function.

- For our dataset, we have fine-tuned the model up to 20 epochs, which gives us a validation accuracy of 97.35%.

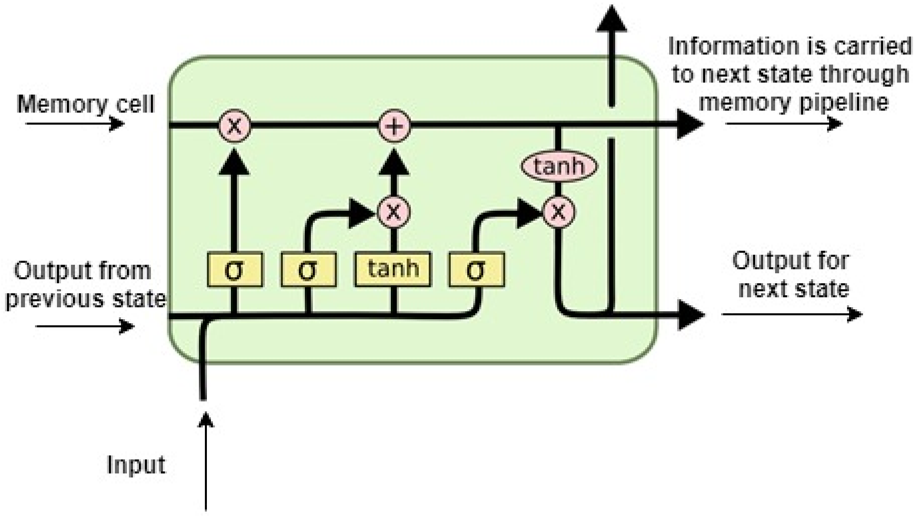


Fig. 12 LSTM Architecture

3. Clustering and Weak Reference Extraction

- In a given cluster, we find the similarity function of every data point in the cluster with each other.
- The similarity function uses ROUGE-1 F1 score. The data point which has the highest average ROUGE-1 F1 score will be selected as the weak reference extraction.

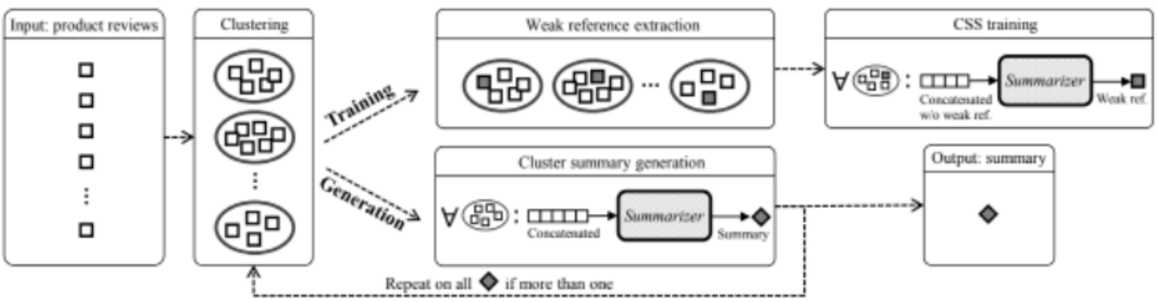


Fig. 13 An illustration of our MMDS training and hierarchical cluster summarization schema.

Chapter 8
Result and Analysis

We have created an end-to-end process for sentiment analysis and text summarization. For sentiment analysis, we have built a sentiment classifier using the LSTM model, which has a 4-layer, 511,194 trainable parameters neural network architecture. The pre-processed reviews were fed to the model using 'Adam' as the optimizer, 'Accuracy' as the accuracy metric, and 'Categorical_Crossentropy' as the loss function. For our dataset, we have fine-tuned the model up to 10 epochs, which gives us a validation accuracy of 94.85%. For the text summarization, we have used the BERT extractive model with default hyperparameters. The results can be found in Table VI.

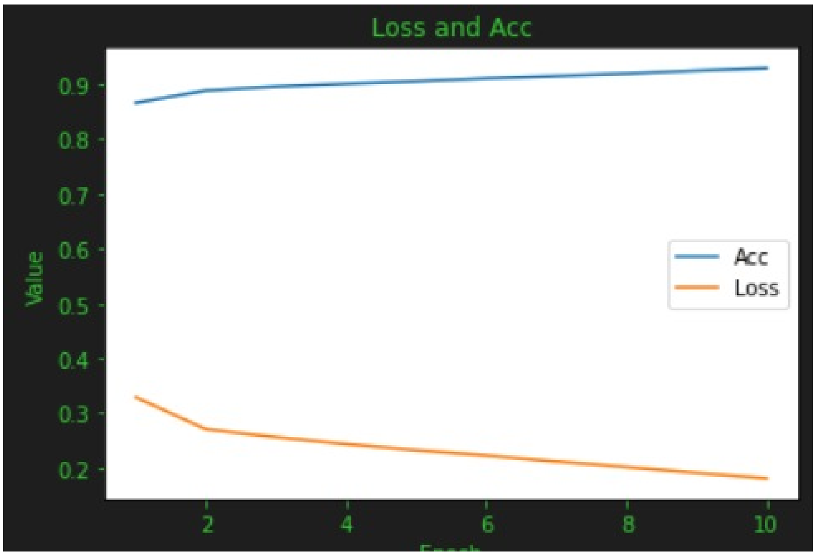


Fig. 14 Loss and Accuracy Curve

```
confusion matrix [[ 1341   861]
 [  460 11867]]
              precision    recall  f1-score   support

           0       0.74      0.61      0.67      2202
           1       0.93      0.96      0.95     12327

    accuracy                           0.91     14529
   macro avg       0.84      0.79      0.81     14529
weighted avg       0.90      0.91      0.91     14529
```

Fig. 15 Confusion Matrix

APPLICATIONS

There is a potential for this project to be implemented and deployed in a web extension which scrapes the reviews from popular e-commerce sites such as Amazon and then finds the summary of the reviews of the product. Text summarization is currently computationally heavy, so the work remains to increase its efficiency and make sure it takes less time to fetch reviews and give the summarized results.

E-commerce Sites can add this extra feature to help their customers have an overall review of the product. They can also use this tool to analyze products and to find an overall sentiment about the product.

CONCLUSION

We have conducted an extensive literature review of multiple research papers. Along with that, we have also compared multiple methodologies by doing a comparative analysis and have shortlisted our conclusions. The first phase of the project development has been started working on i.e. the data collection and preprocessing. From the initial phases of this project, we have a clear understanding of what we are trying to achieve and have set definitive goals and parameters to this project. We have also made a SDLC chart to help streamline our development process.

References

[1] Desai, Z., Anklesaria, K., & Balasubramaniam, H. (2021, July). Business Intelligence Visualization Using Deep Learning Based Sentiment Analysis on Amazon Review Data. In *2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT)* (pp. 1-7). IEEE.

[2] AlQahtani, A. S. (2021). Product Sentiment Analysis for Amazon Reviews. *International Journal of Computer Science & Information Technology (IJCSIT) Vol*, *13*.

[3] L. Rong, Z. Weibai and H. Debo, "Sentiment Analysis of Ecommerce Product Review Data Based on Deep Learning," 2021 IEEE 4th Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC), 2021, pp. 65-68, doi: 10.1109/IMCEC51613.2021.9482223.

[4] Aishwarya, N., Bhuvana, L. S., & Kayarvizhy, N. (2021, August). Summarization and Prioritization of Amazon Reviews based on multi-level credibility attributes. In *2021 International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT)* (pp. 5-9). IEEE

[5] P. Porntrakoon, C. Moemeng and P. Santiprabhob, "Text Summarization for Thai Food Reviews using Simplified Sentiment Analysis," 2021 18th International Joint Conference on Computer Science and Software Engineering (JCSSE), 2021, pp. 1-5, doi: 10.1109/JCSSE53117.2021.9493839.

[6] F. Rustam, A. Mehmood, M. Ahmad, S. Ullah, D. M. Khan and G. S. Choi, "Classification of Shopify App User Reviews Using Novel Multi Text Features," in IEEE Access, vol. 8, pp. 30234-30244, 2020, doi: 10.1109/ACCESS.2020.2972632.

[7] J. Shah, M. Sagathiya, K. Redij and V. Hole, "Natural Language Processing based Abstractive Text Summarization of Reviews," 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), 2020, pp. 461-466, doi: 10.1109/ICESC48915.2020.9155759.

[8] Shapira, O., & Levy, R. (2020). Massive multi-document summarization of product reviews with weak supervision. *arXiv preprint arXiv:2007.11348*.

[9] Chan, H. P., Chen, W., & King, I. (2020, July). A unified dual-view model for review summarization and sentiment classification with inconsistency loss. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 1191-1200).

[10] Ravali Boorugu, Dr. Gajula Ramesh, Dr. Karanam Madhavi (2019). Summarizing Product Reviews Using NLP Based Text Summarization. International Journal of Scientific & Technology Research

[11] Ma, S., Sun, X., Lin, J., & Ren, X. (2018). A hierarchical end-to-end model for jointly improving text summarization and sentiment classification. *arXiv preprint arXiv:1805.01089*.

[12] Tohalino, J. V., & Amancio, D. R. (2018). Extractive multi-document summarization using multilayer networks. *Physica A: Statistical Mechanics and its Applications*, *503*, 526-539.

Publication Details

- CPF-IEEE 2ND ASIANCON 2022-SCOPUS INDEXED CONFERENCE
- Hosted by - PCCOER, Pune, India,
- Conference URL: http://asiancon.org
- Date of Conference: 26th Aug – 28th Aug. 2022

Paper 1200 summary Inbox ×

Microsoft CMT <email@msr-cmt.org>
to me

May 30, 2022, 3:40 PM (1 day ago)

Hello.

Here is submission summary.

Track Name: ASIANCON2022

Paper ID: 1200

Paper Title: Sentiment Analysis and Summarization of Product Reviews using Deep Learning Techniques

Abstract:
Online Customer reviews is a crucial part of the decision making process of buying products from ecommerce. Each product has hundreds of reviews, making it difficult for a consumer to make a decision due to mixed feedback. It takes a long time to read all reviews, and occasionally you end up making decisions based on just a couple of reviews. As a solution, we propose using an automatic review summarizing to assess product reviews and convert them into a more understandable, succinct, and precise style, allowing consumers to save time and effort by not having to browse through all of the reviews. In order to achieve the above goal, we have implemented a LSTM model for the sentiment analysis, followed by clustering and weak-reference extraction for the processing of the summarization, and finally for summarization we have used BERT model for extractive summarization
Keywords– Review Summarization, Multi-document Summarization, Extractive Summarization, LSTM, BERT, Clustering, Sentiment Analysis

Created on: Mon, 30 May 2022 10:09:19 GMT

Last Modified: Mon, 30 May 2022 10:09:19 GMT

Authors:
- bagladivyang03@gmail.com (Primary)
- swaroopnayak1@gmail.com
- kartikbhutada21@gmail.com
- ameybhide2406@gmail.com
- rashmi.rane@mitwpu.edu.in

Secondary Subject Areas:

Submission Files:
Sentiment Analysis and Summarization of Product Reviews using Deep Learning Techniques.pdf (218 Kb, Mon, 30 May 2022 10:01:17 GMT)
Sentiment Analysis and Summarization of Product Reviews using Deep Learning Techniques.docx (92 Kb, Mon, 30 May 2022 10:01:26 GMT)

Submission Questions Response:

Thanks,

Appendices

A. Base Paper(s)

1. Product Sentiment Analysis of Amazon reviews
2. Massive Multi-Document Summarization of Product Reviews with Weak Supervision

B. Plagiarism Report from any open source/proprietary source