A PERSONALISED MALAYALAM TRAVEL RECOMMENDATION MODEL USING DEEP CLUSTERING TECHNIQUES

A Thesis Submitted to the University of Calicut
in partial fulfilment of the requirements for the award of the degree of

DOCTOR OF PHILOSOPHY IN COMPUTER SCIENCE
Under the Faculty of Science

By

MUNEER V.K

Under the guidance of

Dr. MOHAMED BASHEER K.P.
Associate Professor of Computer Science
Sullamussalam Science College, Areekode

P.G & RESEARCH DEPARTMENT OF COMPUTER SCIENCE
Sullamussalam Science College, Areekode - 673639
(Affiliated to the University of Calicut)
Malappuram Dist., Kerala, India

May 2024

DECLARATION

I, Muneer V.K, hereby declare that this thesis entitled "**A Personalised Malayalam Travel Recommendation Model using Deep Clustering Techniques"** is based on the original work done by me under the supervision of Dr. Mohamed Basheer K.P., Assistant Professor, PG & Research Department of Computer Science, Sullamussalam Science College, Areekode, Kerala.

I confirm that,

- The work presented in this Thesis has not been submitted previously for the award of any degree either to this University or to any other University or Institution.
- I have followed the guiding principles given by the University in organizing the Thesis.
- Whenever I have used materials (theoretical analysis, data, figures, and text) from other sources, I have given due credit to them by citing them in the Thesis and giving their particulars in the references.

Muneer V.K.

Areekode
27 May 2024

SULLAMUSSALAM
SCIENCE COLLEGE
AREACODE

Phone: 0483-2850700
Website: www.sscollege.ac.in
Email: mail@sscollege.ac.in

Aided College | Affiliated to the University of Calicut | Re-accredited by NAAC at A Grade

Ref: Date:

CERTIFICATE

This is to certify that the thesis entitled "**A Personalised Malayalam Travel Recommendation Model using Deep Clustering Techniques**", submitted by **Mr. Muneer V.K**, to the University of Calicut, for the partial fulfilment of the requirements for the award of the degree of Doctor of Philosophy (Ph.D.) in Computer Science, is a bonafide research work done by Mr. Muneer V.K under my supervision and guidance in the PG & Research Department of Computer Science, Sullamussalam Science College, Areekode, Malappuram, Kerala. The content embodied in this thesis, in full or in parts, have not been submitted to any other University or Institute for the award of any degree. The thesis is revised as per the modifications and recommendations reported by the adjudicators. Soft copy attached is the same as that of the revised copy. The thesis is submitted as such to the University of Calicut with reference to the letter number 275836/RESEARCH-C-ASST-1/2023/Admn Dated 14.05.2024.

SULLAMUSSALAM SCIENCE COLLEGE ★ AREEKODE ★

Dr. Mohamed Basheer K.P
Associate Professor
PG & Research Department of Computer Science
Sullamussalam Science College, Areekode, Kerala, India

Areekode
27-05-2024

UNIVERSITY OF CALICUT
CERTIFICATE ON PLAGIARISM CHECK

1.	Name of the research scholar	MUNEER V.K		
2.	Title of thesis/dissertation	A PERSONALISED MALAYALAM TRAVEL RECOMMENDATION MODEL USING DEEP CLUSTERING TECHNIQUES		
3.	Name of the supervisor	Dr. MOHAMED BASHEER K.P.		
4.	Department/Institution	P.G & RESEARCH DEPARTMENT OF COMPUTER SCIENCE , Sullamussalam Science College, Areekode, 673639, Malappuram Dist., Kerala		
5.		**Introduction/ Review of literature**	**Materials and Methods**	**Result/ Discussion/Summary/ Conclusion**
	Similar content (%)identified	**2%**	**4%**	**0%**
	Acceptable maximum limit (%)	10	10	10
6.	Software used	Ithenticate		
7.	Date of verification	07/12/2023		

**Report on plagiarism check, specifying included/excluded items with % of similarity to be attached.*

Checked by (with name, designation & Signature)

Name and signature of the Researcher : Muneer. V.K

Name & Signature of the Supervisor Dr. Muhamed Basheer. K.P

The Doctoral Committee* has verified the report on plagiarism check with the contents of the thesis, as summarized above and appropriate measures have been taken to ensure originality of the Research accomplished herein.

Name & Signature of the HoD/HoI (Chairperson of the Doctoral Committee)

** In case of languages like Malayalam, Tamil, etc. on which no software is available for plagiarism check, a manual check shall be made by the Doctoral Committee, for which an additional certificate has to be attached*

Acknowledgments

This thesis is the outcome of the journey I have travelled for around four years. This has been kept on track and seen to completion due the support and encouragement of many people. It's a happy thing to thank those unforgettable people who made this thesis possible.

First of all, I would like to extend thanks to my research supervisor Dr. Mohamed Basheer K.P, Assistant Professor, PG & Research Dept. of Computer Science, Sullamussalam Science College, for his active, concise and encouraging supervision. He was the strongest pillar for me during the entire period of the research. His positive attitude, friendliness, and confidence in my research inspired me and gave me confidence. His careful reviews contributed immensely to the production of the research papers and this thesis. The understanding, patience, kindness, freedom, and support I could enjoy, made my research tenure a memorable one.

I am deeply indebted to my college Principal Dr. P. Muhamed Ilyas, Principal, Sullamussalam Science College, giving insight, motivation, moral support, motivation, and direction to my research studies and helping me in many ways in designing the work pattern. I extend my gratitude to the college Management, Prof. N.V Abdul Rahman, Manager of the College, my colleagues and students of Sullamussalam Science College for cooperation in several ways during the tenure of this research.

I extend my sincere thanks to the esteemed members of the research advisory committee for their invaluable contributions to my Ph.D. journey. Dr. Lajish VL, Associate Professor & Head, Dept. of Computer Science, University of Calicut, Dr. Vasudevan, Dept. of Library Science, University of Calicut, Binu P Chacko, Principal, Dr. Shameem Kappan, Head, PG & Research Department of Computer Science, SS College, for their unwavering support, motivation, and expert guidance which have been instrumental in shaping the direction of my research. Their constructive criticism and critical insights have been a constant source of refinement, ensuring that my work reached the highest standards of academic excellence. I am deeply grateful for their commitment to my academic growth, and I acknowledge their vital role in the successful completion of my work.

During my Ph.D. research, I am deeply grateful for the collaborative efforts and unwavering support I received from my colleague Mrs. Rizwana Kallooravi Thandil with insightful discussions, and continuous encouragement played a pivotal role in shaping the trajectory of research. I am sincerely thankful for her commitment to excellence and the positive impact she has had on this academic journey.

I remember with gratitude the team members and mentors at ICFOSS Trivandrum, especially Dr. R.R. Rajeev and Dr. Achuthsankar of Kerala University, Dr. Premjith, Dr. Prem Sankar and Mr. Amal A Salam for the insightful discussions which helped a lot in my work.

I thank my colleagues Dept. Computer Science, SS College for the extremely enjoyable and productive cooperation while researching and shaping my work. I am also thankful to all the staff members, both teaching and non-teaching, research scholars and the students at the Sullamussalam Science College for helping with required facilities whenever needed.

The support I get from my mother, siblings, wife and sister in laws cannot be expressed in words as they are always there for everything. I also remember with love and gratitude the help, support, and encouragement I could enjoy from my siblings, family members and friends. Special thanks to my kids Rua and Dua for their unconditional love and cooperation, without which I would not have come this far.

And I bow my head before that superpower almighty for giving me the strength, knowledge, ability, and opportunity to undertake this research study and to preserve and complete it satisfactorily.

Muneer V.K

Dedicated

To My Father for His Blessings in Abundance and
My Mother for Her Unconditional Love and Support!!

ABSTRACT

The research aims to develop a Personalized Travel Recommender System tailored for the Malayalam-speaking audience in Kerala, India. Due to the unavailability of a benchmark dataset in Malayalam, a two-pronged data collection strategy was employed: extraction of 13458 travelogues and reviews from Facebook's largest travel group in Kerala, 'Sanchari', and independent travel blogs and collecting 2,006 records via a Google form. As part of this work, firstly, it seeks to develop an automated framework for processing Malayalam text scraped from social media, addressing the language's rich morphological complexity. Second, it intends to create an intelligent system that utilizes opinion mining techniques to continuously learn user preferences, thereby enabling highly personalized travel suggestions. Finally, the work aims to design a recommender model that leverages machine learning algorithms to identify tourist destinations tailored to the preferences of travelers who exhibit similar tastes, employing both collaborative and content-based filtering techniques. Data collection process was the biggest hurdle in the initial phase. Collecting Malayalam lengthy travelogue was the aim to create a dataset. The focus reached the Facebook group named sanchari, which is the largest travel group in Malayalam Language. Data collection faced several challenges such as memory leak, bot detection, performance optimisation and page rendering time. By utilizing some special features exclusively available for group admins utilized to solve these issues. All travelogues written in English, Manglish or any language other than Malayalam removed from the spreadsheet before preprocessing.

To transform this unstructured data into a usable dataset, a variety of Natural Language Processing techniques, along with a Part of Travelogue Tagger (POT Tagger) and Look-up Dictionary, were used for preprocessing. Feature engineering included vectorization and one-hot encoding of important variables to create 'Travel DNA' and 'Location DNA.' Travel DNA is composed by aggregating key travel attributes such as travel type, travel mode, and user preferences into a numerical vector through techniques like one-hot encoding and vectorization, thereby providing a compact representation of travel patterns of users. Location DNA is composed by gathering essential characteristics of various travel destinations, such as location type, climate, and popularity, and converting them into a numerical vector using methods like one-hot encoding and vectorization.

Four distinct recommender models were developed leveraging various algorithms in Artificial Intelligence: Rule-based Cosine Similarity, Collaborative Filtering based on K-Means Clustering, Content-based Filtering through Hierarchical Agglomerative Clustering, and a model utilizing Bidirectional Long Short-Term Memory (BiLSTM) networks. Additionally, a comparative model was designed that combined autoencoders with five different machine learning algorithms. These models underwent rigorous individual testing to evaluate their performance.

The rule-based cosine similarity recommender model utilizes the angle between user and item vectors in a multi-dimensional space to measure similarity, thereby providing personalized travel suggestions based on pre-defined rules and user preferences. The clustering techniques employed in this research include K-Means for collaborative filtering to group similar users, and Hierarchical Agglomerative Clustering for content-based filtering to categorize travel destinations. The BiLSTM recommender model leverages neural networks to capture both past and future context in the data. The autoencoder-based travel recommender model employs neural network architectures to compress and reconstruct the user-item interaction data, effectively capturing latent features that are used for generating more accurate and personalized travel suggestions. The RS designed with various techniques to identify the best suggestions to the users. From these models, collaborative filtering using K-Means Clustering and the model designed with Autoencoder exhibit promising results.

Keywords: Recommendations System, Natural Language Processing, Language Computing, Clustering Techniques, Autoencoder.

സംഗ്രഹം

കേരളത്തിനകത്തും പുറത്തുമുള്ള മലയാളികളായ യാത്രക്കാർക്കും പ്രേക്ഷകർക്കും അനുയോജ്യമായ ഒരു വ്യക്തിഗത യാത്രാ ശുപാർശ സംവിധാനം വികസിപ്പിക്കുകയാണ് ഈ ഗവേഷണം ലക്ഷ്യമിടുന്നത്. മലയാളത്തിൽ ഒരു ബെഞ്ച്മാർക്ക് ഡാറ്റാസെറ്റ് ലഭ്യമല്ലാത്തതിനാൽ, ദ്വിമുഖ ഡാറ്റാ ശേഖരണ രീതിയാണ് അവലംബിച്ചത്. കേരളത്തിലെ *Facebook*-ന്റെ ഏറ്റവും വലിയ ട്രാവൽ ഗ്രൂപ്പായ 'സഞ്ചാരി'യിൽ നിന്നും സ്വതന്ത്ര ട്രാവൽ ബ്ലോഗുകളിൽ നിന്നും *13458* യാത്രാവിവരണങ്ങളും നിരൂപണങ്ങളും അതിനു പുറമേ ഒരു ഗൂഗിൾ ഫോം വഴി *2,006* റെക്കോർഡുകളും ശേഖരിച്ചു. ഈ ഗവേഷണത്തിന്റെ ഭാഗമായി, ഭാഷയുടെ സമ്പന്നമായ രൂപഘടന സങ്കീർണ്ണതയെ അഭിസംബോധന ചെയ്ത് സോഷ്യൽ മീഡിയയിൽ നിന്ന് ശേഖരിച്ച മലയാളം ടെക്സ്റ്റ് പ്രോസസ്സ് ചെയ്യുന്നതിനുള്ള ഒരു ഓട്ടോമേറ്റഡ് സിസ്റ്റം വികസിപ്പിക്കാൻ ഇത് ശ്രമിക്കുന്നു. രണ്ടാമതായി, ഉപയോക്തൃ മുൻഗണനകൾ തുടർച്ചയായി പഠിക്കുന്നതിനായി അഭിപ്രായ ഖനന സാങ്കേതിക വിദ്യകൾ പ്രയോജനപ്പെടുത്തുന്ന ഒരു ഇന്റലിജന്റ് സിസ്റ്റം സൃഷ്ടിക്കുകയും അതുവഴി ഉയർന്ന വ്യക്തിഗതമാക്കിയ യാത്രാ നിർദ്ദേശങ്ങൾ പ്രാപ്തമാക്കുകയും ചെയ്യുന്നു. അവസാനമായി, കൊളാബറേറ്റീവ് രീതിയും യാത്രയുടെ ആശയം അടിസ്ഥാനമാക്കിയുള്ളതുമായ ഫിൽട്ടറിംഗ് ടെക്നിക്കുകളും ഉപയോഗിച്ച് സമാന അഭിരുചികൾ പ്രകടിപ്പിക്കുന്ന സഞ്ചാരികളുടെ മുൻഗണനകൾക്കനുസൃതമായി ടൂറിസ്റ്റ് ഡെസ്റ്റിനേഷനുകൾ തിരിച്ചറിയുന്നതിന് മെഷീൻ ലേണിംഗ് അൽഗോരിതങ്ങൾ പ്രയോജനപ്പെടുത്തുന്ന ഒരു ശുപാർശ മോഡൽ രൂപകൽപ്പന ചെയ്യുക എന്നതാണ് ഈ ഗവേഷണത്തിന്റെ പ്രധാന ലക്ഷ്യം.

ഘടനാരഹിതമായ യാത്രാവിവരണങ്ങളും നിരൂപണങ്ങളും ഉപയോഗയോഗ്യമായ ഒരു ഡാറ്റാസെറ്റാക്കി മാറ്റുന്നതിന്, ട്രാവലലോഗ് ടാഗർ (*POT* ടാഗർ), ലുക്ക്-അപ്പ് നിഘണ്ടു എന്നിവ നിർമ്മിച്ചു. വൈവിധ്യമാർന്ന നാച്ചുറൽ ലാംഗ്വേജ് പ്രോസസിംഗ് ടെക്നിക്കുകൾ പ്രീപ്രോസസിംഗിനായി ഉപയോഗിച്ചു. ഫീച്ചർ എഞ്ചിനീയറിംഗിൽ 'ട്രാവൽ ഡിഎൻഎ', 'ലൊക്കേഷൻ ഡിഎൻഎ' എന്നിവ സൃഷ്ടിക്കാൻ വെക്‌ടറൈസേഷനും വൺ-ഹോട്ട് എൻകോഡിംഗും ഉപയോഗിച്ചു. യാത്രയുടെ രീതി, സ്ഥലം, കാലാവസ്ഥ, സഹയാത്രികർ, അവരുടെ ഉപയോക്തൃ മുൻഗണനകൾ എന്നിങ്ങനെയുള്ള പ്രധാന യാത്രാ ആട്രിബ്യൂട്ടുകൾ വൺ-ഹോട്ട് എൻകോഡിംഗ്, വെക്‌ടറൈസേഷൻ തുടങ്ങിയ സാങ്കേതിക വിദ്യകളിലൂടെ ഒരു സംഖ്യാ വെക്‌ടറിലേക്ക് സംയോജിപ്പിച്ചാണ് ട്രാവൽ ഡിഎൻഎ രചിച്ചിരിക്കുന്നത്.

ആർട്ടിഫിഷ്യൽ ഇന്റലിജൻസിലെ വിവിധ അൽഗോരിതങ്ങൾ പ്രയോജനപ്പെടുത്തി നാല് വ്യത്യസ്ത റെക്കമെന്റേഷൻ മോഡലുകൽ വികസിപ്പിച്ചെടുത്തു. റൂൾ അധിഷ്ഠിത കോസൈൻ സാമ്യത, *K Means* ക്ലസ്റ്ററിംഗിനെ അടിസ്ഥാനമാക്കിയുള്ള ഫിൽട്ടറിംഗ്,

ഹൈറാർക്കിക്കൽ അഗ്ലോമറേറ്റീവ് ക്ലസ്റ്ററിംഗ്, *BiLSTM* മോഡൽ, ഓട്ടോ എൻകോഡറുകർ അഞ്ച് വ്യത്യസ്ത മെഷീൻ ലേണിംഗ് അൽഗോരിതങ്ങളുമായി സംയോജിപ്പിച്ച് ഒരു താരതമ്യ മോഡലും ഇതിന്റെ ഭാഗമായി രൂപകൽപ്പന ചെയ്തിട്ടുണ്ട്.

ഈ ഗവേഷണത്തിൽ ഉപയോഗിച്ചിരിക്കുന്ന ക്ലസ്റ്ററിംഗ് ടെക്നിക്കുകളിൽ സങ്കീർണ്ണമായ പാറ്റേണുകളും യാത്രികരുടെ പെരുമാറ്റത്തിലെ ക്രമങ്ങളും അടിസ്ഥാനമാക്കി സൂക്ഷ്മവും ഉയർന്ന വ്യക്തിഗതമാക്കിയതുമായ യാത്രാ നിർദ്ദേശങ്ങൾ വാഗ്ദാനം ചെയ്യാൻ ന്യൂറൽ നെറ്റ്‌വർക്ക് ആർക്കിടെക്‌ചറുകളും ഉപയോഗിക്കുന്നു.

"You are not a drop in the ocean,
you are the ocean in a drop."

- Rumi

Table of Contents

List of Tables

List of Figures

LIST OF ABBREVIATIONS

AE	Autoencoder
AI	Artificial Intelligence
ANN	Artificial Neural Network
Bi-LSTM	Bidirectional LSTM
CBF	Content Based Filtering
CF	Collaborative Filtering
CNN	Convolutional Neural Network
CSV	Comma Separated Values
CT	Clustering Techniques
DL	Deep Learning
DNN	Deep Neural Network
DT	Decision Tree
HAC	Hierarchical Agglomerative Clustering
KNN	K Nearest Neighbors
L	Location
LC	Location Climate
LSTM	Long Short-Term Memory
LT	Location Type
ML	Machine Learning
MLP	Multilayer Perceptron
NLP	Natural Language Processing
POS	Part of Speech
POT	Part of Travelogue

RNN	Recurrent Neural Network
RS	Recommender Systems
SGD	Stochastic Gradient Descent
TM	Travel Mode
TRS	Travel Recommender Systems
TT	Travel Type

1 Introduction

1.1 Background and Context

In recent years, social media platforms have evolved from mere platforms for social interactions to rich repositories of data that hold significant potential for various research domains. Facebook offers a wealth of user-generated content that can be invaluable for extracting insights into consumer behaviour, trends, and preferences. This shift has made social media platforms an increasingly vital source of research data across multiple disciplines, including travel and tourism.

Facebook, with its expansive database of travelogues and user reviews, offers a goldmine of unstructured data. By facilities provided by facebook.com, travelogues as unstructured Malayalam text with its associated details are retrieved, extracted travel reviews from various other online sources to develop an unstructured dataset. Moreover, as this research focuses on Malayalam language content, the lack of benchmark datasets makes the task particularly challenging. The unstructured nature of travelogues necessitates advanced Natural Language Processing (NLP) techniques to convert this raw data into a structured, usable format.

Within the Facebook ecosystem, the 'Sanchari' group stands as the largest travel community in Kerala, where travel enthusiasts share their experiences through travelogues written in Malayalam. Malayalam, a Dravidian language with intricate linguistic structures, offers a unique challenge and opportunity for researchers as it is a richly inflectional and morphologically complex language. This research aims to delve into this unexplored area by utilizing travelogues in Malayalam to build a personalized travel recommender system. These systems become especially relevant in multicultural contexts where local language-based recommendations can add another layer of personalization.

Natural Language Processing techniques play a vital role in this research for preprocessing Malayalam travelogues to derive meaningful insights. From sentence and word tokenization to stemming and lemmatization, multiple stages of preprocessing prepare the data for the machine learning algorithms that follow. The research explores various machine learning models, including Rule-Based Models, Collaborative Filtering, K-means clustering, and advanced techniques like Bidirectional Long Short-Term Memory (Bi-LSTM) and Autoencoders, to create a robust and personalized travel recommendation system.

1.2 Scope and Motivation

The motivation behind this research stems from a blend of recognized gaps in existing systems, technological advancements, and real-world challenges and needs. Each of these aspects feeds into the overarching goal of this thesis: to design a personalized travel recommendation system that is not only technologically advanced but also deeply rooted in regional and linguistic contexts.

1.2.1 Addressing the Gap in Personalized Travel Recommendations

The field of travel and tourism has seen a significant uptick in data-centric research over the past decade. Despite this, there exists a noticeable gap when it comes to personalized travel recommendation systems that cater to regional and linguistic preferences. One such unexplored avenue is the Malayalam-speaking community in Kerala, whose unique travel habits and preferences have been largely ignored by mainstream travel recommender systems. By focusing on this demographic, this research aims to contribute a meaningful layer of regional personalization to the existing body of work in this domain.

1.2.2 The Need for Local Language Support

Languages carry nuanced cultural, geographic, and personal contexts that have direct implications on travel patterns and preferences. While English-based travel recommendation systems are abundant, localized language support offers an untapped potential to deeply personalize travel experiences. The Malayalam language is highly inflectional and morphologically rich, requiring specialized NLP techniques to understand and utilize the underlying meanings in travelogues effectively. This research aims to fill this gap by developing a personalized travel recommender system in Malayalam, providing more contextually relevant travel suggestions.

1.2.3 Utilizing social media as a Rich Data Source

Social media platforms like Facebook offer an almost inexhaustible resource of travel narratives, reviews, and shared experiences. Active engagement in travel-related groups provides not only a large dataset but also an incredibly diverse one. However, the potential of these platforms remains largely untapped, mainly due to the complexities surrounding data extraction, privacy concerns, and data cleaning. The motivation behind this research is also to utilize these vast resources effectively, demonstrating how social media data can enrich travel recommendation algorithms.

1.2.4 Advancements in Natural Language Processing and Machine Learning

Recent breakthroughs in machine learning algorithms and Natural Language Processing techniques offer newfound possibilities for complex tasks like travel recommendations. Leveraging advanced methodologies such as Bi-LSTM and Autoencoders allows for better pattern recognition, understanding of sequence dependencies, and consequently, more accurate recommendations. The time is ripe for integrating these advanced technologies into developing a more refined and sophisticated travel recommender system, and this research aims to be at the forefront of this integration.

1.2.5 Addressing Real-World Challenges

In today's saturated travel market, consumers are often overwhelmed by the multitude of options available. A system that can sift through this data to provide personalized, reliable, and contextually relevant recommendations is not just an academic exercise but also a real-world necessity. By tackling challenges ranging from data extraction and preprocessing to effective recommendation algorithm design, this research aims to develop a system that can have practical applications, enhancing the overall travel experience for the Malayalam-speaking population.

1.3 Objectives and Contribution

1.3.1 Objective - Creation of a Malayalam Language Travel Dataset

One of the primary objectives is to create a comprehensive dataset of Malayalam travel reviews and travelogues. This data will serve as the backbone of the research, enabling the development of a personalized travel recommendation system. Given the absence of pre-existing benchmark datasets in Malayalam, this part is pivotal for the research.

1.3.2 Natural Language Processing for Malayalam Text

The Malayalam language is rich in morphology and highly inflectional, making it uniquely challenging for computational analysis. Therefore, another objective is to apply Natural Language Processing (NLP) techniques, including but not limited to tokenization, stemming, and lemmatization, to preprocess and understand the intricacies of the language.

1.3.3 Contribution - Feature Extraction and Structured Dataset Creation

The aim here is to transform the unstructured Malayalam travelogues into a structured dataset, annotating key features like Travel Type, Travel Mode, Location, Location Climate, and Location Type. This structured data will be crucial for any machine learning models to make accurate predictions. Specially designed Part of Travelogue Tagger (POT Tagger), Look-up dictionary, and root-pack extractor used for constructing the structured dataset in travel domain.

1.3.4 Development of Machine Learning Models for Recommendations

The research executed different machine learning techniques for recommendation, like Rule-Based models, Collaborative Filtering, and clustering techniques such as K-Means and Hierarchical Agglomerative Clustering. Among these methods, K-Means clustering based Collaborative filtering and Autoencoder model outperforms other techniques.

1.3.5 Implementation of Deep Learning Techniques

Finally, given the complexity and the nature of the data, this research aims to leverage advanced Deep Learning architectures like Bi-LSTM and Autoencoders for generating more nuanced and personalized travel recommendations.

1.4 Conclusion

The introduction chapter of this thesis provides an overarching framework for the development of a personalized travel recommender system based on Malayalam travel reviews. It begins by discussing the importance of social media as a valuable repository for travel data, laying particular emphasis on Malayalam travelogues sourced from Facebook. The chapter sets the context by elaborating on the scope of research in the travel and tourism domain, challenges in data extraction, and the importance of data preprocessing using Natural Language Processing (NLP) techniques. Various models for travel recommendation, ranging from rule-based to machine-learning approaches like Bi-LSTM and autoencoders, are also introduced.

The thesis is carefully structured into eight distinct chapters, with each one tackling a unique aspect of the recommender system. The chapters span from a review of relevant literature to details on data collection, preprocessing, and feature engineering. They also dive deep into specific recommendation models, including rule-based, clustering-based, and deep-learning approaches. Each chapter not only adds to the understanding of its subject matter but also contributes to the complete narrative of creating a Malayalam-based travel recommender system. This structure ensures a comprehensive presentation of the research conducted, guiding the reader through the multiple layers of complexity involved.

2 Literature Review

2.1 Introduction

In the digital age, social media platforms have emerged as invaluable reservoirs of data, offering a wealth of user-generated content that can be harnessed for academic research. These platforms, such as Facebook, Twitter, and Instagram, provide real-time insights into user behaviour, preferences, and social interactions, making them a treasure for data scientists and researchers alike. Specifically, in the realm of travel recommendation systems, social media offers an unprecedented opportunity to capture authentic travel experiences, reviews, and preferences [1]. User-generated content, including travelogues, check-ins, and reviews, can be extracted and analysed to feed into recommendation algorithms [2]. This is particularly beneficial for languages and cultures that are underrepresented in mainstream datasets, such as Malayalam. By collecting data from social media groups and pages dedicated to travel experiences in this linguistic context, researchers can generate culturally and linguistically nuanced datasets. These datasets not only serve as a robust foundation for machine learning models but also offer the granularity needed for highly personalized recommendations [3]. Therefore, social media[4] stands as an indispensable resource for data extraction and research, capable of driving innovation and enhancing the efficacy of personalized travel recommendation systems.

This literature review aims to thoroughly investigate existing studies and methodologies in the fields of recommender systems [5][6], natural language processing (NLP) [7], and travelogues. The field of recommender systems has seen remarkable growth in the past decade, with applications spanning various industries such as e-commerce, entertainment, and travel. While most research has focused on widely spoken languages like English, there is a noticeable gap in studies targeting less common languages like Malayalam [8]. This becomes particularly important as language plays a pivotal role in the effectiveness of any recommender system. The focus of this research is to bridge this gap by developing a travel recommender system tailored specifically for Malayalam-speaking users [3], [9]. This chapter serves as a foundational element, aiming to provide an exhaustive survey of existing work in fields that intersect with this research.

The advent of personalized destination recommendation systems [10] marks a pivotal shift in the landscape of travel planning [11]. Gone are the days of one-size-fits-all travel suggestions; today's travellers seek experiences that align closely with their individual preferences, be it adventure, relaxation, or cultural exploration. Leveraging advanced algorithms and machine learning techniques, these recommendation systems sift through vast amounts of data, including user-generated content such as travelogues and reviews, to offer tailor-made travel itineraries. In the context of this research, the Malayalam travel recommender system goes a step further by incorporating linguistic and cultural nuances, making the recommendations not just personalized but also contextually relevant [12]. By doing so, the system transcends the limitations of generic travel platforms and offers a truly user-centric approach to travel planning. The scope for personalized travel planning [13] is vast, offering users not just a list of destinations but a curated experience that resonates with their unique tastes and preferences.

The chapter delves into multiple interdisciplinary areas such as recommender systems, natural language processing (NLP), data preprocessing techniques, and the unique challenges posed by using travelogues as a data source. Each of these areas has its complexities and challenges, and the review aims to shed light on how they can be navigated or leveraged to enhance this research. Special emphasis is laid on studies and methodologies that offer insights into Malayalam language processing and recommendation systems [14]. Through this, the review identifies existing gaps in the current body of research, especially those related to less-commonly researched languages and specialized domains like travel.

In addition to providing a broad academic context, the review also serves to highlight the unique contributions of this study. Among these is the development of specialized tools for Malayalam language processing, including a Part-of-Travelogue Tagger (POT Tagger) and a dedicated lookup dictionary. These tools are specifically designed to meet the challenges posed by the complex morphology and syntax of Malayalam [15].

Moreover, the chapter outlines various approaches adopted in recommender systems[16], ranging from traditional rule-based methodologies to advanced machine learning and deep learning techniques. It explores how these approaches have evolved and how they can be adapted to the Malayalam language. The chapter also includes a comparative analysis of these techniques, which is particularly relevant for understanding the pros and cons of each approach.

2.2 Data extraction from social media

A comprehensive literature review and classification of recommender systems (RS) in social media is done in [17]. This paper provides an analysis of 61 articles published between 2011 and 2015, shedding light on recommendation approaches, research domains, data sets, data mining techniques, recommendation types, and performance measures, thereby providing valuable insights for future research in the social media RS domain.

Research work in [18] addressed the need for personalized landmark recommendations in trip planning, emphasizing the importance of considering the traveller's characteristics and trip-specific factors. By leveraging geo-tagged social media data, the proposed approach analyses the spatial and temporal aspects of trips, calculates landmark significance, and generates clusters of personalized recommendations [19][20]. Comparative evaluations and user studies demonstrate the superior performance of this approach in terms of accuracy and relevance, particularly for lesser-known places and events, enhancing the overall travel experience.

The authors in paper [11] investigated the utilization of Twitter data for personalizing travel recommendations, employing a machine learning classification model to identify travel-related tweets and subsequently offering recommendations on places of interest based on user preferences [21]. The evaluation of the model, which incorporates social data from the user's friends and followers, shows a 68% prediction accuracy, with potential for improvement through better training datasets and more refined travel category identification, suggesting a promising avenue for enhancing personalized travel recommendations [22].

Paper [23] explored the use of Facebook-targeted advertisements as a method for collecting survey data, demonstrating its potential in building large employee-employer-linked datasets. The study addresses concerns about sample selectivity and highlights the advantages of this approach, including rapid data collection, flexible sample targeting, and cost-effectiveness, while also acknowledging its remaining limitations, making it a valuable contribution to the field of data collection and analysis.

The work in [24] concluded that text mining has emerged as a prominent field, employing Natural Language Processing techniques to extract meaningful information patterns from unstructured textual data. This survey focused on the application of text analytics and mining methods in social media platforms like Facebook and Twitter, offering insights into how these techniques are utilized to identify key themes and providing a valuable foundation for future research in this area.

2.2.1 Challenges

Research work in paper [25] discussed about web scraping method aimed at improving the presentation of information sourced from social media platforms like Facebook [26][27] and Twitter, addressing the challenges of information redundancy and user relevance in timelines or feeds. The study leverages APIs from Facebook and Twitter Developers, along with regular expressions, to structure and reduce information overload.

The investigation of [28] delves into the design and implementation of a real-time data processing ecosystem at Facebook, emphasizing key design decisions that impact usability, performance, fault tolerance, scalability, and correctness. Notably, their focus on achieving seconds of latency, rather than milliseconds, and the use of a persistent message bus for data transport have paved the way for effective real-time stream processing systems, setting a valuable precedent for handling substantial data volumes, and enhancing real-time analytics across various use cases.

Research conducted in paper [25] highlighted the challenges of information overload and redundancy in social media platforms like Facebook, Twitter, and Instagram, and introduced a Web Scraping method to address these issues by searching, combining, and presenting information based on user preferences.

Facebook is considered as one among the largest repository for posting written reviews and multimedia contents. Other social media platforms like Twitter, YouTube and Instagram contains only micro texts and reviews. This is the reason for considering Facebook groups and pages for data extraction. Sanchari is the largest Facebook group in travel domain which contain large number of Malayalam lengthy travel reviews. Extraction of these travelogues faced several challenges like page rendering time, memory leaks, bot detection etc. admin insight is special privilege available for group admins, through which the admins can retrieve required information from the groups and pages. Sharing a google from to the public travel network for collecting travel preferences adopted here, which was a time-consuming process.

2.2.2 NLP Techniques for Extraction of Structured Data

Natural Language Processing (NLP) has become a cornerstone in the fields of linguistics, computer science, and artificial intelligence. Its applications span across various domains, including but not limited to, machine translation, sentiment analysis, and information retrieval. However, most of these applications are geared towards widely spoken languages like English.

The work in [29] focused on the efficient extraction of entities from informal and noisy Malayalam social media text, utilizing pre-processing, feature extraction, and a Support Vector Machine classifier. The incorporation of unsupervised features derived from the Structured Skip-gram model led to improved accuracy in entity extraction, surpassing the performance of existing systems evaluated in the FIRE2015 task. [30] examined the application of Facebook data analysis to suggest career paths based on the content shared on public Facebook profiles. It notably introduced a unique approach to data preprocessing, involving spell correction, emoticon analysis, and multilingual translation, with a focus on enhancing sentiment analysis, offering a fresh perspective on utilizing social media data for career guidance. [8] investigated many machine learning and deep learning methods that have been employed to improve the performance of NLP tasks including agnostic sentence representations.

Facebook groups offer a rich data source for collecting Malayalam travelogues. Malayalam poses unique challenges due to its complex morphological structure and script. [32] presented a novel methodology for deep-level tagging of Malayalam text, leveraging the language's morphological richness and agglutinative nature to advance computational analysis in this linguistic context

[33]. The paper [15] conducted a quantitative analysis of the morphological complexity of the Malayalam language, which is known for its intricate inflections, derivations, and compounding. Research work in [34] addressed the fundamental task of morphological analysis, particularly in the context of agglutinative languages like Malayalam, which involve complex morphological changes at morpheme boundaries due to sandhi. The study introduced a deep learning approach employing Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), and Gated Recurrent Units (GRU) systems, achieving high accuracies in the automatic identification and segmentation of morphemes from original words, thereby enhancing the grammatical analysis of Malayalam. To address this, the Root-Pack Python library specializes in finding root words [35] in Malayalam and has proven to be an effective tool for lemmatization.

The unstructured and noisy text extracted from social media must be converted into a structured format which can be fed into machine learning models. There are several phases included in this conversion. Apart from conventional preprocessing steps, some special purpose tools are created to address unique problems. Specially designed part of travelogue tagger (POT Tagger) for annotating tokens, root-pack extractor for lemmatization, Malayalam look-up dictionary for identifying category of features. With the help of these tools the lengthy passages compressed into a set of discrete features and stored in a spreadsheet.

2.2.3 Feature Extraction and Structured Dataset Creation

Travelogues offer a rich source of user-generated content, often capturing intricate details and user experiences that are invaluable for building a recommender system. [1] focused on the application of multi-criteria Collaborative Filtering (CF) in hotel recommendations within e-tourism platforms, emphasizing the utilization of customer online reviews as a data source for improving recommendation precision on TripAdvisor. The study's analysis of the dataset affirmed that incorporating online reviews into the recommendation process resulted in accurate hotel recommendations, highlighting the value of machine learning techniques in enhancing the user experience in the e-tourism domain. The authors in paper [9] aimed to develop a recommender model for the Malayalam language in the travel and tourism domain[36], emphasizing the challenges posed by Malayalam's low-resource and highly inflected nature.

2.3 Recommender Systems

A recommendation system is a specialized software that provides personalized suggestions or recommendations to users based on various criteria [37][38]. These systems are commonly used in diverse applications ranging from e-commerce websites suggesting products to buy, streaming services recommending movies or music, to travel sites proposing destinations or activities [39][40].

Zheng in paper [41] introduced a personalized friend and location recommender system[42] for geographical information systems (GIS) on the web, utilizing individuals' location histories as implicit ratings to recommend friends and locations. The system employed a hierarchical-graph-based similarity measurement (HGSM) and content-based methods, outperforming related similarity measures and enhancing user experiences with more appealing location recommendations based on real-world GPS data [43], [44]. Hossain in paper [45] conducted a comparative analysis of Collaborative Filtering (CF), Content-Based Filtering (CB), and Sentiment Analysis for building a recommendation engine using a Spotify dataset [46] [47]. More advanced systems use machine learning algorithms, deep learning [48], and natural language processing to provide highly accurate and context-aware recommendations. Data sources for these systems can range from user profiles and transaction histories to user-generated content like reviews and social media posts.

The research work in [49] introduced a collaborative filtering recommendation framework that leveraged social networks to improve the accuracy and relevance of recommendations, particularly in the context of movie ratings and user social connections [50]. Authors of paper [51] focused on the analysis of textual similarity among users in a social network by extracting and processing data from social networking sites. These systems combine both Collaborative [52] [53] and Content-Based filtering to make more accurate recommendations. Hybrid systems can be implemented in various ways: by making predictions separately with each approach and combining them, by adding collaborative and content-based features into a single model, or by unifying the models into a single model. [54] offered an overview of recommender systems, covering collaborative filtering [55], content-based filtering, and the hybrid approach, highlighting their evolution in parallel with the internet [56].

2.4 Machine Learning Models for Recommendation

Personalization is a key feature of modern recommender systems and significantly improves user engagement by providing more relevant and accurate recommendations [57][58][59]. However, achieving a high degree of personalization is a challenging task. Issues like data sparsity, where only a limited amount of data is available for individual users, and scalability, especially in systems with a large user base, pose significant challenges [60][61].

2.4.1 Rule-Based Methodologies

Rule-based recommender systems operate by applying a predefined set of rules to generate recommendations. The methodology incorporates linguistic and cultural nuances, making it particularly effective for this domain [62], [63]. A rule-based recommender model is a type of recommendation system [64] that uses a predefined set of rules or heuristics to generate recommendations for users [65], [66]. Unlike data-driven models like collaborative filtering [67] or content-based filtering [45], which rely on historical data and machine learning algorithms [68][69], rule-based models use logic and explicit rules to make recommendations.

In a rule-based recommender system, rules are created based on domain knowledge, expert judgment, or other structured information. These rules are then used to evaluate the available items and filter or rank them according to how well they meet the criteria outlined in the rules [70].

2.4.2 Clustering Methods

Clustering methods like K-Means and Hierarchical Agglomerative Clustering group similar items or users together. This research introduces a collaborative K-Means clustering model and a content-based Hierarchical Agglomerative Clustering model to explore personalization in Malayalam travel recommendations [71]. K-Means clustering in this research aims to create distinct clusters based on the features TT, TM, LC, LT, and user preferences. For instance, each destination is represented as a vector of these features: TT (bike, train, road), TM (friends, solo, family), LC (summer, winter), and LT (nature, adventure). Users are similarly represented based on their preferences for these features. The K-Means algorithm then assigns each destination and user to a cluster where the mean of these features is closest to their own. By doing so, the system can provide personalized recommendations [72], [73].

2.4.3 Deep Learning Models for Recommendation

Bidirectional Long Short-Term Memory (Bi-LSTM) is an advanced type of recurrent neural network (RNN) that captures the sequential dependencies in data in both forward and backward directions [74]. This research aims to revolutionize travel planning [75] for Malayalam-speaking users through a personalized recommendation [76] system that leverages the power of Bidirectional Long Short-Term Memory (Bi-LSTM) algorithms [77]. The authors in paper [78] conducted an evaluation of ten different recurrent neural networks (RNN) structures for generating recommendations using written reviews in the context of e-commerce [79][80]. Their study included well-known RNN implementations like Multistacked bi-directional Gated Recurrent Unit (GRU) and Long Short-Term Memory [81], [82]. Bi-LSTM, an advanced type of recurrent neural network [83], is employed to capture intricate sequential dependencies in travel data, thereby providing a nuanced understanding of individual user preferences. [84] introduced a novel architecture named AC-Bi-LSTM, which combined bidirectional Long Short-Term Memory (Bi-LSTM), an attention mechanism, and a convolutional layer [84] [85] for text classification, addressing the challenges posed by high-dimensional and sparse text data.

In the evolving landscape of travel recommendation systems [86][87], the incorporation of machine learning algorithms has marked a transformative shift towards highly personalized and contextually relevant suggestions. One such impactful algorithm is the Autoencoder [88] [89][90], which has been specially adapted for Malayalam language processing in recent research endeavors. The paper [91] introduced a novel model for the rating prediction task in recommender systems, based on a deep autoencoder [91], [92] with six layers, demonstrating superior performance compared to previous state-of-the-art models on a Netflix dataset. The research emphasized the importance of deep autoencoder models [93], [94], non-linear activation functions, and regularization techniques in improving generalization and preventing overfitting, alongside a new training algorithm to address the natural sparseness of collaborative filtering [95][96].

2.5 Comparison of Work with Existing Methods

Lack of benchmarking dataset in Malayalam travel domain was the biggest hurdle faced in this research. A new dataset is curated from unstructured travelogues which addressed the problem domain. There are multiple works done similar to this research mainly in English and other established languages. All these works completed with the help of benchmark dataset in that particular language. In this case that was the challenge. Many Indian low resourced languages like Malayalam also facing this issue in various domains. Language computing is still in infancy stage especially for Dravidian languages. Comparative analysis of this research work with existing recommendation models in similar domains are given below. Most of these referred works are done in English language and they enjoyed the availability of benchmark datasets.

In this research, the work done by using autoencoder model could produce an accuracy of 95.84%. The work in paper [152] discuss a movie RS using autoencoder by using Netflix dataset has accuracy of 86.86% and another work in [153] about product RS with AE based on CF by using dataset from amazon has an accuracy of 99.6%. Paper [93] proposed a novel RS by using AE architecture with Movielens dataset resulted an accuracy of 87%.

The obtained result of research work accomplished by using Bi-LSTM algorithm in this research has an accuracy of 83.65%. This result is obtained after multiple attempts of hyper parameter tuning. Similar work in [129] for a traffic forecasting with the help of calibrated micro simulation database resulted in 87.33%. A movie RS using Bi-LSTM and Dilated CNN in [123] with TMDB dataset obtained 97.24% accuracy. The accuracy of K-Means Clustering of this work is 91.01% and CB filtering with HAC is 85.01%. Similar work in [116] for a journal RS by using K-Means algorithm by using RICEST journal finder dataset has accuracy of 80%. In [150] a disease prediction system using KNN algorithm with help of UCI ML repository dataset has accuracy of 83.62%. A hotel RS referred in [99] done by using LDA technique with the help of dataset from TripAdvisor resulted in accuracy of 89.19%.

Authors in [154] proposed a Hindi music RS which collected data from Romanised Hindi lyric dataset. Unsupervised stemming algorithm, self organised feature map (SOFM) and Dov2Vec methodologies used to complete the task with a result of F-measure 0.749. The work in [155] discussed Bangla news RS by using 3 methods of hierarchical clustering & NER. Data collected from Bangla news corpus and Google. The work observed better performance in reverse HC with cosine similarity. Another work in [156] about movie recommendation in Bengali language by using KNN and Cosine similarity methodology with IMDb database focused in chorki and Hoichoi platform contains 381 film with different genres. The performance metrics was RMSE 0.97 and MAE 0.75. A work in paper [157] for Hindi movie recommendation by using classification-based model & Graph Convolutional Network. Flickscore dataset was data source. The observed result was 89.47%. The accuracy of these experiments depends on the quality and quantity of records in the benchmark dataset and methodology used to implement.

2.6 Conclusion

In summary, this literature review sets the academic context and outlines the existing research relevant to this study. It also highlights the unique contributions of this research, which include the development of multiple recommender models and specialized tools for Malayalam language processing. These contributions address identified gaps in existing research and offer new avenues for academic exploration. The interdisciplinary nature of this study, which merges recommender systems, NLP, and domain-specific needs, adds a layer of complexity and novelty that makes it a particularly interesting focus for future academic endeavors. A dedicated lookup dictionary was developed as part of this study. This dictionary is tailored for processing Malayalam travelogues and aids in tasks such as tokenization, tagging, and semantic analysis [97]. The incorporation of this resource significantly enhances the system's ability to provide accurate and culturally relevant travel recommendations.

3 Research Methodology

3.1 Introduction

This chapter explores the processes involved in carrying out investigations starting with the step of extracting data. The extraction of information serves as the foundation for analysis providing insights into the phenomena being studied. Next, the focus emphasises into Natural Language Processing (NLP). Preprocessing techniques which are key for handling and understanding unstructured textual data. The chapter also addresses the complexities of feature engineering and encoding which are steps in transforming raw data into meaningful representations that power analytical tools. Finally, exploring neural networks, AI models for recommendation systems and models shedding light on methods used to gain insights and improve decision making processes. These methodological steps together provide a framework for conducting comprehensive research in a chosen field.

3.2 Data Collection

3.2.1 Data Source and Scope

The initial phase of the research involved extracting travelogues from Facebook's 'Sanchari' group, which resulted in a Spreadsheet containing unstructured data. By using the admin Insight option of Facebook, the growth of groups and engagements of users along with statistical details can be retrieved as a spreadsheet. The extraction process fetches all details about the travelogue, posted time, the profile of the user, user’s personal information which are publicly available. This unstructured travel review and associated information is stored in a spreadsheet, which is considered the primary data source. Apart from this, travelogues collected from travel blogs and websites. A structured dataset is created by sharing a google form to collected travel related data from public users.

3.2.2 Challenges in Data Preparation

The main challenges included dealing with the complex nature of the Malayalam language and the unstructured format of the scraped data. While the linguistic intricacies required specialized preprocessing, the lack of a structured format demanded rigorous data cleansing and transformation. All travelogues written English, Manglish or any language other than Malayalam has removed from spreadsheet before preprocessing. Advanced techniques were employed to navigate these challenges successfully, turning raw data into a structured dataset suitable for machine learning and deep learning algorithms. Utilizing an innovative two-phase approach, this study has gathered and processed 13458 unstructured travelogues and reviews in Malayalam.

3.3 Natural Language Processing and Preprocessing

Conventional preprocessing techniques were not sufficient to address the problem in handling the research work because of the complex morphology of Malayalam Language. Hence, this research employed advanced text preprocessing methods, including sentence and word tokenization, stop-word removal, and punctuation removal. Due to the complexities of the Malayalam language, the research also involved advanced techniques such as root-pack analysis for lemmatization, the construction of a specialized travelogue tagger named POT Tagger, exclusively for annotating travel-related tokens in the travelogue, and the creation of a look-up dictionary.

The objective was to ensure that the tokenized and processed text accurately represented the semantics of the original travelogues. Out of several features selected from each travelogue, the most distinctive features selected to uniquely identify the travel patterns of the users are Travel type, Mode of Travel, Location climate, Type of Location, and destination name. Travel type describes the way the traveller adopted to travel such as by bike, train, road, or flight. Travel mode explains with whom the travel is conducted such as with friends, family, or colleagues. Location climate explains which season the travel is conducted such as rainy season or summer. Location type is the feature that explains the activity or type of location such as adventurous, pilgrimage spot, scenic, etc.

3.4 Feature Engineering and Encoding

Feature engineering is a crucial step in making the machine learning model more effective and interpretable. One-hot encoding is used to transform categorical variables such as Travel Type, Travel Mode, Location climate, and Location Type into a form that could be fed into machine learning algorithms. For instance, if there are three types of travel modes like ‘കാർ; 'ട്രെയിൻ', and 'റോഡ്,' they will be represented as [1,0,0], [0,1,0], and [0,0,1] respectively after one-hot encoding. Travel DNA and Location DNA are innovative approaches to capturing the essence of each travelogue or location in a sequence of numbers. For example, the Travel DNA could be a vector that includes key aspects of individual users which are prepared based on their past travel histories of personal preferences like adventure, historical places, climate, bike traveller with friends, and so on, represented numerically based on the processed text. Similarly, Location DNA would encapsulate the features of various travel destinations in a numerical form. These "DNAs" serve as the foundational blocks for understanding and predicting user preferences. The feature engineering steps are depicted in Figure 1.

Feature engineering involves processing the travelogues by sentence tokenization and word tokenization. The tokens are extracted to their root words. The filtered tokens are compared with look-up dictionary and then annotated by POT Tagger. The tagged tokens then count for the frequency of occurrence to find the suitable features of the travel.

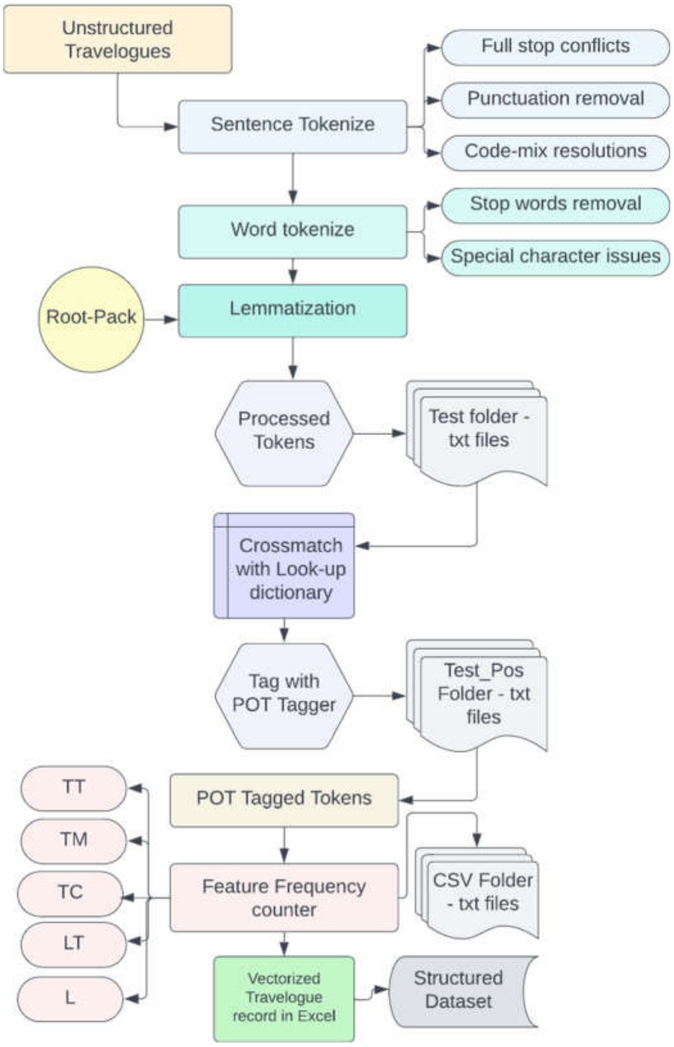


Figure 1 Feature Engineering process

Dataset preparation is an essential step in machine learning models, especially when dealing with unstructured data like Malayalam travelogues. After the initial scraping of travelogues from social media, each travelogue is processed through several NLP techniques, followed by conventional and advanced feature extraction procedures. Features such as Travel Type (TT), Travel Mode (TM), Location (L), Location Climate (LC), and Location Type (LT) are extracted and annotated. One-hot encoding is applied to these categorical features to generate a structured dataset for this travel domain in the Malayalam Language.

3.5 Recommendation Systems and Models

3.5.1 Rule-Based Cosine Similarity Recommender model.

The development of the recommendation model using rule-based cosine similarity is a crucial aspect of this study, designed to offer highly personalized travel recommendations. Unlike conventional machine learning-based models, this rule-based approach employs mathematical metrics to calculate the similarity between vectors, specifically focusing on the user-provided travel preferences and the pre-existing 'Travel DNA' and 'Location DNA' clusters. Upon interacting with the system, users are prompted to enter their preferred Travel Type (TT), Travel Mode (TM), and Location Type (LT). These input preferences are converted into a feature vector that is then compared with vectors in the 'Travel DNA' and 'Location DNA' clusters.

Cosine similarity calculations are performed between the user's input vector and the vectors in the existing clusters to find the best matching travel options. Based on the calculated cosine similarity scores, two sets of recommendations are generated: a primary list and a secondary list. The primary list comprises travel destinations that have the highest similarity scores, thereby closely matching the user's specified preferences and excluding any location that has already been visited. The secondary list serves as an alternative, offering locations that are also similar but with slightly lower cosine similarity scores and may contain pre-visited locations as well. The rationale behind offering a secondary list is to provide users with additional options that, while not perfectly aligned with their preferences, still bear a considerable resemblance, and might interest the user. This dual-list recommendation system enhances the user experience by not only meeting their exact requirements but also suggesting alternative options that they might find appealing.

3.5.2 Collaborative filtering based on K-Means Clustering

The development of a recommendation model using a Collaborative Filtering-Based K-means clustering Approach represents another key dimension of this research. Collaborative Filtering methods utilize user-item interactions to generate personalized recommendations. In this approach, K-Means clustering is integrated to group similar users based on their interactions with different travel destinations, creating a more personalized and effective recommendation system. The dataset of Malayalam travelogues was first transformed into a user-item matrix, which was then fed into the K-Means algorithm to identify user clusters. The performance evaluation of this model was rigorously conducted after obtaining experimental results.

3.5.3 Content-based Hierarchical agglomerative Clustering.

The development of the recommendation model using Content-Based Hierarchical Agglomerative Clustering (HAC) serves as a pivotal component of this research. Unlike K-means clustering, which partitions data into distinct clusters, HAC uses a tree-based structure to arrange data hierarchically, which is particularly beneficial for understanding nested relationships within travel preferences. Utilizing a content-based approach, the algorithm takes into consideration specific features such as travel type, location climate, and mode of travel from the meticulously curated Malayalam travelogues dataset. The user profiles and item profiles are constructed based on these attributes to perform the clustering. The HAC algorithm was implemented, and the dendrogram obtained served as a tool for understanding how different travel destinations or experiences could be grouped. The height and structure of the tree provided valuable insights into the hierarchical relationships between various travel options. The objective was to generate recommendations based on the closest hierarchical associations within the dendrogram.

3.5.4 Bi-LSTM Model

The creation of the recommender model using the Bidirectional Long Short-Term Memory (Bi-LSTM) technique is a significant milestone in this research. Adopting a Bi-LSTM architecture allows the model to capture temporal dependencies and contextual information from both past and future sequences, which is essential for interpreting complex travel patterns in Malayalam travelogues. The methodology involves initially encoding features like Travel Type, Travel Mode, and Location DNA into numerical vectors, which serve as the input to the Bi-LSTM layers. During the experimentation phase, the model was fine-tuned using various hyperparameters, including learning rates and activation functions, and was trained for a total of 1400 epochs to ensure performance optimization. The model's performance was rigorously evaluated using test and validation sets, adopting metrics like accuracy and loss to assess its reliability. The experimental results confirm the viability of using Bi-LSTM techniques in the construction of a robust and accurate travel recommender system, especially in the context of the Malayalam language.

3.5.5 Experiment with Deep Autoencoder

The development of a recommender model using autoencoder techniques is another pivotal aspect of this research project. Autoencoders serve as a powerful tool for dimensionality reduction and feature learning, making them suitable for capturing the intricate patterns within Malayalam travelogues. The methodology adopted involves feeding the pre-processed and encoded travel data into the input layer of the autoencoder, which then passes through hidden layers to get compressed and subsequently reconstructed. The objective is to minimize the reconstruction error, which ensures that the most salient features related to travel recommendations are captured effectively. In terms of experimental performance, the model underwent multiple iterations with different hyperparameters and architectures to optimize its ability to generalize across new data.

3.6 Organization of the Thesis

The thesis is systematically organized into eight key chapters to offer a comprehensive guide on the development of a personalized travel recommender system using Malayalam travel reviews. The overall organization of chapters in this thesis is represented in Figure 2.

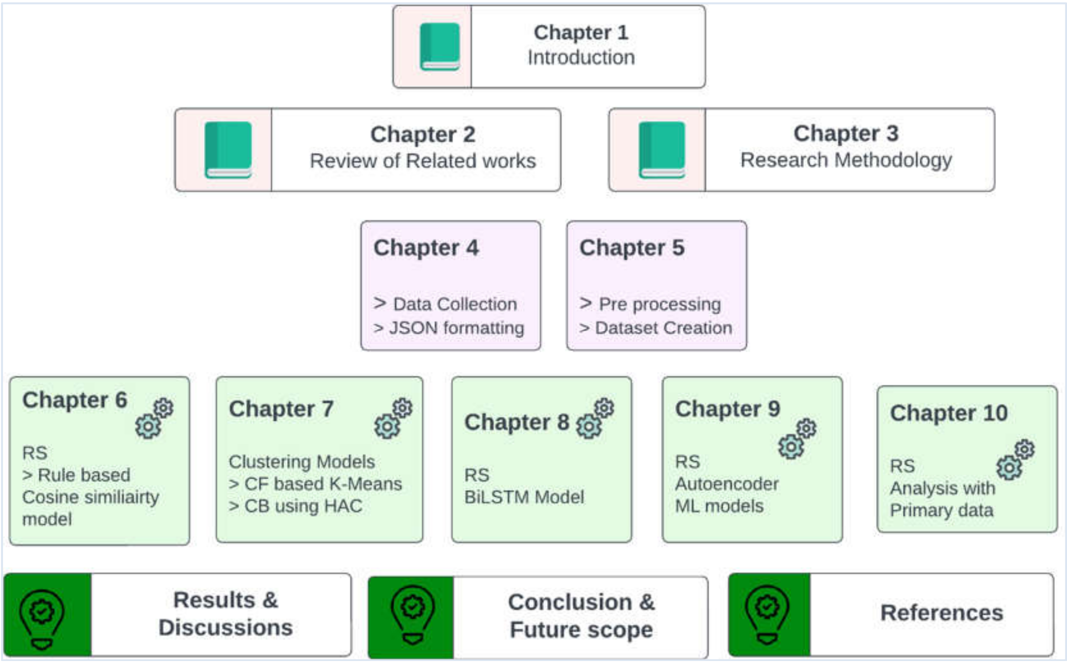


Figure 2 Organization of Chapters

Chapter 1 sets the stage with an introduction that covers the research scope, challenges, and methodologies. Chapter 2 dives into the existing literature relevant to recommender systems, NLP, and machine learning techniques. Entire research methodology discussed in Chapter 3. Data collection processes and challenges, specifically focusing on scraping Malayalam travelogues from Facebook, are detailed in Chapter 4, while Chapter 5 discusses transforming this raw data into a structured dataset using various NLP techniques. Recommendation models form the core of Chapters 6 to 9, with Chapter 6 focusing on a rule-based cosine similarity model, Chapter 7 covering models based on K-Means clustering and Hierarchical Agglomerative Clustering, Chapter 8 detailing a model using Bi-LSTM, and Chapter 9 exploring an autoencoder-based recommendation system. Chapter 10 focused on analysis of various travel tastes and preferences of users based on age, sex, gender and transportation preferences. Results of each experiment and discussions given in chapter 11.

Conclusion of work and future directions given in Chapter 12. Each chapter serves both as a standalone exploration of its topic and as an integral part of the thesis' overarching objective of building a robust Malayalam-language travel recommender system.

3.7 Conclusion

This chapter provides an overview of entire works furnished in this thesis. The pictorial representation of organization of chapters in this work gives a clear idea about topics discussed in the work. Primary focus of the research moves to data collection process. Data collection methods, Challenges and limitations and unavailability of structured dataset discussed.

The next focus on Natural Language Processing (NLP) and preprocessing emphasizes the importance of handling text data so that we can gain meaningful insights, from it. Feature engineering and encoding are components that helped to transform data into understandable representations, which then drive subsequent analyses. Various machine learning algorithms and deep clustering techniques used for developing recommendation models for users and their performance evaluations are also examined in the chapter.

4 Data Collection from Social Media and Travel Blogs

4.1 Introduction

In the era of digitalization, the proliferation of social media platforms has given rise to a wealth of user-generated content spanning a multitude of topics, including travel experiences. These platforms serve as virtual canvases where individuals from diverse backgrounds share their journeys, memories, and insights about destinations they've explored. The evolving landscape of social media has opened unprecedented opportunities for research in various domains, including personalized travel recommendation systems. This chapter delves into the intricate process of crafting a benchmark dataset for the development of personalized travel recommender systems in the context of the Malayalam language, harnessed through an innovative approach to social media scraping.

The significance of travel recommendation systems has grown significantly in recent times, as modern travelers increasingly seek tailored suggestions that align with their preferences, interests, and personal contexts. While existing research has demonstrated the potential of recommendation systems, there remains a conspicuous dearth of studies focusing on personalized travel recommendations in languages other than English. This void, particularly in the context of the rich and intricate Malayalam language, underscores the importance of pioneering research endeavors.

4.2 Challenges in Dataset Creation

Creating a benchmark dataset for personalized travel recommendation systems in the Malayalam language posed multifaceted challenges. The scarcity of prior work, coupled with the absence of readily available benchmark datasets, demanded innovative methods to curate a dataset of sufficient scale and quality.

The key hurdle lay in sourcing data that accurately reflected the nuances of travel experiences expressed by individuals in the Malayalam-speaking community. Overcoming this challenge entailed scraping, collecting, and preprocessing data from diverse sources.

4.2.1 Lack of Prior Work

In the field of travel recommendation systems, especially for the Malayalam language, a significant gap exists in terms of prior research and existing solutions. Prior work is crucial for building upon existing knowledge and methodologies, but in this case, there is a deficiency of established systems or studies that have addressed personalized travel recommendations in Malayalam. This lack of prior work highlights the innovative and exploratory nature of this research work. Travel recommendation systems are a critical component of the tourism industry, offering users personalized suggestions for destinations, activities, and accommodations. While several such systems exist for popular languages and travel destinations, the scarcity of research and systems specifically designed for the Malayalam-speaking community is a notable limitation. The proposed work is novel in this regard, as it seeks to bridge this gap and provide a foundation for future work in the domain of Malayalam travel recommendation.

4.2.2 Absence of a Benchmark Dataset

A benchmark dataset serves as a standard reference point against which the performance of a system or algorithm can be measured. It is a critical resource for evaluating the effectiveness and accuracy of any recommendation system. In this case, the absence of a benchmark dataset for Malayalam travel recommendations further highlights the pioneering nature of research. Benchmark datasets typically consist of well-annotated, high-quality data that is representative of the target domain. These datasets enable researchers to benchmark their systems against a common set of data, facilitating fair comparisons and evaluations. Without such a benchmark for Malayalam travel recommendations, the proposed work faces the challenge of establishing the foundational dataset from scratch.

The processes in collecting and processing travelogues, as well as in developing a Travelogue Tagger, are essential steps in addressing this challenge. Creating a structured dataset from unstructured travel narratives in Malayalam is not only an innovative solution but also a fundamental contribution to the field of travel recommendation. This dataset could potentially become the benchmark against which future researchers evaluate their systems in the domain of Malayalam travel recommendations.

4.3 The Pioneering Approach

To bridge the gap between the scarcity of available resources and the necessity for a comprehensive dataset, curating a dataset, mainly depended on social media, the largest repository of information of all kinds. Social media platforms like Facebook.com, travel segment of media channels like Madhyamam Travel, Mathrubhumi Travel, Manorama Travel, and Travel blogs like Malayalam Native Planet, and YathrikanOnRoad.com are used for collecting travel blogs[98]. The diagrammatic representation of data collection processes is given in Figure 3.

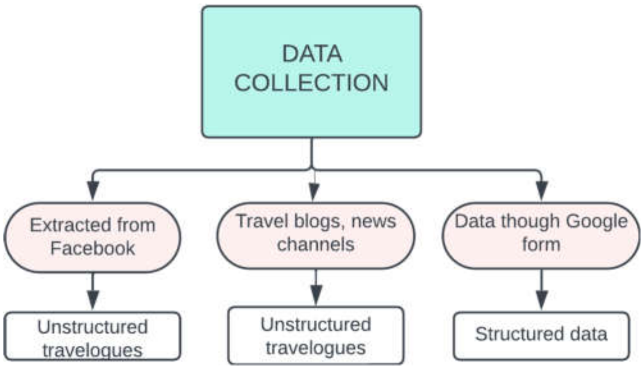


Figure 3 Data collection process

4.4 Extraction of Travelogues from Facebook Group Admin Insights

The primary source of data for this study is the 'Sanchari' Facebook group. Facebook groups are virtual communities where users with similar interests can come together to share content, engage in discussions, and exchange information.

This method involves obtaining a substantial portion of dataset by leveraging the "Admin Insights" feature of a prominent Facebook group. ‘Sanchari’ is the largest travel group in the Malayalam Language in Kerala. The group is exclusively for sharing travel experiences of users, posting photos and reviews about locations, and asking questions about various destinations all over the world. As of June 2023, more than seven lakh people have joined this group. It contains the largest collection of Malayalam descriptive travelogues and adds an average of 50 travelogues per day. This approach focuses on the extraction of travel reviews shared by group members. It's a unique and valuable source of first hand travel experiences provided by the group's members.

4.4.1 Access to Admin Insights

Facebook Admin Insights, commonly known as Facebook Group Insights, serves as a comprehensive analytical tool designed for Facebook Page administrators. It offers a wealth of data and metrics that provide critical insights into the performance and engagement of a Facebook group. One key aspect it illuminates is audience insights, offering in-depth demographic and behavioural information about the group’s members and followers, including age, gender, location, and active hours. Understanding this audience profile is fundamental for tailoring content to effectively resonate with followers. Moreover, Facebook Admin Insights investigates various engagement metrics, such as likes, comments, shares, and post clicks, which enable administrators to analyse the effectiveness of their content strategy. It also provides data on reach and impressions, helping administrators understand the visibility of their posts. Additionally, insights into page views, actions taken on the page, video performance, and growth in page likes equip administrators with a holistic view of their page's performance.

The group growth and engagement section of admin insight provides a facility to retrieve the travelogues and user information, which can then be conveniently downloaded in spreadsheet format. The downloading process offers flexibility by allowing the selection of data for specific timeframes, including options for 7 days, 14 days, and 28 days. To compile a robust dataset, multiple downloads were performed from this feature, and the resulting files were subsequently merged to create the comprehensive dataset used for this study.

4.4.2 Valuable User-Generated Content

Extracting travel reviews directly from this Facebook group's Admin Insights ensures that the data is user-generated and highly authentic. It reflects the diverse travel experiences of the group's members, making it a valuable resource for the study.

4.4.3 Metadata Enrichment

Along with the textual content, it provides the facility to collect essential metadata, including user details (usernames), post-dates, and engagement metrics (comments, likes, shares, reactions). This metadata is crucial for understanding user preferences and the popularity of specific travel reviews within the group.

4.4.4 Ethical Data Collection

Emphasize that data collection was conducted ethically and in compliance with Facebook's data usage policies and community guidelines. The necessary permissions and approvals were obtained to access and analyse group data.

4.5 Travelogues from Travel Blogs and Media Channels

This method complements the Facebook data by sourcing travelogues from various external platforms, such as travel blogs and media channels like Manorama, Mathrubhumi, Madhyamam, and independent travel bloggers. To ensure the dataset's richness and diversity, this work tapped into various online sources. These sources include reputable news portals and established Malayalam media channels like Mathrubhumi, Manorama, and Madhyamam which are exclusively focused on Travel Domain with umpteen Reviews written in Malayalam. In addition, the data collection process explored online travel blogging platforms such as YathrikanOnRoad.com and MalayalamNativePlanet, where enthusiastic travelers often share detailed travelogues about their journeys. Collecting these travelogues involved a manual process. This means that personally visited these websites, identified relevant travel articles or reviews, and extracted the text content. Manual data collection allows to be selective, ensuring that gather high-quality and pertinent travel narratives.

4.5.1 Diversity in Sources

The research systematically targeted a diverse set of sources, which include established media outlets as well as individual travel bloggers. This diversity ensures that the dataset includes a broad spectrum of travel experiences, from professionally curated articles to personal narratives.

4.5.2 Data Structuring

Once these travelogues are collected, organize them by adding them to a spreadsheet. This spreadsheet serves as a structured repository for the unstructured travel narratives. Structuring the data in this manner makes it more manageable and accessible for subsequent analysis.

4.5.3 Content Variety and Data Enrichment

Given the diversity of sources, dataset likely contains travel narratives covering a wide range of destinations, experiences, and writing styles. This diversity is valuable for research as it enables to capture various facets of travel experiences and preferences. Along with this information, the process included steps to enrich the dataset by including additional metadata. This could include information such as the publication date of each travelogue, the author's details, or any other relevant contextual information that might be useful for further analysis.

4.6 Data Collection Through Google Form

The data collection process for this research was multifaceted, involving both the extraction of reviews from social media and the collection of primary data through crowdsourcing. A Google Form was designed to capture user-generated data related to travel experiences. The form included a variety of questions relevant to travel, such as travel type, mode of travel, preferred locations, and climate, as well as demographic information like age and gender. Additionally, participants were asked to rate their chosen travel destinations on a 5-point scale. The Google Form was shared broadly to target a diverse group of travelers, aiming to obtain responses from people of various age groups and genders. This approach ensured that the dataset would be comprehensive, capturing the multifaceted preferences and behaviours of travellers.

The primary data for conducting this research is based on an online survey. The data collected from 2006 responses online, especially through different travelers' platforms and social media chat groups. The respondent of the study includes 2006 individuals from entire districts of the Kerala state, and the survey questions are asked about their travel behaviour and the most preferred destination which is located either in Kerala or outside Kerala. For the analysis purpose, the study has employed percentage analysis, cross-tabulation, independent sample t-test, and analysis of variance. These 2006 responses were then exported to a Spreadsheet, where the fields were carefully designed to align with the structured dataset generated from the Facebook scraping process.

These three distinct data collection methods together contributed to a robust and diverse dataset, encompassing a wide array of travel experiences and preferences within the Malayalam-speaking community. They form the foundation for research in building a personalized travel recommender system for the Malayalam language. The travelogues and travel reviews collected from various online sources are listed in Table 1.

Table 1 Data Collection -Travel reviews from data sources

Sources	No. of posts	Remarks
Admin Insight of Sanchari Group	12500	Unstructured Reviews- Malayalam
Manual Data Collection	403	Spreadsheet prepared in English
Yathikanonroad.com	125	Travel Blogs - Malayalam
Mathrubhumi/travel	105	
MalayalamNativePlanet	325	
	13458	
Google Form as primary data	2006	Structured fields - Malayalam
Check-ins Facebook	84463	Extracted from FB - English
	99927	

After the data collection phase, the spreadsheet is created with 13458 full-length unstructured travelogues in Malayalam language, 2006 structured response from google form. Check-ins extracted from Facebook not used in this work.

4.7 Research Design

The research design of this work is a structured approach divided into four core segments: Data Collection, Dataset Preparation, Feature Engineering and Construction of Recommendation Models. The first segment focuses on accumulating data from various online sources such as Facebook.com, other travel blogs and through Google Forms. The second segment is dedicated to dataset refinement, employing NLP techniques for preprocessing and token annotation, and it further incorporates a Lookup Dictionary for feature cross-matching. The third segment describes the feature engineering process of these processed tokens to a numerical form that can be used by various machine learning models. The final segment involves creating various recommendation systems using different algorithms, including Rule-Based Cosine Similarity, Clustering Techniques, Bi-LSTM, Autoencoders, and an ensembled model of deep autoencoder with machine learning algorithms. This methodical design aims to facilitate the development of a robust, personalized travel recommender system specialized for Malayalam language users. The wireframe of research design is given in Figure 4.

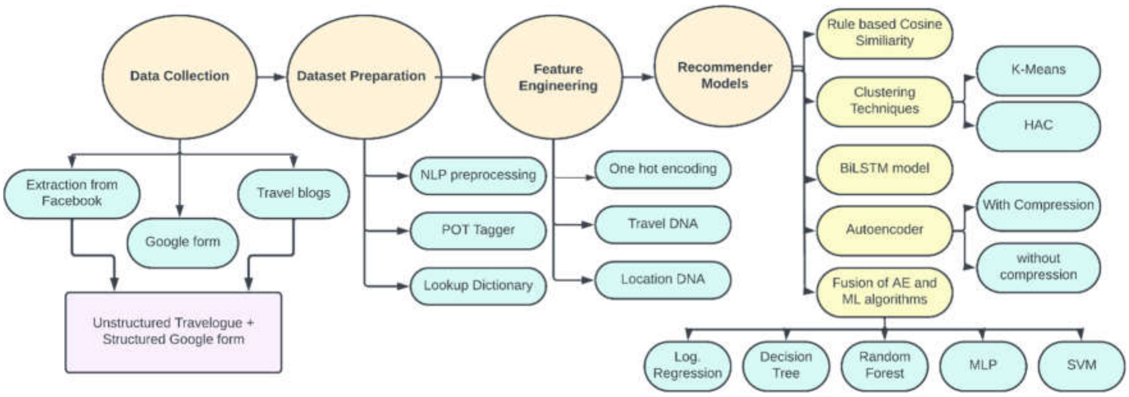


Figure 4 Stages involved in Research Design

4.8 Conclusion

In conclusion, the endeavour to extract and analyse data from Facebook groups presents a complex yet rewarding journey. This chapter has explored the multifaceted challenges inherent in this process and explained innovative solutions employed to scale them. The initial phase of data extraction showcased the algorithm's prowess in navigating the intricacies of Facebook's platform. With the help of admin insight, the customised extraction of travel posts and associated details are retrieved and stored in the spreadsheet. Other prominent travel websites are searched for travel blogs posted by independent users. Gathering all these reviews forms the preliminary dataset in the unstructured format.

The algorithm's second phase showcased its adaptability to Natural Language Processing, refining Malayalam text data through preprocessing techniques like tokenization, stemming, and sentiment analysis [99]. This enriched textual dataset lays the foundation for a potent recommender system, poised to cater personalized travel recommendations. Through innovation and meticulous execution, the algorithm empowers researchers to unravel the potential of social media data for tailored insights and enriched user experiences in the realm of travel recommendations [100].

5 Structuring Malayalam Travelogues to Benchmark Dataset

5.1 Introduction

This research sets out to tackle the multifaceted challenge of creating a structured dataset from 13,458 Malayalam travelogues, originally scraped from social media platforms. A meticulous and tailored approach was adopted to handle the highly inflectional and morphologically [101] rich nature of the Malayalam language. The undertaking of this task was not just an attempt to transmute unstructured travelogues into an accessible dataset, but also an exploration into the underlying patterns, preferences, and peculiarities of travel within the Kerala context. By establishing a novel and systematic approach to preprocess and annotate the travelogues, this study aims to break new ground in the realm of personalized travel recommender systems for the Malayalam-speaking populace. In doing so, it hopes to create a tangible connection between technology and cultural heritage, reflecting the true essence of travel in Kerala.

Embarking on the unprecedented task of creating a structured dataset from unstructured Malayalam travelogues, this study unveils a new pathway in personalized travel recommendation systems. The inherent complexity of the Malayalam language, coupled with the diversity of travel experiences narrated, called for an innovative approach. By crafting a Part of Travelogue Tagger, a unique tool was developed to annotate each processed token. A critical part of this approach was the creation of a look-up dictionary containing several predefined tags, designed to categorize every possible item from the processed tokens, cross-matching it with the corresponding tag in the look-up dictionary. This allowed the extraction of essential features such as travel Type (TT), Travel Mode (TM), Location (L), Location Climate (LC), Location Type (LT), and Username (U). Furthermore, specialized preprocessing techniques, including sentence tokenization, word tokenization, removal of punctuations, code-mixing, stop words, stemming, and lemmatization, were employed to handle the highly inflectional and morphologically rich nature of the Malayalam language. This rigorous methodology resulted in a set of processed tokens for each travelogue, transforming scattered narratives into actionable insights.

In the chapters that follow, the intricate stages of preprocessing, experimental analysis, and key findings will be detailed, shedding light on the methodological rigor and innovative techniques employed in this pioneering work.

5.2 Data Cleaning and Preprocessing

Feature engineering in the context of Malayalam travelogues is a complex and nuanced process, owing to the diversity and variability in content length, writing styles, and language mixing found within the data. Travel posts can range from lengthy narratives to brief comments, and often include a mix of Malayalam and English, reflecting the linguistic intricacies of the region. This heterogeneity necessitates an adaptive and sensitive approach to extracting relevant features such as Travel Type, Travel Mode, Location, Location type, climate and other attributes. Through a combination of conventional preprocessing, custom functions tailored to Malayalam's morphological richness, and innovative tools like the Part of Travelogue Tagger and Look-Up Dictionary, the research aims to transform this unstructured and diverse data into a structured format, capable of fuelling a personalized travel recommender system that captures the authentic context and experiences described in the travelogues. Stages of feature engineering are given in Figure 5.

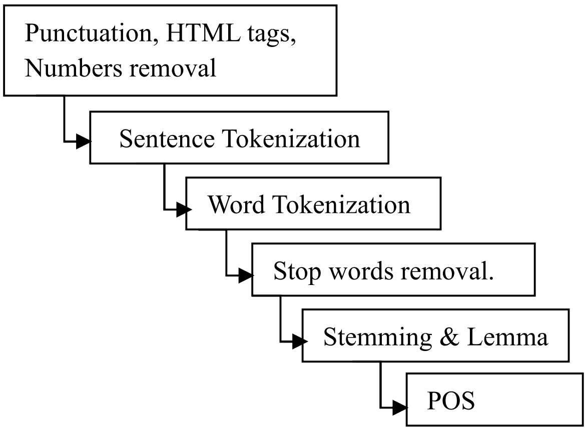


Figure 5 Steps involved in text pre-processing.

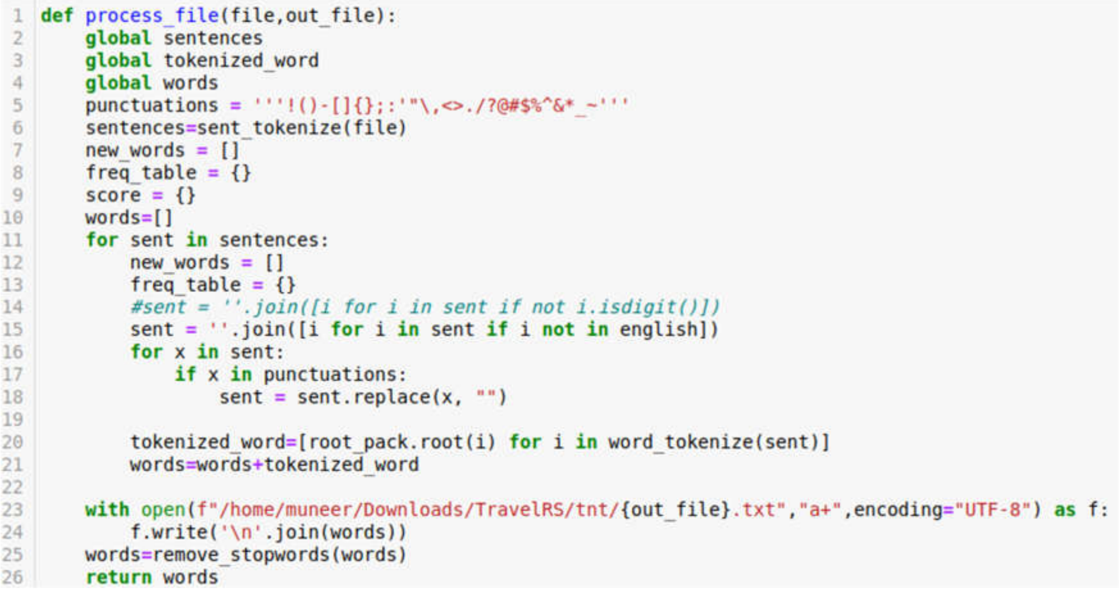
5.2.1 Preprocessing of Travelogues:

The extracted travelogues, encompassing a wide array of relevant features, are stored in a Spreadsheet that consists of 13458 rows. Within this dataset, the column named 'message' houses the unstructured and unprocessed travelogues, which are often lengthy and contain a substantial amount of noise. These raw travelogues serve as the primary input for the processing stage, where the goal is to meticulously extract the most significant features. These extracted elements are then utilized to populate the corresponding entries in a structured dataset, transforming the disparate and unorganized information into a coherent and analysable form.

This module focuses on the initial treatment of raw Malayalam travelogues. It includes techniques such as sentence tokenization, word tokenization, removal of punctuation, code-mixing, handling of stop words, stemming, and lemmatization. Given the inflectional and morphological richness of the Malayalam language, this phase involves specific functionalities tailored to the nuances of the language, setting the groundwork for subsequent stages.

5.2.2 Sentence Tokenization and Word Tokenization

Sentence tokenization of the Malayalam travelogues involves breaking down the extensive and often complex narratives into individual sentences, enabling more precise analysis and processing of the text. Sentence tokenization is not a one-size-fits-all process, especially for a morphologically rich language like Malayalam[102]. It involves various techniques, rules, and tools, all tailored to the specific characteristics and challenges of the language to ensure that the text is accurately divided into individual sentences. This process involves understanding the unique syntactic and grammatical rules of the Malayalam language and using punctuation marks and specific delimiters to identify sentence boundaries, multiple dots (…), and special characters to denote the end of sentences. Specialized tools like the Punkt tokenizer from the Natural Language Tool Kit (NLTK) are employed here. Figure 6 describes the Python code snippet of tokenization.



```python
def process_file(file,out_file):
    global sentences
    global tokenized_word
    global words
    punctuations = '''!()-[]{};:'"\,<>./?@#$%^&*_~'''
    sentences=sent_tokenize(file)
    new_words = []
    freq_table = {}
    score = {}
    words=[]
    for sent in sentences:
        new_words = []
        freq_table = {}
        #sent = ''.join([i for i in sent if not i.isdigit()])
        sent = ''.join([i for i in sent if i not in english])
        for x in sent:
            if x in punctuations:
                sent = sent.replace(x, "")

        tokenized_word=[root_pack.root(i) for i in word_tokenize(sent)]
        words=words+tokenized_word

    with open(f"/home/muneer/Downloads/TravelRS/tnt/{out_file}.txt","a+",encoding="UTF-8") as f:
        f.write('\n'.join(words))
    words=remove_stopwords(words)
    return words
```

Figure 6 Python implementation of text tokenization

Let T be the travelogue, and let S={s1,s2,…,sn} be the set of sentences in T. The sentence tokenization can be mathematically represented as a function f that maps T to S,

f(T)=Si Equation (1)

where si represents an individual sentence within the travelogue.

Word tokenization follows the sentence tokenization stage, further dissecting each sentence into individual words or tokens, reflecting the morphological richness and unique structure of the Malayalam language. Now, for each sentence si, further, break it down into individual words or tokens.

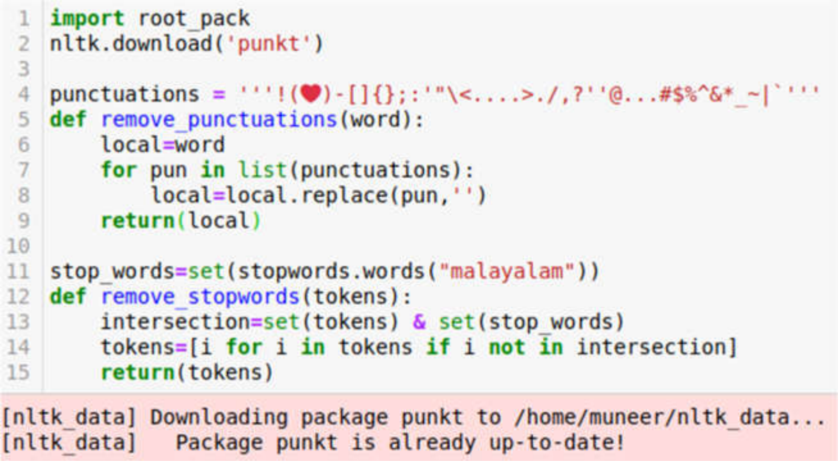
Let Wi={w1,w2,…,wm} be the set of words in sentence si. The word tokenization can be mathematically represented as a function g that maps each sentence is to a set of words Wi:

$$g(Si)=Wi \qquad \text{Equation (2)}$$

where Wi represents an individual word within sentence Si.

5.2.3 Removal of Punctuation and Special Characters

The removal of impurities from the tokenized Malayalam travelogues is a crucial step to ensure that only relevant information is considered for analysis. This process involves the elimination of non-Malayalam tokens, such as English words, which are regarded as irrelevant. Punctuation, numbers, and emojis are also targeted for removal, as shown in Figure 7, as they may not contribute meaningful information to the study. Specific Python packages, including the regex module, are utilized to perform these operations, and additional normalization and cleaning may be implemented to standardize the tokens.



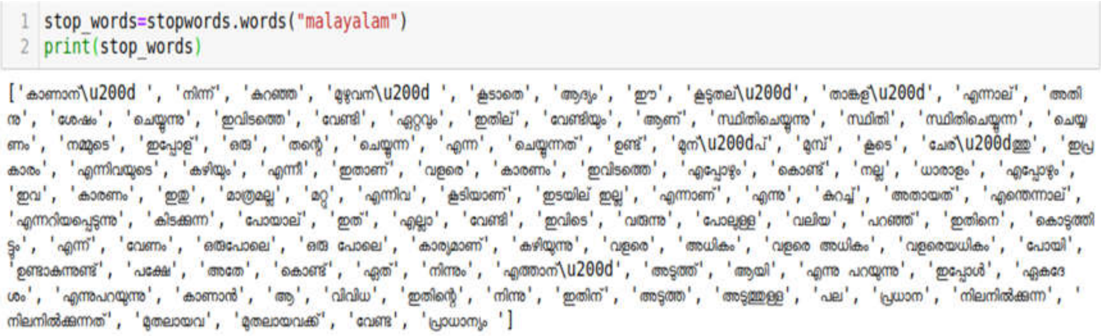
```
import root_pack
nltk.download('punkt')

punctuations = '''!(❤)-[]{};:'"\<....>./,?''@...#$%^&*_~|`'''
def remove_punctuations(word):
    local=word
    for pun in list(punctuations):
        local=local.replace(pun,'')
    return(local)

stop_words=set(stopwords.words("malayalam"))
def remove_stopwords(tokens):
    intersection=set(tokens) & set(stop_words)
    tokens=[i for i in tokens if i not in intersection]
    return(tokens)
```

```
[nltk_data] Downloading package punkt to /home/muneer/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
```

Figure 7 Removal of punctuation code snippet.

5.2.4 Removal of Stop Words

The removal of stopwords is an essential step in text processing, especially in the context of Malayalam travelogues. Stopwords are words that often appear in a text but typically do not carry significant meaning by themselves, such as conjunctions, prepositions, and common adverbs. In Malayalam, these may include certain particles and inflected forms that are ubiquitous in the language but do not contribute to the semantic analysis. For this work, a list of 114 stopwords has been identified as given in Figure 8 and used to filter the cleaned data. These stopwords are removed from the tokenized travelogues, leaving behind words and tokens that are more likely to carry substantial meaning and context. The process of removing stopwords helps to reduce the dimensionality of the dataset, making subsequent analysis, such as machine learning modelling, more efficient and meaningful.



```
stop_words=stopwords.words("malayalam")
print(stop_words)
```

```
['കാണാന്\u200d ', 'നിന്ന്', 'കുറഞ്ഞ', 'മുഴുവന്\u200d ', 'കൂടാതെ', 'ആദ്യം', 'ഈ', 'കൂടുതല്\u200d', 'താങ്കള്\u200d', 'എന്നാല്', 'അതി
നു', 'ശേഷം', 'ചെയ്യുന്നു', 'ഇവിടത്തെ', 'വേണ്ടി', 'ഏറ്റവും', 'ഇതില്', 'വേണ്ടിയും', 'ആണ്', 'സ്ഥിതിചെയ്യുന്നു', 'സ്ഥിതി', 'സ്ഥിതിചെയ്യുന്ന', 'ചെയ്യു
ണം', 'നമ്മുടെ', 'ഇപ്പോള്', 'ഒരു', 'തന്റെ', 'ചെയ്യുന്ന', 'എന്ന', 'ചെയ്യുന്നത്', 'ഉണ്ട്', 'മുന്\u200dപ്', 'മുമ്പ്', 'കൂടെ', 'ചേര്\u200dത്തു', 'ഇപ്ര
കാരം', 'എന്നിവയുടെ', 'കഴിയും', 'എന്നീ', 'ഇതാണ്', 'വളരെ', 'കാരണം', 'ഇവിടത്തെ', 'എപ്പോഴും', 'കൊണ്ട്', 'നല്ല', 'ധാരാളം', 'എപ്പോഴും',
'ഇവ', 'കാരണം', 'ഇതു', 'മാത്രമല്ല', 'മറ്റ്', 'എന്നിവ', 'കൂടിയാണ്', 'ഇടയില് ഇല്ല', 'എന്നാണ്', 'എന്നു', 'കുറച്ച്', 'അതായത്', 'എന്തെന്നാല്',
'എന്നറിയപ്പെടുന്നു', 'കിടക്കുന്ന', 'പോയാല്', 'ഇത്', 'എല്ലാ', 'വേണ്ടി', 'ഇവിടെ', 'വരുന്നു', 'പോലുള്ള', 'വലിയ', 'പറഞ്ഞ്', 'ഇതിനെ', 'കൊടുത്തി
ട്ടും', 'എന്ന്', 'വേണം', 'ഒരുപോലെ', 'ഒരു പോലെ', 'കാര്യമാണ്', 'കഴിയുന്നു', 'വളരെ', 'അധികം', 'വളരെ അധികം', 'വളരെയധികം', 'പോയി',
'ഉണ്ടാകുന്നുണ്ട്', 'പക്ഷേ', 'അതേ', 'കൊണ്ട്', 'ഏത്', 'നിന്നും', 'എത്താന്\u200d', 'അടുത്ത്', 'ആയി', 'എന്നു പറയുന്നു', 'ഇപ്പോൾ', 'ഏകദേ
ശം', 'എന്നുപറയുന്നു', 'കാണാൻ', 'ആ', 'വിവിധ', 'ഇതിന്റെ', 'നിന്നു', 'ഇതിന്', 'അടുത്ത', 'അടുത്തുള്ള', 'പല', 'പ്രധാന', 'നിലനിൽക്കുന്ന', '
നിലനിൽക്കുന്നത്', 'മുതലായവ', 'മുതലായവക്ക്', 'വേണ്ട', 'പ്രാധാന്യം ']
```

Figure 8 List of stop words in Malayalam language.

5.2.5 Stemming and Lemmatization

Stemming is a fundamental step in natural language processing, particularly for languages that have complex morphological structures like Malayalam. This language is considered one of the most agglutinative in India [103], meaning that words often consist of a series of morphemes (the smallest units of meaning) that are combined. This complexity poses significant challenges in extracting the most significant or root part of the words. Given Malayalam's rich morphology [104], the stemming process must be sophisticated enough to accurately derive the root of a word, irrespective of the number of suffixes or additional morphemes attached to the stem. Samples of stopwords, Malayalam tokens and its lemmas with English translation given in Table 2.

Table 2 Samples of stopwords, Malayalam tokens and its root word

Stop words		Tokens and root words in Malayalam and English		
Malayalam	**English**	**Tokens**	**Root words**	**English**
അത്	That	പോയിരിക്കും	പോകുക	Go
അങ്ങനെ	So	കുടുംബമായി	കുടുംബം	Family
മതി	enough	കാറിലേക്ക്	കാർ	Car
എങ്കിൽ	If	തണുപ്പിന്റെ	തണുപ്പ്	Cool
മറ്റു	other	മരത്തിന്റെ	മരം	tree

The Root-pack package developed at ICFOSS in Trivandrum is specifically designed to tackle this challenge. It's a specialized module capable of implementing the stemming process for Malayalam, understanding its intricate structure, and accurately extracting the root of any given words. By doing so, it plays a crucial role in transforming the unstructured travelogues into a structured dataset.

A language processing module named 'root-pack' specially developed to find out the root words of Malayalam tokens. For example,

Given a sample Malayalam inflated word "ഞങ്ങൾക്കെല്ലാവർക്കുമായി”

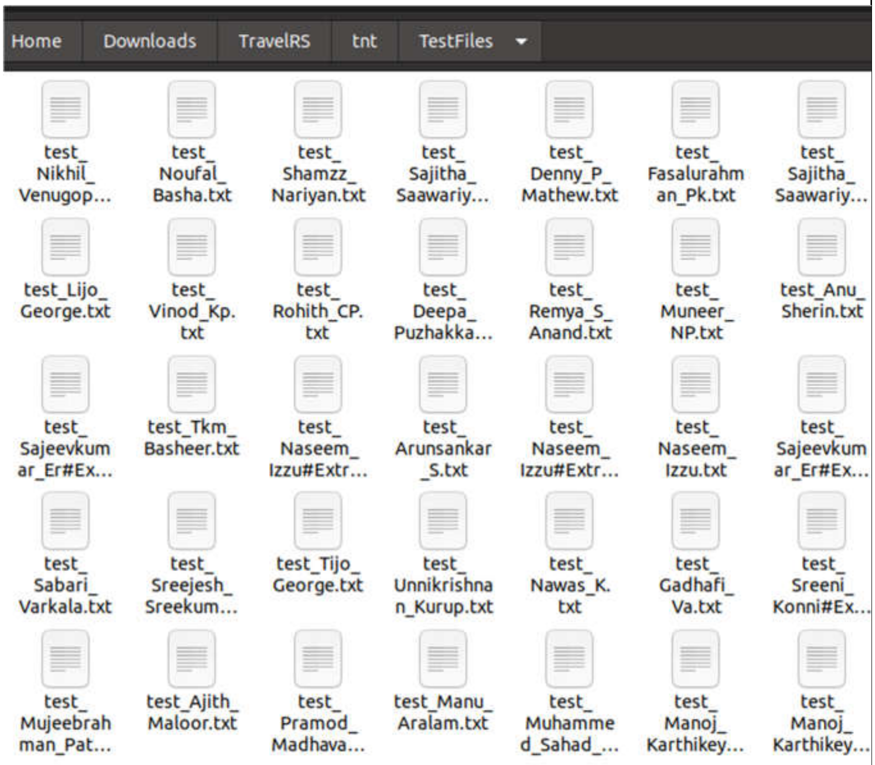
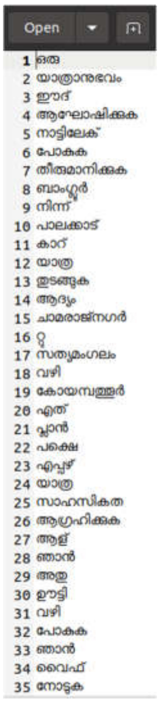
import root_pack

root_pack.root(“ഞങ്ങൾക്കെല്ലാവർക്കുമായി”)

The extractor will find out the root word as “ഞങ്ങൾ”.

All the processed tokens of a travelogue are then kept in a separate text file within a folder named 'test'. So, there are 13458 processed files for 13458 travelogues. Each file contains the cleaned, filtered, and processed tokens corresponding to each travelogue. Table 3 shows the file structure and a sample file opened condition.

Table 3 File structure and a sample file opened condition.

Files stored in the folder	Content of an Opened file (test_Nikhil_Venugopal)
 Home Downloads TravelRS tnt TestFiles test_Nikhil_Venugop... test_Noufal_Basha.txt test_Shamzz_Nariyan.txt test_Sajitha_Saawariy... test_Denny_P_Mathew.txt test_Fasalurahman_Pk.txt test_Sajitha_Saawariy... test_Lijo_George.txt test_Vinod_Kp.txt test_Rohith_CP.txt test_Deepa_Puzhakka... test_Remya_S_Anand.txt test_Muneer_NP.txt test_Anu_Sherin.txt test_Sajeevkumar_Er#Ex... test_Tkm_Basheer.txt test_Naseem_Izzu#Extr... test_Arunsankar_S.txt test_Naseem_Izzu#Extr... test_Naseem_Izzu.txt test_Sajeevkumar_Er#Ex... test_Sabari_Varkala.txt test_Sreejesh_Sreekum... test_Tijo_George.txt test_Unnikrishnan_Kurup.txt test_Nawas_K.txt test_Gadhafi_Va.txt test_Sreeni_Konni#Ex... test_Mujeebrahman_Pat... test_Ajith_Maloor.txt test_Pramod_Madhava... test_Manu_Aralam.txt test_Muhammed_Sahad_... test_Manoj_Karthikey... test_Manoj_Karthikey...	 Open 1 ഒരു 2 യാത്രാനുഭവം 3 ഈദ് 4 ആഘോഷിക്കുക 5 നാട്ടിലേക്ക് 6 പോകുക 7 തീരുമാനിക്കുക 8 ബാംഗ്ലൂർ 9 നിന്ന് 10 പാലക്കാട് 11 കാറ് 12 യാത്ര 13 ഉടങ്ങുക 14 ആദ്യം 15 ചാമരാജ്നഗർ 16 ഉ 17 സത്യമംഗലം 18 വഴി 19 കോയമ്പത്തൂർ 20 എത്ത് 21 പ്ലാൻ 22 പക്ഷെ 23 എപ്പഴ് 24 യാത്ര 25 സാഹസികത 26 ആഗ്രഹിക്കുക 27 ആള് 28 ഞാൻ 29 അത് 30 ഊട്ടി 31 വഴി 32 പോകുക 33 ഞാൻ 34 വൈഫ് 35 നോട്ടുക

5.3 Configuring Part of Travelogue Tagger

The challenge of the absence of a Tagger suitable for annotating Malayalam text within the travel domain was overcome by developing a new Tagger, called the Part of Travelogue tagger (POT)[105]. This was achieved by modifying an existing Part of Speech tagger [106] named Dhwanimam, which was created by ICFOSS, and adapting it to the specific requirements of the travel domain. The newly created travel tagger includes special tags that are particularly relevant to travelogues as shown in Table 4. These include TM for travel mode, denoting variations such as solo travel, travel with friends, family, spouse, etc.; TT for travel type, categorizing transportation means like trains, buses, cars, bicycles, motorcycles, flights, and so on; L for location, marking the places described in the travelogues; LT for Location Type; and LC for location climate, tagging the weather conditions of the locations such as sunny, rainy, snowy, dry, wet, cold, hot, etc.

Table 4 Organization of POT Tagger

Sl.No.	Category		Label	No. of Newly added Tags	Example
	Top Level	Sub level			
1	Location	1	L	400	ലണ്ടൻ, മണാലി
2	Travel Type	1	TT	68	ട്രെയിൻ, റോഡ്, ബൈക്ക്
3	Travel Mode	1	TM	46	സോളോ, കുടുംബം, കൂട്ടുകാർ
4	Climate	1	LC	6	തണുപ്പ്, മഞ്ഞ്, സമ്മർ
5	Location Type	1	LT	15	ഹൈറേഞ്ച്, തീർത്ഥാടനം

The main tags used in POT Tagger are,

Location (L): London, Manali - ലണ്ടൻ, മണാലി

Travel Type (TT): Train, Road, Bike - ട്രെയിൻ, റോഡ്, ബൈക്ക്

Travel Mode (TM): Solo, Family, Friends - സോളോ, കുടുംബം, കൂട്ടുകാർ

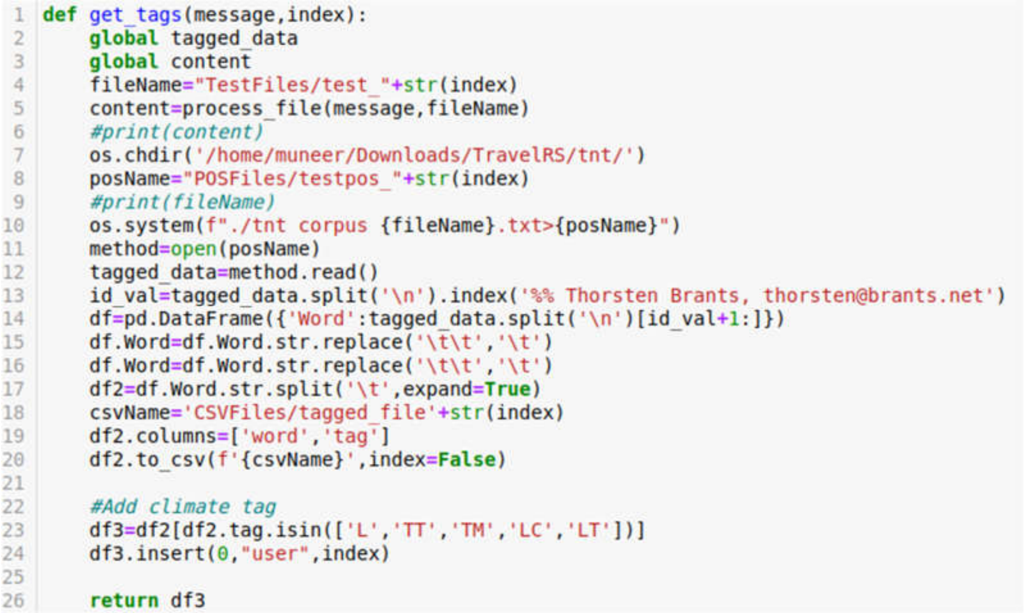
Location Climate (LC): Summer, Winter, Rainy തണുപ്പ്, മഞ്ഞ്, സമ്മർ

Location Type (LT): Historical, Pilgrimage, Natural, Adventurous - ഹൈറേഞ്ച്, തീർത്ഥാടനം, ചരിത്രം

A total of 68 new travel types were added to this POT Tagger, 46 distinct travel modes, 6 different location climates, and 15 location types that denote various geographical and destination categories [107]. Furthermore, the tagger was enriched with the addition of 400 specific locations, a significant enhancement that allows for precise annotation of the places described in the travelogues. With these additions, the POT tagger became a highly specialized tool, a total 541,958 tokens used in this corpus.

TnT (Trigrams'n'Tags) and TnT-Para are packages often used in Natural Language Processing (NLP), specifically in the area of part-of-speech tagging. TnT is a statistical part-of-speech tagger that utilizes trigram models to perform the tagging. It's known for its accuracy and efficiency in dealing with various languages. TnT employs a second-order Markov model to predict the part-of-speech tags, considering the probability of a tag given the previous two tags. This is built upon a frequency analysis of the trigrams in the training corpus. TnT-Para extends the functionality of the TnT tagger to handle paragraph and sentence-level structure in addition to word-level tagging. While standard TnT focuses primarily on individual words and their immediate context, TnT-Para incorporates information about the broader syntactic and semantic structure of the paragraph.

This enhanced contextual understanding enables TnT-Para to produce more accurate and nuanced tagging, particularly in complex texts where local word-level clues may be insufficient to determine the correct tag. TnT-Para's ability to recognize and incorporate higher-level structure makes it valuable in tasks requiring a deep understanding of text, such as information extraction, summarization, and translation. Figure 9 explains the tagging of tokens.

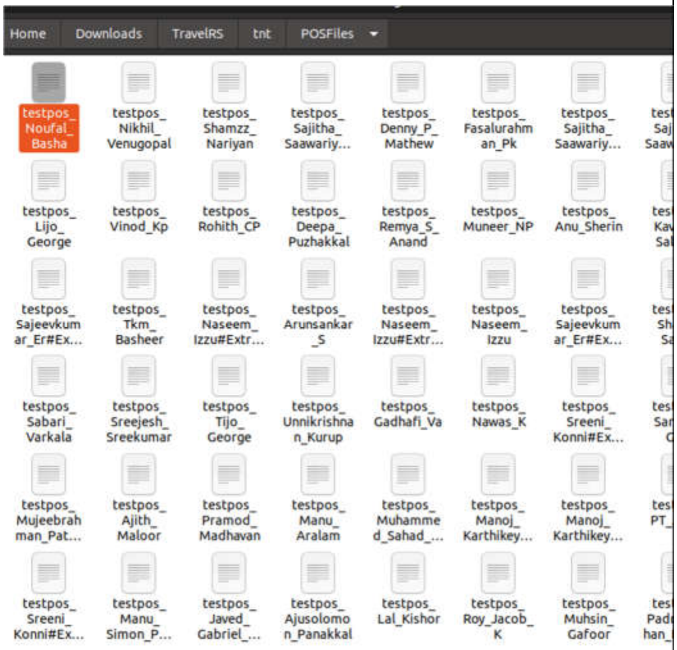
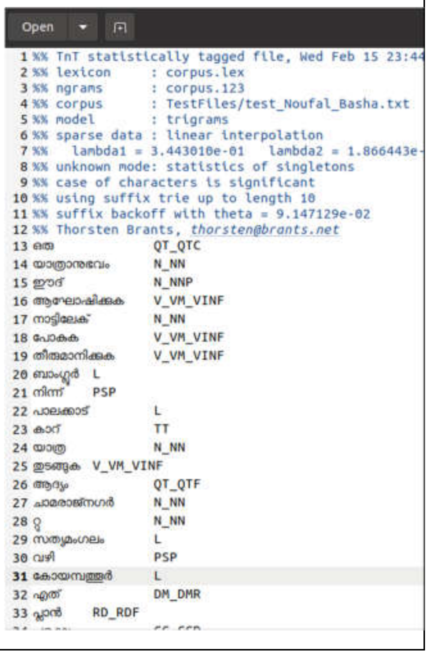


```
def get_tags(message,index):
    global tagged_data
    global content
    fileName="TestFiles/test_"+str(index)
    content=process_file(message,fileName)
    #print(content)
    os.chdir('/home/muneer/Downloads/TravelRS/tnt/')
    posName="POSFiles/testpos_"+str(index)
    #print(fileName)
    os.system(f"./tnt corpus {fileName}.txt>{posName}")
    method=open(posName)
    tagged_data=method.read()
    id_val=tagged_data.split('\n').index('%% Thorsten Brants, thorsten@brants.net')
    df=pd.DataFrame({'Word':tagged_data.split('\n')[id_val+1:]})
    df.Word=df.Word.str.replace('\t\t','\t')
    df.Word=df.Word.str.replace('\t\t','\t')
    df2=df.Word.str.split('\t',expand=True)
    csvName='CSVFiles/tagged_file'+str(index)
    df2.columns=['word','tag']
    df2.to_csv(f'{csvName}',index=False)

    #Add climate tag
    df3=df2[df2.tag.isin(['L','TT','TM','LC','LT'])]
    df3.insert(0,"user",index)

    return df3
```

Figure 9 Tagging of processed tokens

All tokens annotated by this process are kept in a separate folder with the same file name and a sample of such file opened mode is given Table 5.

Table 5 Tagged folder and its files. One among them opened.

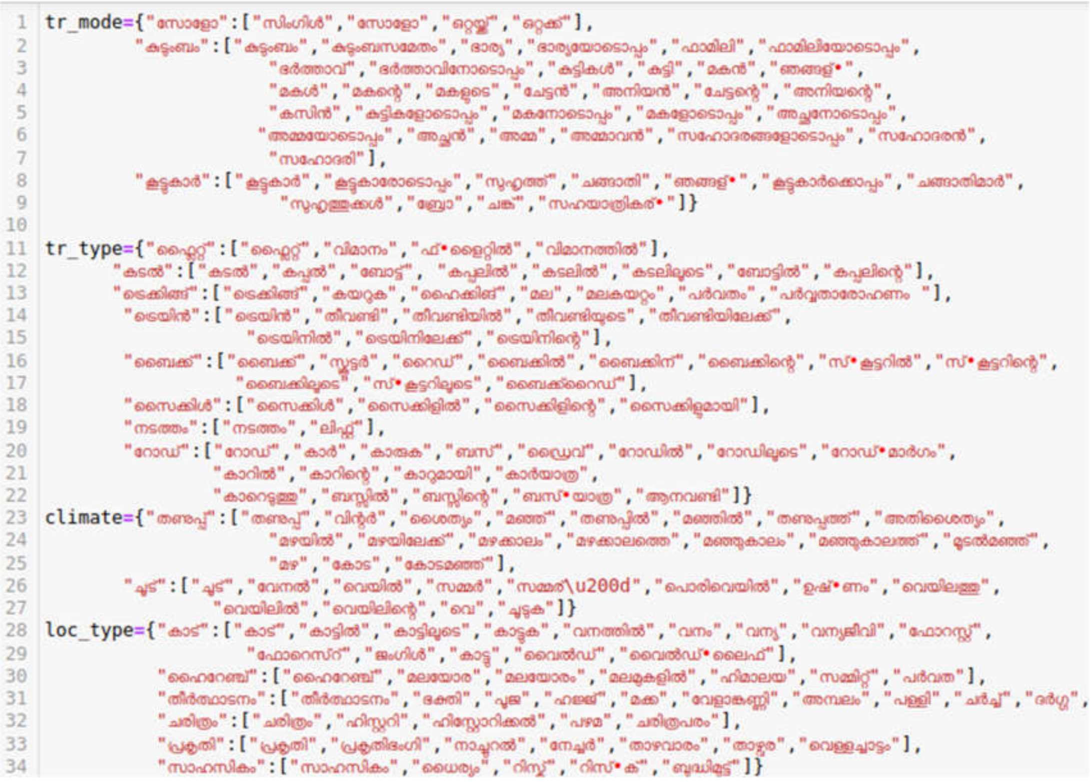
POT Tagged files	An opened POT Tagged file (testpos_Noufal_Basha)
	

Home
Downloads
TravelRS
tnt
POSFiles
testpos_ Noufal_ Basha
testpos_ Nikhil_ Venugopal
testpos_ Shamzz_ Nariyan
testpos_ Sajitha_ Saawariy...
testpos_ Denny_P_ Mathew
testpos_ Fasalurahm an_Pk
testpos_ Sajitha_ Saawariy...
testpos_ Lijo_ George
testpos_ Vinod_Kp
testpos_ Rohith_CP
testpos_ Deepa_ Puzhakkal
testpos_ Remya_S_ Anand
testpos_ Muneer_NP
testpos_ Anu_Sherin
testpos_ Sajeevkum ar_Er#Ex...
testpos_ Tkm_ Basheer
testpos_ Naseem_ Izzu#Extr...
testpos_ Arunsankar _S
testpos_ Naseem_ Izzu#Extr...
testpos_ Naseem_ Izzu
testpos_ Sajeevkum ar_Er#Ex...
testpos_ Sabari_ Varkala
testpos_ Sreejesh_ Sreekumar
testpos_ Tijo_ George
testpos_ Unnikrishna n_Kurup
testpos_ Gadhafi_Va
testpos_ Nawas_K
testpos_ Sreeni_ Konni#Ex...
testpos_ Mujeebrah man_Pat...
testpos_ Ajith_ Maloor
testpos_ Pramod_ Madhavan
testpos_ Manu_ Aralam
testpos_ Muhamme d_Sahad_...
testpos_ Manoj_ Karthikey...
testpos_ Manoj_ Karthikey...
testpos_ Sreeni_ Konni#Ex...
testpos_ Manu_ Simon_P...
testpos_ Javed_ Gabriel_...
testpos_ Ajusolomo n_Panakkal
testpos_ Lal_Kishor
testpos_ Roy_Jacob_ K
testpos_ Muhsin_ Gafoor
Open

```
%% TnT statistically tagged file, Wed Feb 15 23:44
%% lexicon     : corpus.lex
%% ngrams      : corpus.123
%% corpus      : TestFiles/test_Noufal_Basha.txt
%% model       : trigrams
%% sparse data : linear interpolation
%%   lambda1 = 3.443010e-01   lambda2 = 1.866443e-
%% unknown mode: statistics of singletons
%% case of characters is significant
%% using suffix trie up to length 10
%% suffix backoff with theta = 9.147129e-02
%% Thorsten Brants, thorsten@brants.net
ഒരു             QT_QTC
യാത്രാനുഭവം       N_NN
ഊര്              N_NNP
ആഘോഷിക്കുക       V_VM_VINF
നാട്ടിലേക്ക്         N_NN
പോകുക           V_VM_VINF
തീരുമാനിക്കുക      V_VM_VINF
ബാംഗ്ലൂർ   L
നിന്ന്      PSP
പാലക്കാട്          L
കാറ്              TT
യാത്ര             N_NN
ഇടങ്ങുക  V_VM_VINF
ആദ്യം             QT_QTF
ചാമരാജ്നഗർ       N_NN
ഉ                N_NN
സത്യമംഗലം         L
വഴി              PSP
കോയമ്പത്തൂർ       L
എത്ത്              DM_DMR
പ്ലാൻ      RD_RDF
```

5.4 Mapping from Lookup Dictionary

The look-up dictionary utilized in this study has a specific structure designed to facilitate the annotation process within the travel domain in Malayalam. Within the dictionary, index key items are established, to which all related tokens can be mapped. These key items serve as categories or labels that summarize a group of related concepts. For instance, under the category of Travel Mode (TM), all tokens relating to familial relationships such as 'സഹോദരൻ', 'അമ്മാവൻ', 'അച്ഛൻ', 'മുത്തശ്ശി', 'കുട്ടികൾ' equivalent to 'brother', 'uncle', 'father', 'grandmother', 'kids' in English etc., are mapped to the 'കുടുംബം', ie 'family' key. This method of grouping allows for the abstraction of specific details into broader concepts that can be easily managed and analysed. The structure of lookup dictionary developed for this travel domain is as given in figure 10.



```
tr_mode={"സോളോ":["സിംഗിൾ","സോളോ","ഒറ്റയ്ക്ക്","ഒറ്റക്ക്"],
          "കുടുംബം":["കുടുംബം","കുടുംബസമേതം","ഭാര്യ","ഭാര്യയോടൊപ്പം","ഫാമിലി","ഫാമിലിയോടൊപ്പം",
                     "ഭർത്താവ്","ഭർത്താവിനോടൊപ്പം","കുട്ടികൾ","കുട്ടി","മകൻ","ഞങ്ങള്‌",
                     "മകൾ","മകന്റെ","മക്കളുടെ","ചേട്ടൻ","അനിയൻ","ചേട്ടന്റെ","അനിയന്റെ",
                     "കസിൻ","കുട്ടികളോടൊപ്പം","മകനോടൊപ്പം","മകളോടൊപ്പം","അച്ഛനോടൊപ്പം",
                    "അമ്മയോടൊപ്പം","അച്ഛൻ","അമ്മ","അമ്മാവൻ","സഹോദരങ്ങളോടൊപ്പം","സഹോദരൻ",
                     "സഹോദരി"],
          "കൂട്ടുകാർ":["കൂട്ടുകാർ","കൂട്ടുകാരോടൊപ്പം","സുഹൃത്ത്","ചങ്ങാതി","ഞങ്ങള്‌","കൂട്ടുകാർക്കൊപ്പം","ചങ്ങാതിമാർ",
                      "സുഹൃത്തുക്കൾ","ബ്രോ","ചങ്ക്","സഹയാത്രികര്‌"]}

tr_type={"ഫ്ലൈറ്റ്":["ഫ്ലൈറ്റ്","വിമാനം","ഫ്‌ലൈറ്റിൽ","വിമാനത്തിൽ"],
       "കടൽ":["കടൽ","കപ്പൽ","ബോട്ട്", "കപ്പലിൽ","കടലിൽ","കടലിലൂടെ","ബോട്ടിൽ","കപ്പലിന്റെ"],
       "ട്രെക്കിങ്ങ്":["ട്രെക്കിങ്ങ്","കയറുക","ഹൈക്കിങ്ങ്","മല","മലകയറ്റം","പർവതം","പർവ്വതാരോഹണം "],
        "ട്രെയിൻ":["ട്രെയിൻ","തീവണ്ടി","തീവണ്ടിയിൽ","തീവണ്ടിയുടെ","തീവണ്ടിയിലേക്ക്",
                   "ട്രെയിനിൽ","ട്രെയിനിലേക്ക്","ട്രെയിനിന്റെ"],
        "ബൈക്ക്":["ബൈക്ക്","സ്കൂട്ടർ","റൈഡ്","ബൈക്കിൽ","ബൈക്കിന്","ബൈക്കിന്റെ","സ്‌കൂട്ടറിൽ","സ്‌കൂട്ടറിന്റെ",
                  "ബൈക്കിലൂടെ","സ്‌കൂട്ടറിലൂടെ","ബൈക്ക്റൈഡ്"],
        "സൈക്കിൾ":["സൈക്കിൾ","സൈക്കിളിൽ","സൈക്കിളിന്റെ","സൈക്കിളുമായി"],
        "നടത്തം":["നടത്തം","ലിഫ്റ്റ്"],
        "റോഡ്":["റോഡ്","കാർ","കാറുക","ബസ്","ഡ്രൈവ്","റോഡിൽ","റോഡിലൂടെ","റോഡ്‌മാർഗം",
                "കാറിൽ","കാറിന്റെ","കാറുമായി","കാർയാത്ര",
                "കാറെടുത്തു","ബസ്സിൽ","ബസ്സിന്റെ","ബസ്‌യാത്ര","ആനവണ്ടി"]}
climate={"തണുപ്പ്":["തണുപ്പ്","വിന്റർ","ശൈത്യം","മഞ്ഞ്","തണുപ്പിൽ","മഞ്ഞിൽ","തണുപ്പത്ത്","അതിശൈത്യം",
                     "മഴയിൽ","മഴയിലേക്ക്","മഴക്കാലം","മഴക്കാലത്തെ","മഞ്ഞുകാലം","മഞ്ഞുകാലത്ത്","മൂടൽമഞ്ഞ്",
                     "മഴ","കോട","കോടമഞ്ഞ്"],
        "ചൂട്":["ചൂട്","വേനൽ","വെയിൽ","സമ്മർ","സമ്മര്\u200d","പൊരിവെയിൽ","ഉഷ്‌ണം","വെയിലത്തു",
                "വെയിലിൽ","വെയിലിന്റെ","വെ","ചൂടുക"]}
loc_type={"കാട്":["കാട്","കാട്ടിൽ","കാട്ടിലൂടെ","കാട്ടുക","വനത്തിൽ","വനം","വന്യ","വന്യജീവി","ഫോറസ്റ്റ്",
                  "ഫോറെസ്റ്റ്","ജംഗിൾ","കാട്ടു","വൈൽഡ്","വൈൽഡ്‌ലൈഫ്"],
           "ഹൈറേഞ്ച്":["ഹൈറേഞ്ച്","മലയോര","മലയോരം","മലമുകളിൽ","ഹിമാലയ","സമ്മിറ്റ്","പർവത"],
           "തീർത്ഥാടനം":["തീർത്ഥാടനം","ഭക്തി","പൂജ","ഹജ്ജ്","മക്ക","വേളാങ്കണ്ണി","അമ്പലം","പള്ളി","ചർച്ച്","ദർഗ്ഗ",
           "ചരിത്രം":["ചരിത്രം","ഹിസ്റ്ററി","ഹിസ്റ്റോറിക്കൽ","പഴമ","ചരിത്രപരം"],
           "പ്രകൃതി":["പ്രകൃതി","പ്രകൃതിഭംഗി","നാച്ചുറൽ","നേച്ചർ","താഴ്‌വാരം","താഴ്‌വര","വെള്ളച്ചാട്ടം"],
           "സാഹസികം":["സാഹസികം","ധൈര്യം","റിസ്ക്","റിസ്‌ക്","ബുദ്ധിമുട്ട്"]}
```

Figure 10 Look up dictionary structure.

After extracting essential features like travel type, travel mode, location, and climate from each travelogue, a CSV/spread sheet file was created for each traveler, with separate entries for individuals with multiple travelogues. This method allowed the conversion of a set of unstructured data into a structured dataset, encapsulating the travel preferences of each traveler.

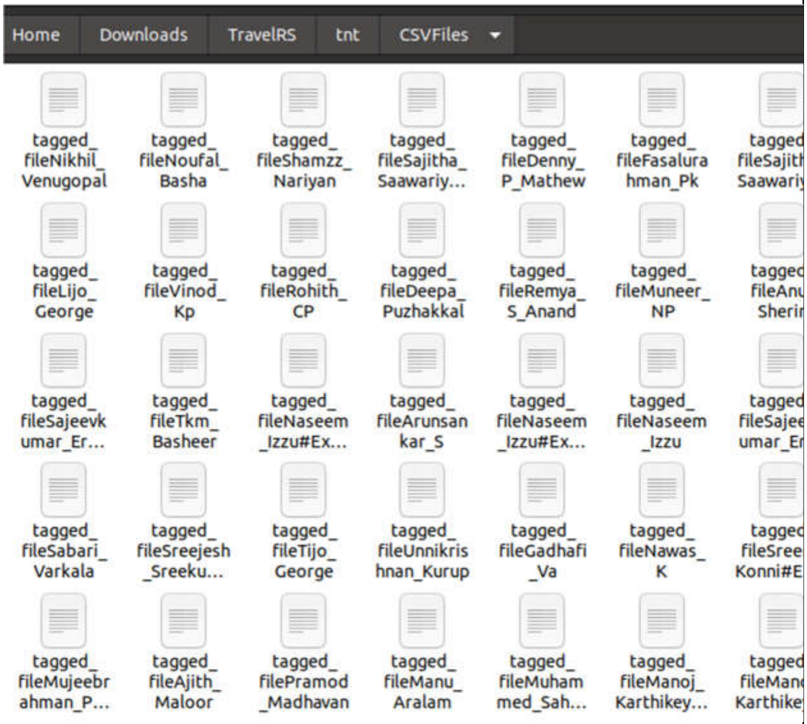
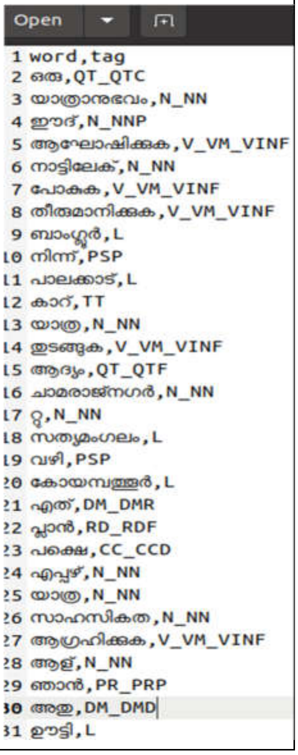
Similar key items are established for other categories such as Travel Type (TT), Location Type (LT), and Location Climate (LC). Each key item is numerically numbered to ensure a standardized reference system within the dictionary. The items belong to key, numerical values belong to index field and corresponding remarks are as shown in Table 6.

Table 6 Key-index values of Lookup Dictionary

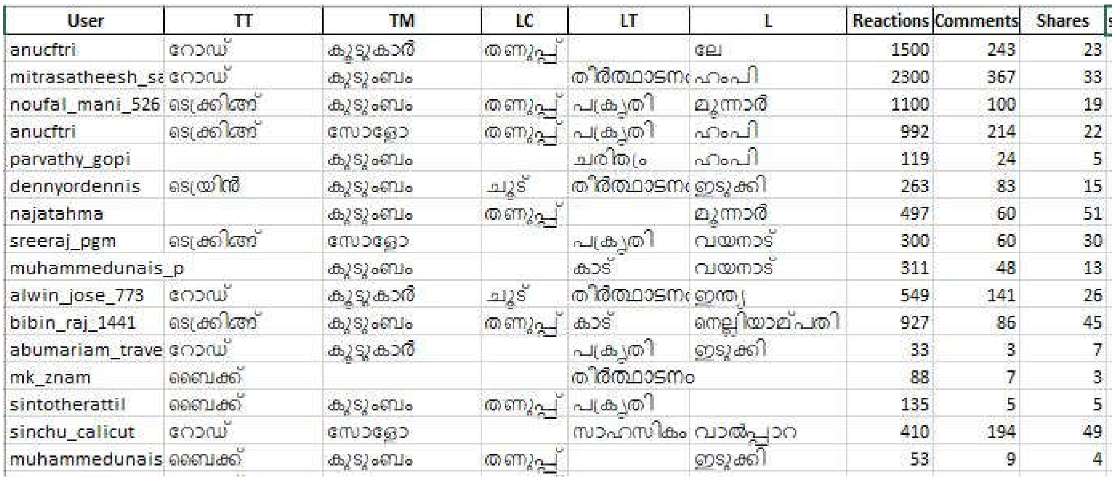
Key	Index	Remarks
Travel Mode TM		
സോളോ	0	To whom with Travel
കുടുംബം	1	
കൂട്ടുകാർ	2	
Travel Type TT		
ഫ്ലൈറ്റ്	0	Vehicle/ Method used
കടൽ	1	
ട്രെക്കിംഗ്	2	
ട്രെയിൻ	3	
ബൈക്ക്	4	
സൈക്കിൾ	5	
നടത്തം	6	
റോഡ്	7	
Location Climate LC		
തണുപ്പ്	0	Climate information
ചൂട്	1	
Location Type LT		
കാട്	0	Type/ Activity of destination
ഹൈറെ യ്ഞ്ച്	1	
തീർഥാടനം	2	
ചരിത്രം	3	
പ്രകൃതി	4	
സാഹസികം	5	

Based on the annotation from POT Tagger and cross matching with look-up dictionary, all annotated tokens are stored in a CSV file that contains each token and its tag separated with a comma. The structure of CSV file is as given in table 7.

Table 7 CSV file generated tokens using look up dictionary.

CSV folder with 13458 POT Tagged files	An opened file
 Home Downloads TravelRS tnt CSVFiles tagged_fileNikhil_Venugopal tagged_fileNoufal_Basha tagged_fileShamzz_Nariyan tagged_fileSajitha_Saawariy... tagged_fileDenny_P_Mathew tagged_fileFasalurahman_Pk tagged fileSajit Saawari tagged_fileLijo_George tagged_fileVinod_Kp tagged_fileRohith_CP tagged_fileDeepa_Puzhakkal tagged_fileRemya_S_Anand tagged_fileMuneer_NP taggec fileAnu Sherir tagged_fileSajeevkumar_Er... tagged_fileTkm_Basheer tagged_fileNaseem_Izzu#Ex... tagged_fileArunsankar_S tagged_fileNaseem_Izzu#Ex... tagged_fileNaseem_Izzu tagged fileSajee umar_Er tagged_fileSabari_Varkala tagged_fileSreejesh_Sreeku... tagged_fileTijo_George tagged_fileUnnikrishnan_Kurup tagged_fileGadhafi_Va tagged_fileNawas_K tagged fileSree Konni#E tagged_fileMujeebrahman_P... tagged_fileAjith_Maloor tagged_filePramod_Madhavan tagged_fileManu_Aralam tagged_fileMuhammed_Sah... tagged_fileManoj_Karthikey... tagged fileMan Karthike	 Open word,tag ഒരു,QT_QTC യാത്രാനുഭവം,N_NN ഈദ്,N_NNP ആഘോഷിക്കുക,V_VM_VINF നാട്ടിലേക്ക്,N_NN പോകുക,V_VM_VINF തീരുമാനിക്കുക,V_VM_VINF ബാംഗ്ലൂർ,L നിന്ന്,PSP പാലക്കാട്,L കാറ്,TT യാത്ര,N_NN തുടങ്ങുക,V_VM_VINF ആദ്യം,QT_QTF ചാമരാജ്നഗർ,N_NN റ്റ,N_NN സത്യമംഗലം,L വഴി,PSP കോയമ്പത്തൂർ,L എത്ത്,DM_DMR പ്ലാൻ,RD_RDF പക്ഷെ,CC_CCD എപ്പഴ്,N_NN യാത്ര,N_NN സാഹസികത,N_NN ആഗ്രഹിക്കുക,V_VM_VINF ആള്,N_NN ഞാൻ,PR_PRP അതു,DM_DMD ഊട്ടി,L

Once the tokens are mapped to their respective key items, the frequency of each item and its tag is counted. This counting process plays an essential role in identifying the most prevalent or significant key item within a particular travelogue. By recognizing these recurring tags and elements, the study gains insights into the patterns and preferences reflected in the travelogues. For each category, say L, LT, TT, TM and LC, the token which have highest frequency will be treated as most preferred item in that category. For example, consider TT tag, if there are 5 entries for Road, 2 entries for bike, and 1 entry for train, then the travel type is considered as Road as it has a higher frequency. This is added to the TT columns of that travelogue in the spreadsheet. Each field is filled with the same scenario to form a travel preference of the users. The spreadsheet is then treated as the preliminary dataset for further processing. The sample representation is given in Figure 11.

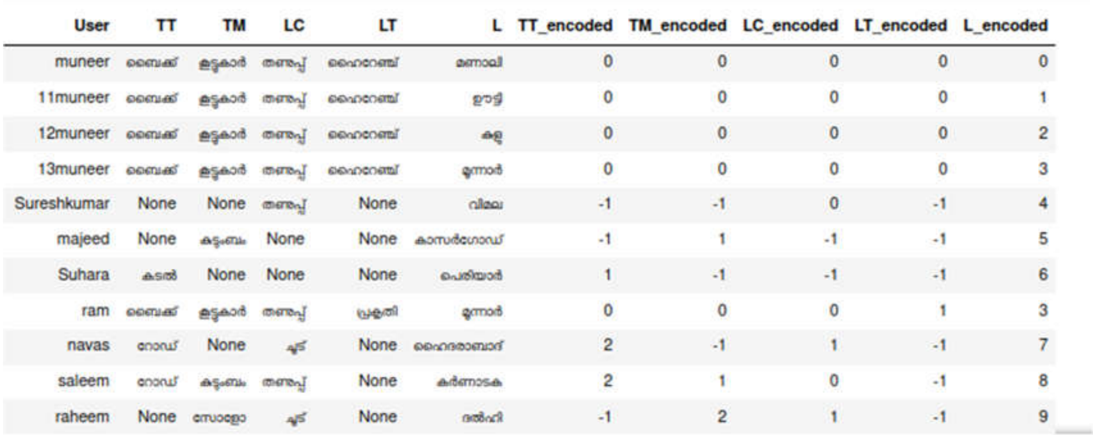


User	TT	TM	LC	LT	L	Reactions	Comments	Shares
anucftri	റോഡ്	കൂട്ടുകാർ	തണുപ്പ്		ലേ	1500	243	23
mitrasatheesh_sa	റോഡ്	കുടുംബം		തീർത്ഥാടന	ഹംപി	2300	367	33
noufal_mani_526	ട്രെക്കിങ്	കുടുംബം	തണുപ്പ്	പ്രകൃതി	മൂന്നാർ	1100	100	19
anucftri	ട്രെക്കിങ്	സോളോ	തണുപ്പ്	പ്രകൃതി	ഹംപി	992	214	22
parvathy_gopi		കുടുംബം		ചരിത്രം	ഹംപി	119	24	5
dennyordennis	ട്രെയിൻ	കുടുംബം	ചൂട്	തീർത്ഥാടന	ഇടുക്കി	263	83	15
najatahma		കുടുംബം	തണുപ്പ്		മൂന്നാർ	497	60	51
sreeraj_pgm	ട്രെക്കിങ്	സോളോ		പ്രകൃതി	വയനാട്	300	60	30
muhammedunais_p		കുടുംബം		കാട്	വയനാട്	311	48	13
alwin_jose_773	റോഡ്	കൂട്ടുകാർ	ചൂട്	തീർത്ഥാടന	ഇന്ത്യ	549	141	26
bibin_raj_1441	ട്രെക്കിങ്	കുടുംബം	തണുപ്പ്	കാട്	നെല്ലിയാമ്പതി	927	86	45
abumariam_trave	റോഡ്	കൂട്ടുകാർ		പ്രകൃതി	ഇടുക്കി	33	3	7
mk_znam	ബൈക്ക്			തീർത്ഥാടനം		88	7	3
sintotherattil	ബൈക്ക്	കുടുംബം	തണുപ്പ്	പ്രകൃതി		135	5	5
sinchu_calicut	റോഡ്	സോളോ		സാഹസികം	വാൽപ്പാറ	410	194	49
muhammedunais	ബൈക്ക്	കുടുംബം	തണുപ്പ്		ഇടുക്കി	53	9	4

Figure 11 spreadsheet generated from these CSV files.

5.5 Feature File Creation and Encoding

With the help of SciKit-Learn and OneHotEncoder modules, one hot encoding is applied to the processed Spreadsheet containing the extracted travel features, transforming it into a structured dataset. This encoding technique converts categorical data, such as travel type, travel mode, location, and climate, into a binary matrix. During the one-hot encoding process, the index provided by the look-up dictionary is used, and the corresponding item numbers begin from 0. This means that each unique category within a feature is assigned a unique integer starting from 0 and every feature column is appended with a suffix '_encoded'. The largest values come in the column of Location (L_encoded) as there can be 400 destinations mentioned in POT Tagger. By using one hot encoding, the dataset becomes suitable for machine learning algorithms, turning the travelogues' rich, descriptive information into a format that can be readily analysed and modeled. One hot encoded sample is given in Figure 12.



User	TT	TM	LC	LT	L	TT_encoded	TM_encoded	LC_encoded	LT_encoded	L_encoded
muneer	ബൈക്ക്	കൂട്ടുകാർ	തണുപ്പ്	ഹൈറേഞ്ച്	മണാലി	0	0	0	0	0
11muneer	ബൈക്ക്	കൂട്ടുകാർ	തണുപ്പ്	ഹൈറേഞ്ച്	ഊട്ടി	0	0	0	0	1
12muneer	ബൈക്ക്	കൂട്ടുകാർ	തണുപ്പ്	ഹൈറേഞ്ച്	കുളു	0	0	0	0	2
13muneer	ബൈക്ക്	കൂട്ടുകാർ	തണുപ്പ്	ഹൈറേഞ്ച്	മൂന്നാർ	0	0	0	0	3
Sureshkumar	None	None	തണുപ്പ്	None	ഷിംല	-1	-1	0	-1	4
majeed	None	കുടുംബം	None	None	കാസർഗോഡ്	-1	1	-1	-1	5
Suhara	കടൽ	None	None	None	പെരിയാർ	1	-1	-1	-1	6
ram	ബൈക്ക്	കൂട്ടുകാർ	തണുപ്പ്	പ്രകൃതി	മൂന്നാർ	0	0	0	1	3
navas	റോഡ്	None	ചൂട്	None	ഹൈദരാബാദ്	2	-1	1	-1	7
saleem	റോഡ്	കുടുംബം	തണുപ്പ്	None	കർണാടക	2	1	0	-1	8
raheem	None	സോളോ	ചൂട്	None	ഡൽഹി	-1	2	1	-1	9

Figure 12 One hot encoded version of the dataset.

5.6 Conclusion

This chapter meticulously detailed the innovative approach to building a personalized travel recommender system for Malayalam language travelogues, filling a significant gap in the field where no benchmark dataset was previously available. Through a comprehensive process of extraction of travelogues from social media, particularly the 'sanchari' group in Kerala, and other independent travel blogs, the research led to the formation of a robust and unique dataset. Preprocessing stages were executed with precision, addressing the highly inflectional and morphologically rich nature of the Malayalam language. Techniques like sentence tokenization, word tokenization, removal of impurities, stemming, and lemmatization were employed to extract significant features, followed by the creation of a novel Part of Travelogue Tagger (POT) to annotate the processed tokens. The introduction of additional tags and the utilization of a look-up dictionary have emerged as salient features of this study. These include 68 new travel types, 46 travel modes, 6 location climates, 15 location types, and 400 locations, adding up to a corpus of 541958 tokens. The creation of the look-up dictionary, which mapped related tokens to key items like Travel mode and Travel type, played a crucial role in identifying the most frequent key items in each travelogue. This facilitated the transformation of the unstructured and noisy travelogue data into a highly structured and significant dataset.

The chapter also explored into advanced techniques like one hot encoding, based on indexing from the look-up dictionary, enabling the construction of a structured dataset ready for machine learning algorithms. Utilizing NLP packages like TnT and TnT-Para added a layer of complexity and efficiency to the part-of-speech tagging processes. In summary, the methodologies and strategies outlined in this chapter have not only contributed a novel dataset but also provided a framework for future research in personalized travel recommendations in regional languages, demonstrating the potential to expand these techniques to other linguistic and cultural contexts.

6 Rule-based Cosine Similarity Recommender System

6.1 Introduction

The rise of personalized travel recommendation systems has redefined how travelers plan and embark on journeys, enhancing their experiences by catering to individual preferences and desires. However, the Malayalam language, spoken predominantly in the Indian state of Kerala, has been underrepresented in this domain, owing to the lack of a benchmark dataset and prior research in this area. The absence of structured information on travelogues and experiences in Malayalam presents a unique challenge and opportunity for innovation.

To address this gap, this chapter focuses on an ambitious effort to extract and process extensive travelogues from travel blogs and the largest travel group in Kerala named 'Sanchari,' found on social media platforms such as Facebook. A total of 13458 unstructured travelogues have been collected and undergone rigorous preprocessing, including sentence tokenization, removal of punctuation, code-mixing, stop words, stemming, and lemmatization.

Central to the research is the development of Travel DNA and Location DNA, which are crafted through extensive preprocessing and analysis of Malayalam travelogues. Travel DNA encapsulates a traveler's unique travel attributes, including type, mode, location, and climate preferences, while Location DNA provides consolidated information about each destination. These DNA constructs are further subjected to advanced mathematical techniques, including cosine similarity and collaborative filtering. To enhance the accuracy and relevance of the recommendations, cosine similarity plays a pivotal role. In a multi-dimensional space where each dimension corresponds to a word in the document, cosine similarity measures the cosine of the angle between two vectors, effectively capturing the orientation of the documents. Unlike Euclidean distance, it focuses on the angle, making it robust against the magnitude variations. This method allows the model to discern subtle similarities between different travel experiences and aligns them based on travelers' preferences and tastes.

Collaborative filtering, a cornerstone of the methodology, is employed to make automatic suggestions based on travelers' shared interests. The system considers two primary models: User-based Collaborative Filtering, measuring resemblance between target travelers and other users, and Item-based (Location-based) Collaborative Filtering, gauging connections between locations that travelers interact with. This two-pronged approach forms a powerful tool to understand and predict preferences, ensuring the recommender system's efficacy and personalized nature.

The experimental results are derived from methodically testing the algorithm's ability to suggest the most suitable destinations to users based on their unique Travel DNA, applying both cosine similarity and collaborative filtering techniques. By prompting the users to enter specific travel preferences, the system was able to provide primary and secondary recommendation lists, which were then rigorously analysed. The testing phase revealed an impressive 80% and 75% accuracy rate for primary recommendations and secondary recommendations respectively, validating the overall model's precision and reliability. This part of the chapter not only illustrates the model's performance but also emphasizes the robustness of the methodologies employed, marking a significant stride in personalized travel recommendation within the Malayalam language context. Figure 13 depicts the entire wireframe of RS.

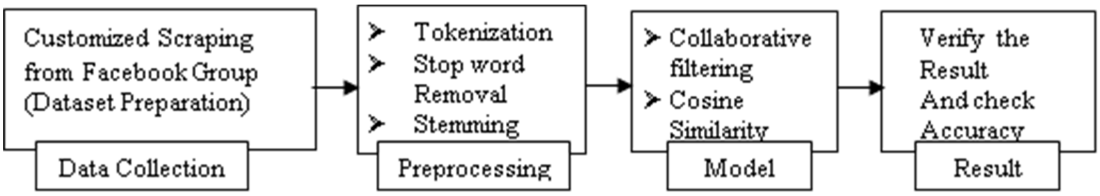


Figure 13 steps involved in Rule-based RS

6.2 Dataset Preparation and Creation of Encoded Feature File

POT tagging involves identifying and annotating specific features within travelogues that are significant to understanding the preferences and behaviours of travellers. This process can include tagging attributes such as Travel Type (TT), Travel Mode (TM), Location (L), Location Climate (LC), Location Type (LT), and User (U). By assigning these tags to the relevant portions of the text, the travelogues are structured into a format that allows for systematic analysis. For example, in a sentence like "I went in train to the beautiful beaches of Goa during summer," “വേനൽക്കാലത്ത് ഞാൻ ഗോവയിലെ മനോഹരമായ ബീച്ചുകളിലേക്ക് തീവണ്ടിയിൽ പോയി” POT tagging would recognize and tag the Travel Mode (train) (തീവണ്ടി), Location (Goa) (ഗോവ), Location Type (beaches)(ബീച്ച്), and Location Climate (summer) (വേനൽ). This makes the data more interpretable for the algorithms that will be used to process it later.

One-hot encoding is a process used to convert categorical data variables into a form that could be provided to machine learning algorithms. Since features like Travel Type, Travel Mode, Location, etc., are categorical, they need to be transformed into a numerical format. In one-hot encoding, each unique category within a feature is represented by a binary vector. For example, if there are three travel modes (train, bus, plane), each mode will be represented by a separate binary vector like [1, 0, 0], [0, 1, 0], [0, 0, 1. Together, POT tagging and one-hot encoding provide a robust methodology to process and interpret complex, unstructured travelogues. The POT tagging provides valuable insight into the underlying travel preferences and behaviours, while one-hot encoding transforms these insights into a format suitable for algorithmic processing. These techniques lay the foundation for the subsequent stages of analysis, including similarity calculations and recommendations. They structured dataset formation from tagged features given in Table 8.

Table 8 Structured dataset formation from tagged features.

User	Travel Type (TT)	Travel Mode (TM)	Location climate(LC)	Location (L)	Reactions	Comments	Shares
Anucftri	റോഡ്/ Road	സുഹൃത്ത്/ Friend	തണുപ്പ്/ Winter	ലേ / Leh	1500	243	23
Anucftri	ട്രെക്കിംഗ്/ Trekking		വേനൽ/ Summer	ഹംപി/ Hampi	992	214	22
Anucftri	തീവണ്ടി/ Train	ഭാര്യ/ Wife	ചൂട്/ Hot	ഇടുക്കി/ Idukki	263	83	15
Awin jose	ബസ്/ Bus	ബ്രോ/ Bro	ചൂട്/ Hot	ഡൽഹി/ Delhi	549	141	26
Awin jose	ബസ്/ Bus	സുഹൃത്ത്/ Friend		ഡൽഹി/ Delhi	48	23	2
Aslam	ട്രെയിൻ/ Train	സുഹൃത്ത്/ Friend		മൈസൂർ/ Mysore	108	24	2
Aslam	ബോട്ട്/ Boat	സുഹൃത്ത്/ Friend	മഴ/ Rain	ജോഗ്/ Jog	45	6	9
Joy Cheray	ജീപ്പ്/ Jeep	ചേട്ടൻ /Brother	വേനൽ/ Summer	വാഗമൺ / Vagamon	27	1	1
Joy Cheray	ട്രെക്കിംഗ്/ Trekking	സോളോ/ Solo		ഇടുക്കി/ Idukki	17	1	1

6.3 Travel DNA Formation

Travel DNA represents a unique fingerprint or identity of a traveller, derived from their past travel experiences, behaviours, and preferences. Just as genetic DNA encompasses the fundamental characteristics of an organism, Travel DNA encapsulates the essential travel traits of an individual. It is a complex but precise construction of the various dimensions of travel, including user, location, travel mode, travel type, location climate, and type of location.

6.3.1 Essential Components of Travel DNA

The essential components of Travel DNA include the following:

User (The unique identifier representing a specific traveller), Location (Destinations visited by the traveller). Travel Mode (The means of transportation utilized by the traveller, such as car, bus, bike, flight, etc). Travel Type (The nature of the travel, including solo, family, friends, etc). Location Climate (The weather conditions of the destinations, like sunny, rainy, snowy, dry, etc). Type of Location (The categorization of the destination, like beach, mountain, city, rural area, etc).

6.3.2 Construction of Travel DNA

The steps to construct the Travel DNA are as listed below.

1. Extraction of Essential Features: From the unstructured, lengthy, and noisy travel reviews, a structured dataset containing six essential keywords is extracted. This dataset represents the travel behaviour of individual travelers.

2. Combining Multiple Travel Experiences: Since a traveller may visit multiple places with varying combinations of features, all these diverse experiences need to be consolidated. The various travel attributes are combined to understand and extract the traveler's overall pattern and preferences.

3. Creation of Individual Travel DNA: By analysing and comparing each traveler's history and pattern, an individual Travel DNA is created. This unique representation acts as a compact and insightful summary of the traveler's behaviours and preferences.

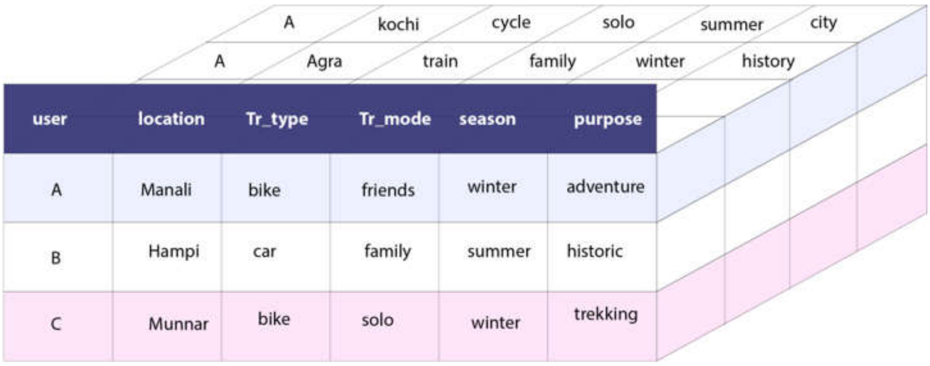
4. Dynamic Clustering of Travel DNAs: The individual Travel DNAs can be used to create user clusters. These clusters group together travelers with similar tastes and preferences, fostering a more targeted and relevant recommendation process. Moreover, the clusters can be dynamically created and updated, considering different situations and parameters.

Table 9 Travel DNA formation

Traveler	Location	Tr_Type	Tr_Mode	Climate	Location Type
A	മണാലി 1	ബൈക്ക് 1	സുഹൃത്ത് 3	തണുപ്പ് 1	സാഹസികം 1
	ഹംപി 3	തീവണ്ടി 4	ഒറ്റക്ക് 1	സമ്മർ 2	ചരിത്രം 2
	മൂന്നാർ 4	കാർ 2	കുടുംബം 2	തണുപ്പ് 1	പ്രകൃതി 5
B	ആഗ്ര 5	ട്രെയിൻ 4	കുടുംബം 2	തണുപ്പ് 1	ചരിത്രം 2
	ഗോവ 2	കാർ 2	ഒറ്റക്ക് 1	മഞ്ഞ്	വിനോദം 8
	കൊച്ചി 9	സൈക്കിൾ 6	സുഹൃത്ത് 3	തണുപ്പ് 1	തീർഥാടനം 3
	കോവളം 25	വിമാനം 5	സുഹൃത്ത് 3	സമ്മർ 2	വിനോദം 8

Table 9 above represents the Travel DNA of two travelers, A and B, based on the features extracted from their respective travelogues and annotated using the Part of Travelogue (POT) tagger. This DNA encapsulates the nature and tastes of the travelers, providing insight into their travel preferences.

For traveller A, three distinct trips are depicted. Each trip illustrates the location, travel type (TT), travel mode (TM), climate, and purpose of the visit. For example, A's trip to Manali was an adventurous winter journey undertaken by bike with friends. A's another trip was with family in car to enjoy the scenic natural beauty of winter in Munnar. Similarly, traveller B's Travel DNA captures four different trips. In the case of traveller B, this includes diverse destinations and experiences ranging from a family trip to ആഗ്ര by ട്രെയിൻ in തണുപ്പ് weather with historical interest to a solo car journey to ഗോവ during the മഞ്ഞ് season for enjoyment. This illustrates that every traveller may have varying preferences for travel modes and may visit multiple destinations, each with unique characteristics such as different seasons, travel modes, and travel types.



A	kochi	cycle	solo	summer	city
A	Agra	train	family	winter	history

user	location	Tr_type	Tr_mode	season	purpose
A	Manali	bike	friends	winter	adventure
B	Hampi	car	family	summer	historic
C	Munnar	bike	solo	winter	trekking

Figure 14 Travel DNA model of 3 users.

A 3D diagrammatic representation is given in figure 14 to demonstrate the orientation of travel preferences of each user with various combinations. Based on the similarity of values, the clusters are formed for further analysis and calculations.

6.3.3 Importance and Applications

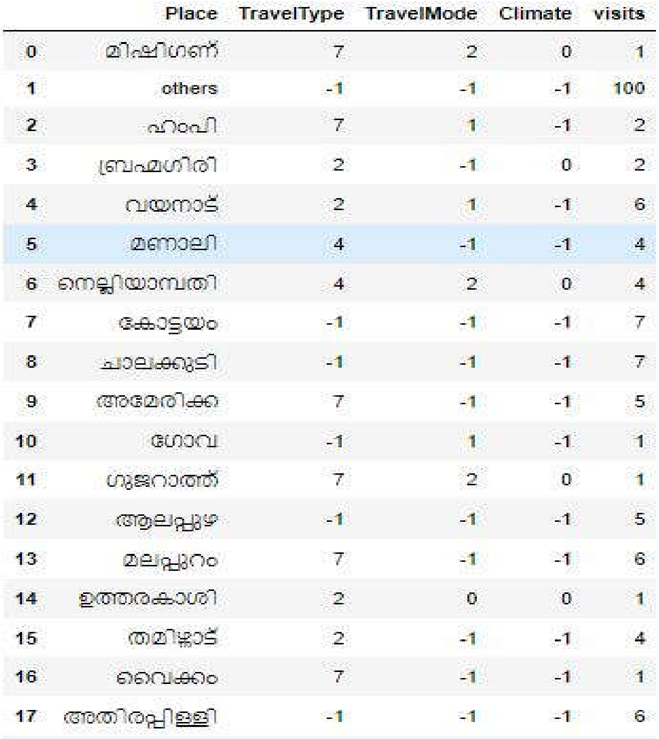
Travel DNA serves as a foundational component in understanding and catering to the unique requirements and tastes of each traveller. It enables the recommender system to personalize recommendations by aligning them with the core travel preferences of the user, Identify and leverage hidden patterns and relationships within the travel behaviours, and facilitate dynamic updates and adaptation to changing travel trends and individual preferences. The concept of Travel DNA transcends traditional recommendation methodologies by providing a more profound and nuanced understanding of travelers. By assimilating the multifaceted travel experiences into a cohesive structure, Travel DNA allows for more accurate and personalized recommendations. Its ability to dynamically adapt and create user clusters further enhances the precision of the system, marking a significant advancement in the field of travel recommender Systems. In essence, Travel DNA is not merely a data representation but a reflection of the traveler's soul, capturing their desires, choices, and affinities within the travel domain.

6.4 Location DNA Preparation

The creation of the Location DNA table is a crucial step in understanding travel preferences related to individual destinations [108]. While the Travel DNA focuses on encapsulating the characteristics of individual travelers, the Location DNA emphasizes the features of various destinations. The steps to create Location DNA is given as follows,

1. Selection of Features: The Location DNA table is created by considering the 5 features from the entire dataset that are the most relevant with respect to location. These features are the travel mode used to reach the destination, the type of travel people preferred, the climate in which people visited the location, and the total number of visits.
2. Frequency Calculation: For each destination, the frequency or count of each feature is recorded. This means that if a certain location is often visited by car, during summer, for recreational purposes, these characteristics will be reflected in the Location DNA.
3. Consolidation: This process creates consolidated information about every destination, providing a comprehensive view of how people interact with each location. It encapsulates how people reach specific places, their preferred travel styles, the favoured climate for visiting, and overall visitation rates.
4. Optimization for Prediction: One vital aspect of the Location DNA is that it helps in increasing the accuracy of prediction through clustering and collaborative filtering. By extending the length of the Location DNA (i.e., incorporating more features and details), the accuracy can be enhanced. Therefore, the most frequent feature is selected as the preferred feature for that location, providing a focused snapshot of what travelers usually prefer in that destination.

5. Practical Implications: The Location DNA provides insightful data that can be leveraged for creating personalized travel recommendations. By understanding the unique DNA of each location, the recommender system can suggest destinations that align with the user's preferences and behaviour. It also helps in clustering locations that have similar features, making it easier to provide relevant suggestions to travelers.



	Place	TravelType	TravelMode	Climate	visits
0	മിഷിഗണ്‍	7	2	0	1
1	others	-1	-1	-1	100
2	ഹംപി	7	1	-1	2
3	ബ്രഹ്മഗിരി	2	-1	0	2
4	വയനാട്	2	1	-1	6
5	മണാലി	4	-1	-1	4
6	നെല്ലിയാമ്പതി	4	2	0	4
7	കോട്ടയം	-1	-1	-1	7
8	ചാലക്കുടി	-1	-1	-1	7
9	അമേരിക്ക	7	-1	-1	5
10	ഗോവ	-1	1	-1	1
11	ഗുജറാത്ത്	7	2	0	1
12	ആലപ്പുഴ	-1	-1	-1	5
13	മലപ്പുറം	7	-1	-1	6
14	ഉത്തരകാശി	2	0	0	1
15	തമിഴ്നാട്	2	-1	-1	4
16	വൈക്കം	7	-1	-1	1
17	അതിരപ്പിള്ളി	-1	-1	-1	6

Figure 15 Location DNA with details

The Location DNA table as shown in Figure 15 serves as a valuable tool in the personalized travel recommendation process. It offers a nuanced understanding of different destinations and helps to make more accurate and tailored recommendations to individual travelers. By considering the most frequent features, it encapsulates the essence of each destination, which aids in providing more personalized and relevant suggestions.

6.5 Vector Representation and Traveler-Location Mapping

The vector representation is a vital step in transforming both Travel DNA and Location DNA into a format suitable for computational analysis. In this process, each travel-related feature such as travel type, mode, location, climate, and purpose is represented as a numerical value within a vector. These vectors capture the multidimensional nature of travel preferences and location characteristics, encapsulating the essential details in a mathematical form. By converting the textual and categorical data into a numerical vector space, similarities, and patterns can be identified using mathematical operations, like cosine similarity. The vectorization not only ensures computational efficiency but also creates a structured form that encapsulates the complexity of travel behaviours, thereby playing a pivotal role in the personalized travel recommender system.

6.6 Methodology

6.6.1 Cosine Similarity

Cosine similarity is a measure used to calculate the similarity between two vectors, often employed in text analysis to understand the similarity between two documents [109], or in this context, the Travel DNA and Location DNA. Mathematically, cosine similarity is expressed as the cosine of the angle between two vectors.

Cosine similarity is a metric used to measure how similar two vectors are, and it does so by calculating the cosine of the angle between them in a multi-dimensional space [110]. In this space, each dimension represents a term in the document, allowing the cosine similarity to focus on the orientation or the angle between the two vectors rather than their magnitude. This property distinguishes it from the Euclidean distance, which takes both magnitude and orientation into account. The advantage of using cosine similarity lies in its sensitivity to the orientation rather than the size of the vectors. For instance, if the term 'family' appears 50 times in one travelogue and 10 times in another, the Euclidean distance might suggest these two travelogues are far apart due to the difference in magnitude. However, the cosine similarity may find a smaller angle between them, indicating a higher similarity. This characteristic makes cosine similarity particularly useful for text analysis where the pattern or direction of the data is more significant than the absolute values, providing a more nuanced understanding of the similarity between documents [111]. The formula for cosine similarity is given by equation 3.

$$\cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum\limits_{i=1}^{n} A_i B_i}{\sqrt{\sum\limits_{i=1}^{n} A_i^2} \sqrt{\sum\limits_{i=1}^{n} B_i^2}} \qquad \text{Equation (3)}$$

Here A and B are the two vectors being compared, and n is the dimension of the vectors. The numerator calculates the dot product of the vectors, while the denominator normalizes the vectors by their magnitudes. The result is a value between -1 and 1, where 1 indicates complete similarity, -1 complete dissimilarity, and 0 no similarity at all. The representation of similarities between TT, TM, LC as shown in figure 16.

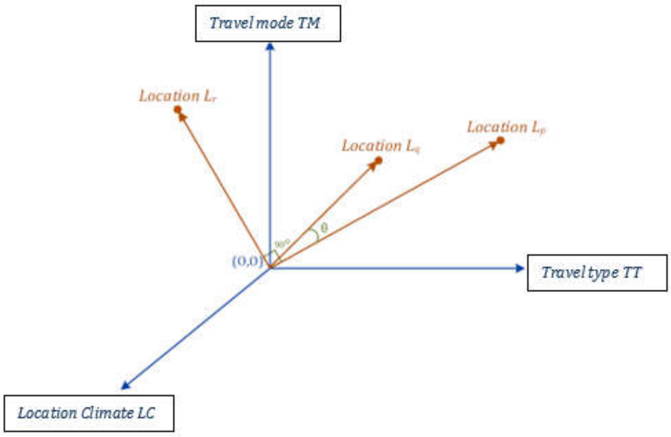


Figure 16 Cosine similarity measure between metrics

In the context of a personalized travel recommender system, cosine similarity is used to compare the Travel DNA of a user with the Location DNA of various destinations. By measuring the cosine of the angle between these vectors, the similarity or dissimilarity between a user's travel preferences and the features of different destinations can be quantified. This offers an insightful metric to gauge how closely a location matches a user's travel pattern, thereby enabling the system to recommend destinations that align well with the user's unique preferences. It plays a crucial role in filtering and ranking destinations based on the personalized alignment between travelers and locations, contributing significantly to the efficacy of the recommender system.

6.6.2 Collaborative Filtering

Collaborative filtering in the context of Travel DNA and Location DNA provides a robust approach to developing a travel recommender system, tailored to the specific preferences and interests of individual travelers [112].

6.6.2.1 User-Based Collaborative Filtering with Travel DNA:

By utilizing Travel DNA, which encapsulates a traveler's preferences and interests, the similarity between users is computed. The prediction for user u for location l can be mathematically expressed as equation (4):

$$P_{ul} = \overline{r_u} + \frac{\sum_{v \in N} \text{sim}\,(u,v) \cdot (r_{vl} - r_v)}{\sum_{v \in N} |\text{sim}\,(u,v)|} \qquad \text{Equation (4)}$$

where r_{u^-} is the mean preference of user *u*, *N* is the set of *k* nearest neighbours to user *u* in terms of Travel DNA, sim(*u*,*v*) is a similarity measure between users' Travel DNAs, and r_{ul} is the reactions given by user *v* to location *l*.

User-based CF model focuses on the similarities between target travelers and other users. By examining the preferences, behaviours, and choices of travelers who have displayed similar interests in the past, the system can extrapolate likely interests for the target traveller. If two users have historically liked the same travel destinations, the system might recommend to one user a location that the other user has liked.

6.6.2.2 Item-based Collaborative Filtering with Location DNA:

Utilizing Location DNA, which describes the features and characteristics of each location, item-based collaborative filtering is applied. The prediction for user u for location l can be represented as in equation (5):

$$P_{ul} = \frac{\sum_{j \in L} \text{sim}(l,j) \cdot r_{uj}}{\sum_{j=L} |\text{sim}\ (l,j)|} \qquad \text{Equation (5)}$$

where L is the set of locations with similar characteristics to location l in terms of Location DNA, sim(l,j) is a similarity measure between locations' Location DNAs, and r_{uj} is the rating or interest level given by user u to location j.

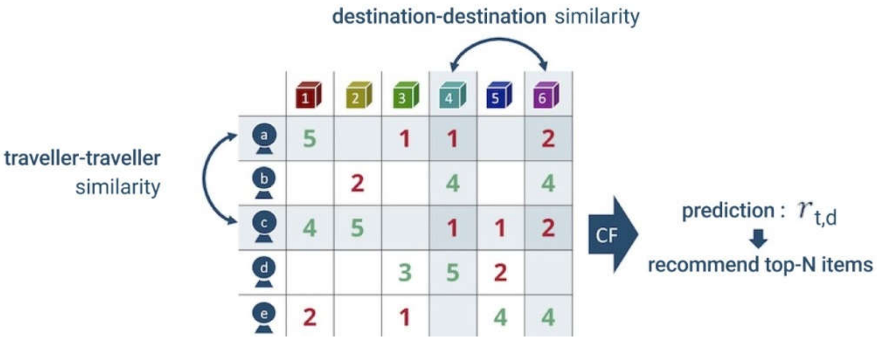
This approach measures the connection between the locations that travelers rate or interact with and other locations. Instead of looking at similarities between users, it focuses on the relationships between items (in this case, travel destinations). If a traveller has shown interest in destinations with certain features (such as adventure or historical significance), the system might recommend other destinations with those same features.

6.7 Personalized Destination Recommendation

In combining Travel DNA and Location DNA, collaborative filtering offers a personalized and content-rich way to model traveler's preferences. Travel DNA captures the nuanced preferences of individual travelers, while Location DNA encapsulates the distinct features and characteristics of each travel destination. By leveraging these two facets in collaborative filtering, the recommender system can provide more precise and context-aware travel suggestions [113], ultimately leading to a more personalized and satisfying travel planning experience.

6.7.1 Traveler-Traveler Similarity

Traveler-Traveler similarity measures the likeness between different travelers based on their Travel DNA. This comparison helps in understanding common preferences, tastes, and travel behaviours among users. By calculating the cosine similarity between the Travel DNA vectors of different travelers, the model can identify similar patterns and preferences. This enables the recommendation system to suggest destinations that have been preferred by travelers with similar tastes. User-based Collaborative Filtering leverages this similarity to predict and generate suggestions tailored to the interests of an individual traveller. Figure 17 shows the diagrammatic representation of traveller and destination similarities.



destination-destination similarity
traveller-traveller similarity
CF
prediction : $r_{t,d}$
recommend top-N items

	1	2	3	4	5	6
a	5		1	1		2
b		2		4		4
c	4	5		1	1	2
d			3	5	2	
e	2		1		4	4

Figure 17 Similarity between different features.

6.7.2 Location-Location Similarity

Location-Location similarity, on the other hand, focuses on finding connections between different locations based on their Location DNA. This involves analysing the features and characteristics of various destinations, such as travel mode, type, climate, and overall visits. By utilizing Item-based (or Location-based) Collaborative Filtering, the model can measure the similarity between different locations that traveler's rate or interact with. This leads to a more nuanced understanding of how different locations relate to each other in terms of traveller preferences.

In essence, Traveler-Traveler similarity, and Location-Location similarity act as key components in clustering and mapping the relationships between travelers and locations. They allow the recommender system to provide more personalized and relevant suggestions, thereby enhancing the overall user experience. By understanding the intrinsic connections between travelers and locations, the model can better cater to individual preferences, making the recommendations more precise and meaningful.

The Location-Activity (Loc. Type) Matrix as given in Figure 18, is a specialized structure that plays a vital role in travel planning, recommendation, and analysis. It represents the relationship between distinct geographical locations (l1, l2, ... ln) and various activities or experiences (adventure, trekking, historic, pilgrimage, etc.) that those locations offer. Each cell within this matrix illustrates the relationship between a particular location (x-axis) and an activity (y-axis). If a specific location offers an activity, the corresponding cell may be marked with corresponding index or a score that indicates the quality or popularity of that activity at that location. If the location doesn't offer the activity, the cell might be marked with a 0.

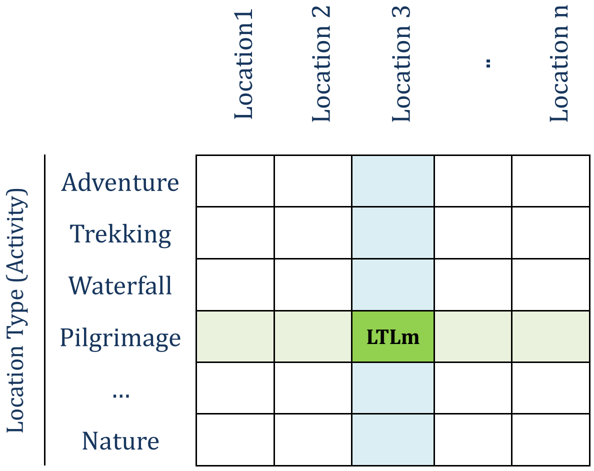


Figure 18 Location - Activity matrix

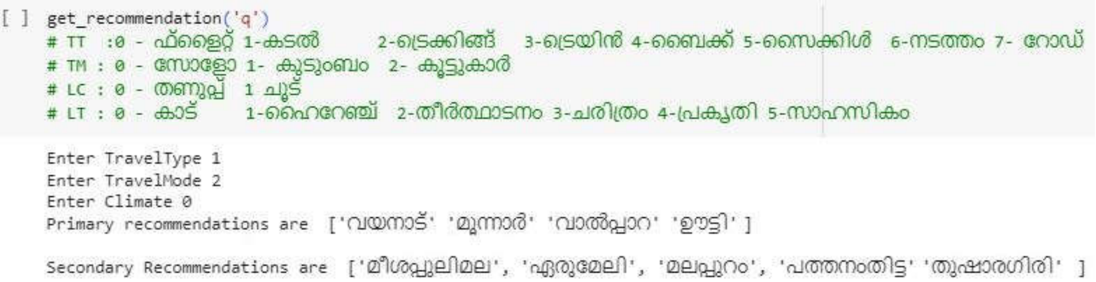
Through the analysis of Travel DNA, Location DNA, and the Rating matrix, the system can calculate implicit correlations between both travelers and locations. This comparative data is instrumental for internal clustering, enhancing the traveller–destination mapping process in various ways. Once the model is fully developed and adequately trained, it can deliver two specific types of recommendations: Primary and Secondary.

a) Primary Recommendations: This set consists of four locations that the system suggests for the user to visit, derived from analysing similar travelers with matching preferences and tastes. If any locations in this list have already been visited by the user, they are excluded from the recommendations.

b) Secondary Recommendations: This set is a curated list of five locations that are tailored to the user, ranked in descending order of suitability. These locations are considered particularly apt for the user's tastes and may include places that the user has previously visited. During the model's training phase, all these procedures are meticulously carried out, preparing the algorithm to recommend the most fitting five destinations for each user.

c) Prompt for User input: In the testing phase, the recommendation model included an interface to prompts the user to input specific details such as their preferred Travel Mode, Travel Type, Location type and Traveling Season. The Malayalam text and its corresponding index are displayed which will help the user to enter the corresponding numeric values of their wish.

Upon processing this information as user preference vector, the model presents the user with both the primary and secondary recommendation lists. These tailored suggestions enhance the user's travel planning experience, allowing for a more personalized and enjoyable journey. Prompt for user input and recommended locations are shown in figure 19.



```
[ ]  get_recommendation('q')
     # TT  :0 - ഫ്ളൈറ്റ് 1-കടൽ      2-ട്രെക്കിങ്ങ്   3-ട്രെയിൻ 4-ബൈക്ക് 5-സൈക്കിൾ  6-നടത്തം 7- റോഡ്
     # TM : 0 - സോളോ 1- കുടുംബം  2- കൂട്ടുകാർ
     # LC : 0 - തണുപ്പ്  1 ചൂട്
     # LT : 0 - കാട്     1-ഹൈറേഞ്ച്  2-തീർത്ഥാടനം 3-ചരിത്രം 4-പ്രകൃതി 5-സാഹസികം

     Enter TravelType 1
     Enter TravelMode 2
     Enter Climate 0
     Primary recommendations are  ['വയനാട്' 'മൂന്നാർ' 'വാൽപ്പാറ' 'ഊട്ടി' ]

     Secondary Recommendations are  ['മീശപ്പുലിമല', 'എരുമേലി', 'മലപ്പുറം', 'പത്തനംതിട്ട' 'തുഷാരഗിരി'  ]
```

Figure 19 Prompt for User Input

6.8 Experimental Result

Out of the listed recommendations provided by the model in each list, an average of four out of five locations were correct in the secondary list, translating to an 80% accuracy rate, and three out of four places were correct in the primary list, corresponding to a 75% accuracy rate as shown in Table 11. These results underscore the model's proficiency in identifying suitable travel destinations, with particularly strong performance in secondary recommendations, reflecting the robust integration of Travel DNA, Location DNA, cosine similarity, and collaborative filtering techniques within the model. A sample of list of recommended destinations for different users has shown in Table 10.

Table 10 List of recommended destinations for users

User	TT	TM	LC	Recommendations	
				Primary	Secondary
Bibin joseph	0 റൈഡ്	0 സഹോദരൻ	0 തണുപ്പ്	റാണിപുരം, ഓാമശ്ശേരി, മൂന്നാർ, ഊട്ടി	മൂന്നാർ, വയനാട്, കശ്മീർ, പൂനെ, പഞ്ചാബ്
	3 ട്രെയിൻ	1 കുടുംബം	0 തണുപ്പ്	പൊള്ളാച്ചി, ലഡാക്ക്, ഗോവ	കോട്ടയം, പഞ്ചാബ്, ഇടുക്കി, വയനാട്,ഗവി
	0 റൈഡ്	1 കുടുംബം	1 മഞ്ഞ്	വയനാട്	കാഞ്ഞങ്ങാട്, ഇടുക്കി,വയനാട്, മൈസൂർ, കൽക്കട്ട
Test	2 ട്രെക്കിംഗ്	1 കുടുംബം	0 തണുപ്പ്	രാജസ്ഥാൻ, ആനമുടി, മീഷപുലിമല, ഊട്ടി	നിലന്പൂർ, മേഘാലയ, ഭോപ്പാൽ, ഡൽഹി, ഷിംല
	3 ട്രെയിൻ	0 സഹോദരൻ	0 തണുപ്പ്	സേലം, ബാംഗ്ലൂർ, കൊടൈക്കനാൽ	പാലക്കാട്, കൊല്ലം, റോസ്മല, ലക്ഷദ്വീപ്, പൂനെ
	0 റൈഡ്	0 സഹോദരൻ	1 മഞ്ഞ്	ഗോവ	ഗവി, തെന്മല, ഗോവ, സേലം, ജിനി

For the primary recommendations:

$$\text{Accuracy}_{\text{primary}} = \frac{\text{Number of correct recommendations in the primary list}}{\text{Total number of recommendations in the primary list}} \text{ X } 100$$

$$\text{Accuracy}_{\text{primary}} = 3 / 4 \times 100 = 75\%$$

For the secondary recommendations:

$$\text{Accuracy}_{\text{secondary}} = \frac{\text{Number of correct recommendations in the Secondary list}}{\text{Total number of recommendations in the Secondary list}} \text{ X } 100$$

$$\text{Accuracy}_{\text{secondary}} = 4/5 \times 100 = 80\%$$

Table 11 Performance of accuracy of RS

Recommendation Type	Average testing count	Correct Recommendations	Total Recommendations	Accuracy
Primary list	50	3	4	75%
Secondary list	50	4	5	80%

6.9 Conclusion

For this research, a total of 13,458 full-length Malayalam travelogues were extracted, with each post containing an average of 40 sentences and 2006 records collected through google form. These extensive travelogues presented both a unique opportunity and a considerable challenge. The raw data, rich but unstructured, underwent rigorous data cleansing and preprocessing through natural language processing techniques. The Malayalam language, being highly inflectional and morphologically rich, required specialized handling in the preprocessing stages. Sentence tokenization, word tokenization, removal of punctuations, code mixing, stop words, stemming, and lemmatization were meticulously performed.

Travel DNA and Location DNA were pivotal in this work, as they provided structured representations of individual travelers' preferences and the characteristics of various destinations, encapsulating essential details like travel mode, type, climate, and purpose. By transforming unstructured travelogues into these structured forms, they facilitated precise clustering and correlation, enabling the collaborative filtering and cosine similarity techniques to make accurate and personalized travel recommendations.

Collaborative filtering played a crucial role in this work by enabling the recommendation system to make personalized suggestions based on the interests and preferences of individual travelers. By measuring similarities between different travelers and locations and utilizing both user-based and item-based models, collaborative filtering allowed the system to pinpoint suitable travel destinations, reflecting the implicit connections and tastes within the extensive Travel DNA and Location DNA datasets. Cosine similarity significantly contributed to this work by measuring the angle between vectors in a multi-dimensional space, representing the travel preferences in Travel DNA and Location DNA. This mathematical approach allowed the system to capture the orientation and not just the magnitude of the travel preferences, thereby identifying the similarity between various travel attributes, even if they were far apart in numeric terms, leading to more accurate and meaningful recommendations.

The system's two-tiered approach, offering Primary and Secondary Recommendations, showcases the potential for more nuanced and targeted recommendations. Primary recommendations facilitate exploration, guiding travelers towards new experiences that align with their tastes, while Secondary Recommendations provide a sense of familiarity by suggesting locations already visited and loved. The experimental results validated the efficiency and precision of the system, with an impressive 4 out of 5 destinations matching accurately in most attempts. Manual testing revealed an impressive 80% and 75% accuracy rate in the secondary recommendations and primary recommendations respectively. This not only exemplifies the technological success but also underlines the potential impact on enhancing user experience in travel planning. What sets this work apart is its unique focus on the Malayalam language, where there are no prior similar works in personalized Travel Recommender systems.

7 RS based on Clustering Techniques

7.1 Introduction

In an era where information is abundant and travel options are countless, personalizing travel recommendations has emerged as a key area of exploration. The Malayalam-speaking community, primarily in the southern state of India, Kerala, is aware of this technological evolution. However, Malayalam, known for its complex morphology, agglutinative nature, and rich inflectional structure, presents unique challenges [7]. Being a low-resourced language lacking standardization in spelling and sentence structure, and with the unavailability of benchmark datasets, building robust models in text and speech processing becomes difficult. This study embarks on a journey to overcome these hurdles by developing a personalized travel recommender system tailored to the Malayalam language.

Utilizing an innovative two-phase approach, this study has gathered and processed 13458 unstructured travelogues and reviews in Malayalam, sourced from various online platforms including social media. The travel data covers various aspects such as mode of transportation, type of travel, the location visited, and the climate of the destination. The absence of structured data has led to the employment of unsupervised clustering techniques [114], and the novelty of the task at hand lies in converting noisy, unstructured information into a usable, structured format.

In the realm of Recommender Systems (RS), collaborative filtering techniques play a critical role, where the nearest matches among users are determined by computing similarities in their behaviour. This experiment explores different metrics like Euclidean distance[115], cosine similarity, and city block distance, with cosine similarity emerging as the most effective in determining similarities among users. The choice of clustering techniques, including K-means and hierarchical agglomerative clustering, has been instrumental in shaping the Recommender System.

The construction of the recommender system was orchestrated in two discerning phases. Initially, a K-means clustering technique founded on collaborative filtering [116] was applied, succeeded by a hierarchical agglomerative clustering [117] method centered on content. These clustering techniques were pivotal in elevating both the efficacy and precision of the travel recommender system, with a striking 90% of suggested destinations featuring in the top three positions post-K-means clustering [118], and an impressive 85% after the agglomerative clustering. The corresponding F1 score measures stand at 92.04% and 84.25% for each approach respectively. The application of agglomerative hierarchical clustering, K-Means clustering, and collaborative filtering techniques has demonstrated the profound potential of deep learning algorithms in personalizing travel recommendations.

The major contributions of this study are:

- This research is distinctive in its endeavor to extract and methodically analyse a well-structured dataset derived from messy and unstructured Malayalam travel reviews and narratives found on social media platforms.
- The study introduces an innovative methodology for crafting a tailored travel recommendation system specifically for the Malayalam language, utilizing unsupervised clustering techniques alongside collaborative filtering approaches.
- The methodical approach encompassed the application of agglomerative hierarchical clustering, K-Means clustering, and collaborative filtering techniques. Additionally, empirically demonstrated the efficacy of deep learning algorithms [119] in precisely predicting travel destinations for individual travelers.

The study's systematic approach is delineated through five main sections, encompassing the data collection, feature engineering, construction of the recommender systems using collaborating filtering-based K-Means clustering, and content filtering-based hierarchical agglomerative clustering techniques, followed by performance evaluation. By extracting favorite locations, travel types, travel modes, and climate information, the research encodes the travel behaviour of each traveler. This research not only presents a milestone in Malayalam text processing but also provides a pathway to an era where language is no longer a barrier to accessing personalized travel recommendations.

7.2 Methodology

The process of developing the travel recommender system for the Malayalam language unfolded through a well-structured sequence of steps. It commenced with the Data Collection phase, where unstructured travel-related information was gathered. This was followed by Travel Feature Vectorization, a crucial step in translating the raw data into a workable format. Subsequently, Travel DNA construction was carried out to encapsulate the essence of various travel attributes. Collaborative Filtering using K-Means Clustering was then employed to identify patterns and similarities, followed by Content Filtering using Agglomerative Clustering to further refine the recommendations. The sixth phase involved constructing the recommender systems themselves, utilizing the insights and patterns identified in earlier stages. Finally, a Performance Evaluation of the two approaches was carried out, assessing the efficiency and accuracy of the systems, and ensuring they met the desired standards. The stages of development of RS are given in Figure 20.

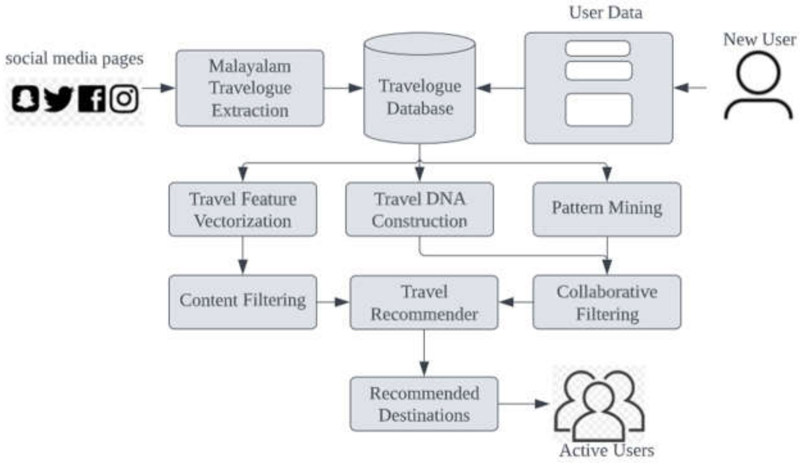


Figure 20 The proposed system architecture.

7.2.1 Dataset Preparation

In the dataset preparation phase of this study, data was drawn from an array of community sites, including 'Sanchari', the largest Malayalam Travel group on Facebook, along with various travel blogs that housed reviews and travelogues. The collection process involved web extraction techniques to glean online write-ups from random users, a task that posed significant challenges. The data thus obtained was notably unstructured, imbalanced, and inconsistent. To bring order to this data chaos, each write-up was carefully stored in individual files, designated by the respective traveler's name. The system was also designed to accommodate new user-generated travel content, with separate files dedicated to preserving these individual travel details for future retrieval. After rigorous preprocessing of the intricate and unstructured travelogues, the data was transformed into a well-structured tabular format, comprising 9 features. Moreover, the model's adaptability ensures that it can be periodically retrained to incorporate new information from the web, making it responsive to the constantly evolving landscape of travel experiences and preferences.

7.2.2 Feature Travel Vectorization

The collected data for the study exhibits considerable diversity in terms of content, posing challenges in structuring it for analysis. Variances in the length of travel posts, with some extending into detailed accounts while others remaining brief, add complexity to the dataset. Moreover, the presence of English words within Malayalam text introduces inconsistencies, as some posts contain a significant proportion of English terms, while others might not contain any. This blend of factors—varying length, language mixing, and diverse content—creates substantial disparities that necessitate a sophisticated feature engineering process, as illustrated in Figure 21, to transform the unstructured data into a form suitable for constructing a personalized travel recommender system.

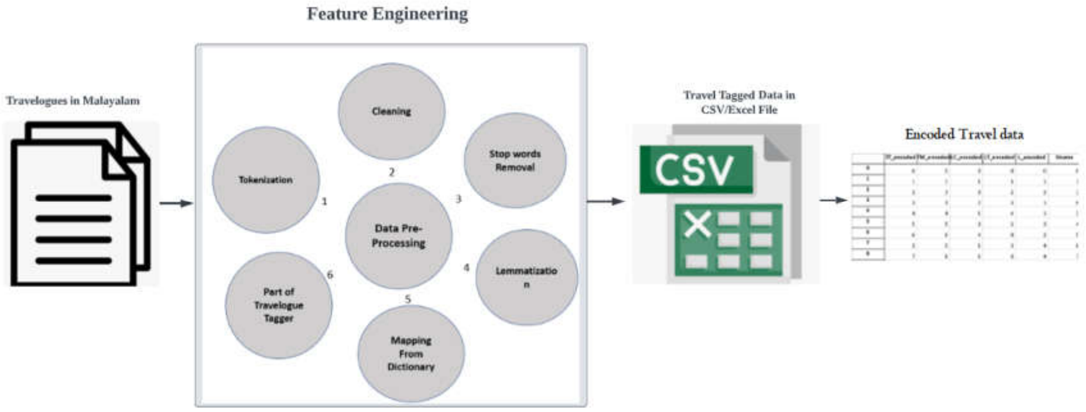


Figure 21 Feature engineering process

7.2.2.1 Tokenization

In the vectorization process, travelogues are individually extracted from each file and processed sequentially. The procedure commences with sentence tokenization, where the travelogues are broken down into separate sentences. Subsequently, word tokenization is carried out, dividing the sentences into individual words or tokens. For these tokenization tasks, the Punkt tool from the Natural Language Toolkit (NLTK) package in Python is employed, facilitating the transformation of normal sentences into word-level tokens.

7.2.2.2 Cleaning

The assembled data comprised travelogues that were also written in English, a language that was not pertinent to the specific study focusing on Malayalam. Consequently, any tokens that were not in Malayalam were deemed impurities and were meticulously removed utilizing suitable Python packages and methods. Additionally, the data set contained extraneous elements such as punctuation marks, numbers, and emojis. These were also eliminated using the regex module in Python, ensuring that the data was tailored exclusively to the Malayalam language.

7.2.2.3 Stopwords Removal

In the subsequent stage of data processing, the cleaned data is further refined through the removal of stopwords. Stopwords are common words that are generally filtered out during text processing, as they usually lack significant meaning in the context of analysis. In this specific study focusing on Malayalam, a curated list of 114 stopwords was used to filter out these unnecessary elements. It's worth noting that the precise selection and number of stopwords can vary, depending on the specific requirements and nature of the application for which the text processing is being conducted.

7.2.2.4 Stemming/Lemmatization

Following the removal of stopwords, the tokens are subjected to a stemming process. This step is crucial in extracting the most meaningful part of each word, known as the root or stem. For the Malayalam language, a common practice is to utilize the "Root_pack" lemmatization package, developed by ICFOSS, to accomplish this task. This package is known for its efficiency in handling the complex morphological structures of Malayalam.

7.2.2.5 Mapping from Dictionary

In the process of feature extraction, four primary attributes were identified and extracted from each file in the dataset: travel type, travel mode, location, and climate information. The root or base form of the words served as the input for this phase. Travel type pertains to the transportation means employed by the travelers, such as by road, train, air, or water. The mode of travel included categorizations like solo, family, colleagues, or friends. Climate information was also gleaned to understand travelers' weather preferences, capturing details like rainy, snowy, sunny, hot, or cold conditions. Location data was additionally extracted from the travelogues, detailing the specific destinations described. After this stage, the dataset was condensed to these four essential features from each travelogue, providing a more focused and relevant set of information for the study.

7.2.2.6 Part of Travelogue Tagger (POTT)

A personalized travel tagger was constructed utilizing Dhwanimam, a Part-of-Speech (POS) tagger designed specifically for the Malayalam language by ICFOSS. This travel tagger was further enhanced to include specialized tags representing the specific features of interest in this study, including TM for travel mode, TT for travel type, L for location, and LC for location climate. The result of this stage is a refined dataset with four key pieces of information extracted from each file. By applying the travel tagger to the tokenized data, appropriate tags were assigned to each token according to their maximum occurrence within the file. Travel modes such as solo, with friends, family, husband, wife, etc., were labelled with the TM tag, while travel types like train, bus, car, bicycle, bike, bullet, flight, etc., were classified under TT. Locations received the L tag, and the climatic conditions of the destinations, such as sunny, rainy, snowy, dry, wet, cold, hot, etc., were identified with the LC tag. An illustration of the applied POS tagging, and user behaviour analysis is provided in Table 12.

Table 12 The travel behaviour of the users.

User	Travel Type (TT)	Travel Mode (TM)	Location climate(LC)	Location (L)	Reactions	Comments	Shares
Anucftri	റോഡ്/ Road	സുഹൃത്ത്/ Friend	തണുപ്പ്/ Winter	ലേ / Leh	1500	243	23
Anucftri	ട്രെക്കിംഗ്/ Trekking		വേനൽ/ Summer	ഹംപി/ Hampi	992	214	22
Anucftri	തീവണ്ടി/ Train	ഭാര്യ/ Wife	ചൂട്/ Hot	ഇടുക്കി/ Idukki	263	83	15
Awin jose	ബസ്/ Bus	ബ്രോ/ Bro	ചൂട്/ Hot	ഡൽഹി/ Delhi	549	141	26
Awin jose	ബസ്/ Bus	സുഹൃത്ത്/ Friend		ഡൽഹി/ Delhi	48	23	2
Aslam	ട്രെയിൻ/ Train	സുഹൃത്ത്/ Friend		മൈസൂർ/ Mysore	108	24	2
Aslam	ബോട്ട്/ Boat	സുഹൃത്ത്/ Friend	മഴ/ Rain	ജോഗ്/ Jog	45	6	9
Joy Cheray	ജീപ്പ്/ Jeep	ചേട്ടൻ /Brother	വേനൽ/ Summer	വാഗമൺ / Vagamon	27	1	1
Joy Cheray	ട്രെക്കിംഗ്/ Trekking	സോളോ/ Solo		ഇടുക്കി/ Idukki	17	1	1

7.2.2.7 Save POTT data in CSV/spreadsheets.

Following the extraction of essential features from each travelogue, individual CSV/spreadsheets were created for every traveller, encapsulating the specific details of travel type, travel mode, location, and climate information. In instances where users had multiple travelogues, each one was accounted for as a distinct entry within the CSV file. Through this meticulous process, succeeded in transforming a collection of unstructured data into a systematically structured dataset. This dataset accurately symbolizes the travel predilections of each traveller, laying the groundwork for the personalized travel recommendation system.

7.2.2.8 Encode the data.

In the subsequent phase, the data contained within the CSV file was subject to one-hot encoding. This encoding technique assigns unique values to each entry within the file, thereby transforming the existing data into a format that is more readily processed. The result of this transformation was the creation of an encoded and structured dataset, optimized for the direct construction of the recommender system (RS). This encoding process ensures that the data is appropriately formatted for analytical processing, serving as a critical step in the development of the personalized travel recommender system.

7.3 Travel DNA Construction

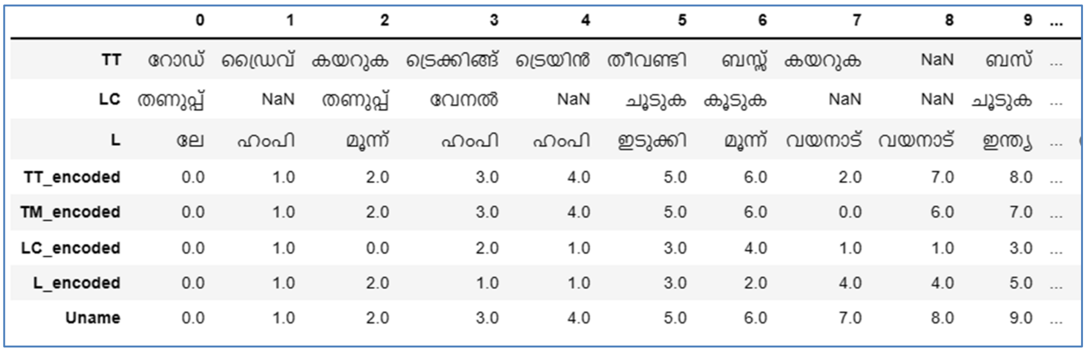
The method devised a unique Travel DNA for each traveler, which was built upon their encoded travel behaviour, encapsulating their distinct travel preferences and habits. This construction process was carried out in three primary steps:

7.3.1 Constructing a Preference Matrix

The preference matrix is meticulously constructed to encapsulate the user's travel behaviour by counting the repetition of four specific travel actions. If any behaviour is repeated three or more times by a user, it is regarded as their preferred mode of travel. To keep track of these repetitive behaviours within the feature vector matrix, a count vectorizer is employed. This technique allows for the observation of patterns, with a count of three or more indicating a favorite behaviour for a particular user. As a result, the user favorite matrix is shaped with dimensions of 8 rows and 6447 columns, each reflecting unique aspects of the user's travel preferences. This structure enables a detailed understanding of individual travel habits, forming an essential foundation for the personalized recommendation process.

7.3.2 Constructing the Travel List of Users

A travel list is meticulously crafted from the user's favorite destination matrix, which is derived from the previously established preference matrix. This list encapsulates all the destinations that the user has shown interest in, reflecting their specific preferences across the four assessed travel behaviours. By understanding these preferences, the travel list becomes a targeted compilation of locations that align with the user's likes and tendencies, creating a personalized selection tailored to individual desires. The intricate details of the user's favorite locations, captured within this process, are illustrated in Figure 22, providing a visual representation of the alignment between travel behaviour and destination preference.

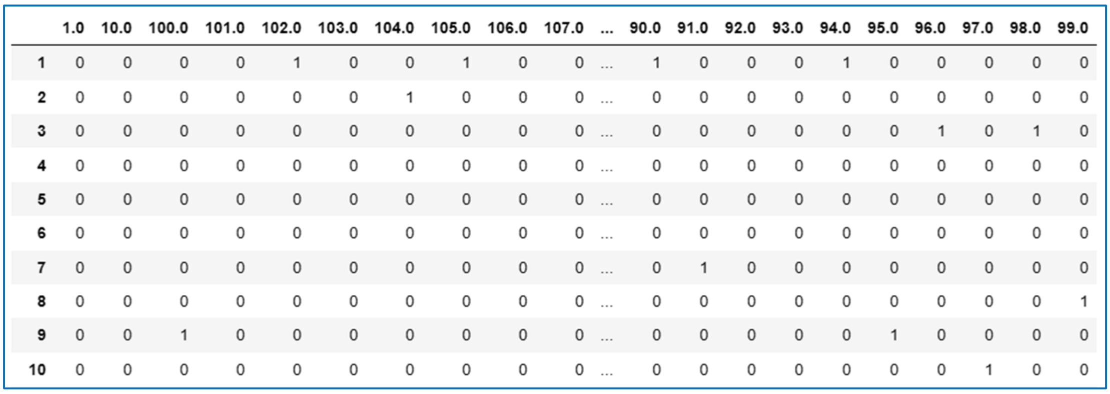


	0	1	2	3	4	5	6	7	8	9	...
TT	റോഡ്	ഡ്രൈവ്	കയറുക	ട്രെക്കിങ്ങ്	ട്രെയിൻ	തീവണ്ടി	ബസ്സ്	കയറുക	NaN	ബസ്	...
LC	തണുപ്പ്	NaN	തണുപ്പ്	വേനൽ	NaN	ചൂടുക	കൂടുക	NaN	NaN	ചൂടുക	...
L	ലേ	ഹംപി	മൂന്ന്	ഹംപി	ഹംപി	ഇടുക്കി	മൂന്ന്	വയനാട്	വയനാട്	ഇന്ത്യ	...
TT_encoded	0.0	1.0	2.0	3.0	4.0	5.0	6.0	2.0	7.0	8.0	...
TM_encoded	0.0	1.0	2.0	3.0	4.0	5.0	6.0	0.0	6.0	7.0	...
LC_encoded	0.0	1.0	0.0	2.0	1.0	3.0	4.0	1.0	1.0	3.0	...
L_encoded	0.0	1.0	2.0	1.0	1.0	3.0	2.0	4.0	4.0	5.0	...
Uname	0.0	1.0	2.0	3.0	4.0	5.0	6.0	7.0	8.0	9.0	...

Figure 22 Travel List of Users

7.3.3 Constructing a Sparse Matrix Based on the Preference Matrix

A sparse matrix is constructed from the traveler's curated list of favorite destinations, translating their interests into a binary format. Specifically, the matrix registers a '1' for any destination that appeals to a particular traveler and '0' for features they show no interest in. Alongside this, the feature names utilized to forge the sparse matrix are diligently preserved. The architectural arrangement of the matrix positions the columns to correspond to each specific feature, while the rows align with individual user IDs. With a dimensional structure of 6447 X 118, Figure 23 visually exhibits the sparse matrix transposed from the travel list table. This matrix serves as a comprehensive map of the entire user-travel behaviour, condensing a wide array of preferences into a streamlined and interpretable format.



	1.0	10.0	100.0	101.0	102.0	103.0	104.0	105.0	106.0	107.0	...	90.0	91.0	92.0	93.0	94.0	95.0	96.0	97.0	98.0	99.0
1	0	0	0	0	1	0	0	1	0	0	...	1	0	0	0	1	0	0	0	0	0
2	0	0	0	0	0	0	1	0	0	0	...	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	1	0	1	0
4	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	...	0	1	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	1
9	0	0	1	0	0	0	0	0	0	0	...	0	0	0	0	0	1	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	1	0	0

Figure 23 Sparse Matrix Based on the Preference Matrix

7.4 K-Means Clustering

K-means clustering is a widely recognized and straightforward algorithm in the realm of unsupervised machine learning. Unsupervised algorithms, unlike their supervised counterparts, derive insights from datasets solely through input vectors without relying on known or labeled outcomes. The K-means algorithm's operation within data mining begins with an initial set of randomly chosen centroids, designated as the starting points for each cluster. The algorithm then engages in an iterative process, executing repetitive calculations that continually refine and optimize the positions of the centroids. This refinement aims to minimize the sum of the squared differences between the data points in a cluster and its centroid, ultimately leading to more coherent and meaningful clusters. The diagrammatic representation of K-Means clustering is shown in Figure 24.

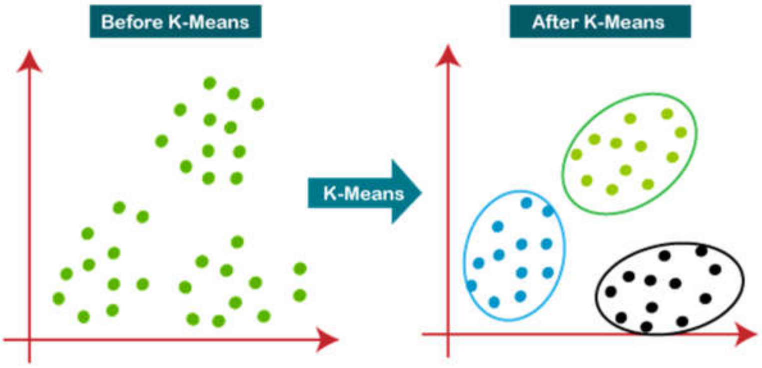


Figure 24 K Means Clustering

The K-means algorithm functions through a systematic six-step process to classify data into similar groups or clusters. These steps are as follows:

1. Determine the number of desired clusters, denoted as k.
2. Randomly select initial centroids for the clusters.
3. Begin a repetitive process involving the following two steps:
    a. Expectation: Assign each data point in the dataset to the nearest centroid, thereby forming distinct clusters.

    b. Maximization: Adjust the position of each centroid until it becomes the geometric center of its respective cluster.
4. Continue this process until the positions of the centroids no longer change, indicating that an optimal clustering solution has been reached.

The initial stage in K-means clustering involves identifying the optimal number of clusters for the given dataset. To achieve this, employed the elbow method [120], a technique that calculates the sum of squared distances of each data point from its assigned centroid. This metric is known as the Within Clusters Sum of Squares (WCSS) [121]. The objective is to minimize the WCSS, and the optimal number of clusters is determined where a noticeable change or "elbow" in the WCSS curve occurs [122]. The mathematical formulation for WCSS can be represented by Equation (6).

$$\text{WCSS} = \sum_{i=1}^{k} \sum_{j=1}^{N} Distance\left(P_{|i,j|,}C_{i}\right)^{\wedge}2 \qquad \text{Equation (6)}$$

Where K is the number of clusters and N is the total number of data points in each cluster, Ci is the centroid of ith cluster and P|i,j| is the jth data point in ith cluster.

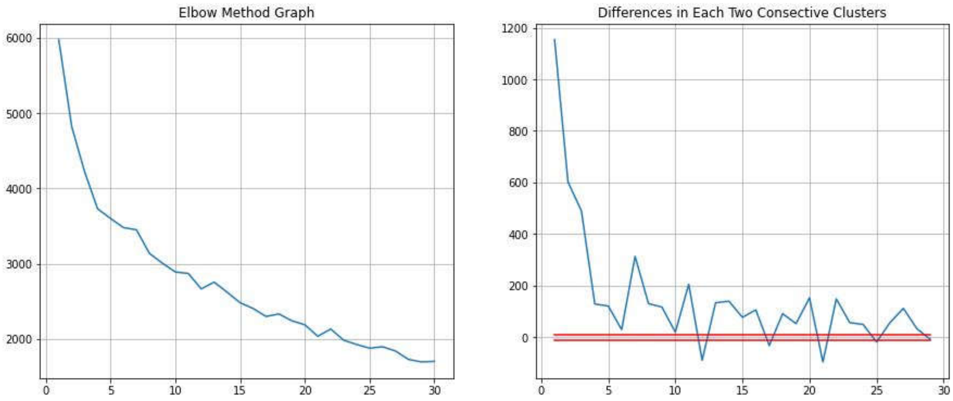


Figure 25 Elbow method graph

Utilizing the elbow method, ascertained that the optimal value for K was four, leading to the formation of four distinct clusters in this study.

Table 13 Clusters and Users

Sl. No	Cluster Number	No. of Users
1	Cluster 0	3166
2	Cluster 1	815
3	Cluster 2	1770
4	Cluster 3	696

Table 13 represents the distribution of users among the clusters. Cluster 0 contained 3166 users, Cluster 1 had 815 users, Cluster 2 comprised 1770 users, and Cluster 3 had 696 users. Figure 25 illustrates the elbow method graph and highlights the differences between consecutive clusters, effectively visualizing how arrived at the optimal number of clusters. The detailed representation of the users within each cluster can be found in Figure 26.

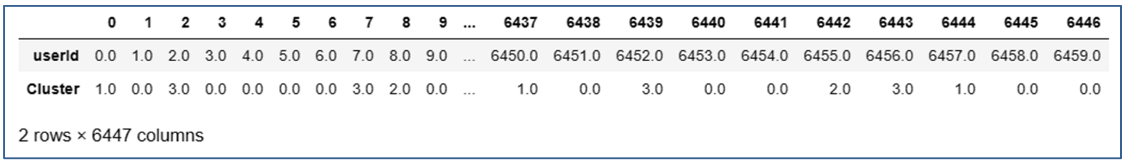

```
           0    1    2    3    4    5    6    7    8    9  ...    6437    6438    6439    6440    6441    6442    6443    6444    6445    6446
 userId  0.0  1.0  2.0  3.0  4.0  5.0  6.0  7.0  8.0  9.0  ...  6450.0  6451.0  6452.0  6453.0  6454.0  6455.0  6456.0  6457.0  6458.0  6459.0
Cluster  1.0  0.0  3.0  0.0  0.0  0.0  0.0  3.0  2.0  0.0  ...     1.0     0.0     3.0     0.0     0.0     2.0     3.0     1.0     0.0     0.0

2 rows × 6447 columns
```

Figure 26 Representation of users in their most suitable clusters.

7.5 Collaborative Filtering using K-Means Clustering

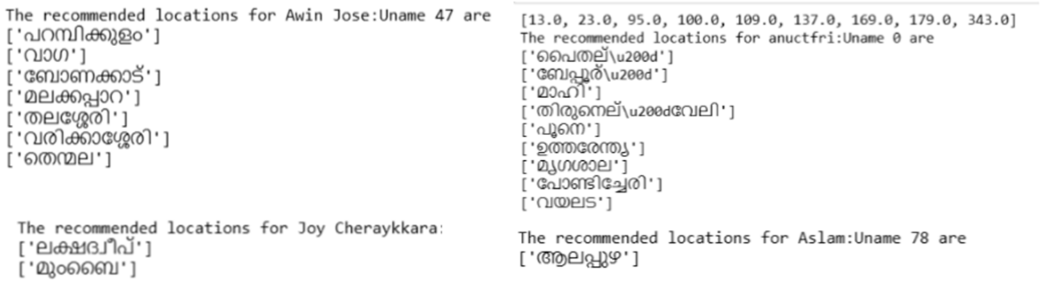
The travel preferences and behaviours of each user are meticulously captured and categorized. Users displaying similar travel inclinations are grouped together into clusters. Recommendations are then made by suggesting locations visited by users within the same cluster, who have a minimum cosine distance to one another. The ranking of these locations is influenced by the distance between the users in the cluster.

This recommendation model boasts a 91.01 % accuracy rate in proposing destinations to users that align with their individual preferences. Performance measures such as accuracy, F1 score, precision and Recall are given in Table 14. Utilizing collaborative filtering coupled with K-Means clustering, the system offers destination suggestions based on users' previous choices and the behaviours of similar travelers.

Table 14 Experimental result of K-Means Clustering

Observed result	
Accuracy	91.01 %
F1 Score	92.04 %
Precision	91.5%
Recall	92.06%

The construction of users' travel DNA considers their preferred modes of travel, types of travel, locations, and climatic conditions of previously explored destinations. Figure 27 provides a sample view of the recommendations generated by this method.



```
The recommended locations for Awin Jose:Uname 47 are
['പറമ്പിക്കുളം']
['വാഗ']
['ബോണക്കാട്']
['മലക്കപ്പാറ']
['തലശ്ശേരി']
['വരിക്കാശ്ശേരി']
['തെന്മല']
```

```
 The recommended locations for Joy Cheraykkara:
 ['ലക്ഷദ്വീപ്']
 ['മുംബൈ']
```

```
[13.0, 23.0, 95.0, 100.0, 109.0, 137.0, 169.0, 179.0, 343.0]
The recommended locations for anuctfri:Uname 0 are
['പൈതല്\u200d']
['ബേപ്പൂര്\u200d']
['മാഹി']
['തിരുനെല്\u200dവേലി']
['പൂനെ']
['ഉത്തരേന്ത്യ']
['മൃഗശാല']
['പോണ്ടിച്ചേരി']
['വയലട']
```

```
The recommended locations for Aslam:Uname 78 are
['ആലപ്പുഴ']
```

Figure 27 Sample Recommendations using Collaborative K-Means Clustering

For newcomers to the system, collected travelogues to formulate their unique travel DNA. This involves analyzing their preferences and behaviours to assess their proximity to the existing centroids of the clusters. Based on the cluster they are closest to, then recommended destinations that fall within the characteristics of that cluster. These recommendations align with the travel patterns of the cluster, ensuring a personalized experience. Figure 28 visually represents the clusters, plotted against parameters such as Location, Travel Mode, and Travel Type, offering a clear view of how the users are grouped based on these key features.

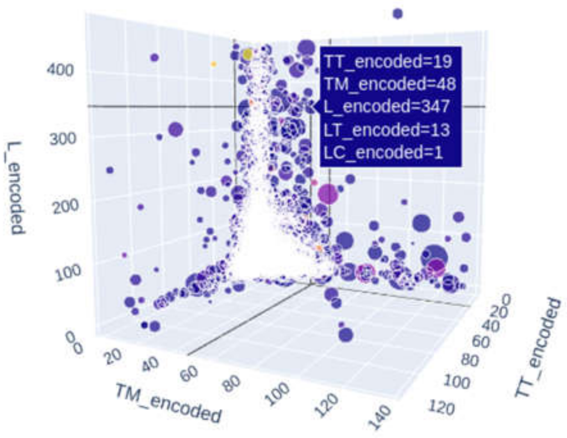


Figure 28 clusters, plotted against L, TT, TM

7.6 Content-Based Filtering Using Hierarchical Agglomerative Clustering

In the approach of Content-Based Collaborative Filtering Using Agglomerative Clustering, a personalized recommendation system is created by combining both content-based and collaborative methods and using the hierarchical clustering technique known as Agglomerative Clustering. Unlike partitioning methods like K-means, Agglomerative Clustering starts with each data point as a separate cluster and gradually merges them based on similarity. Content-based filtering analyses items and constructs profiles for user preferences, while collaborative filtering identifies users with similar behaviours. By considering both the content of the items and the collaborative patterns among users, this method can produce highly personalized recommendations. Integrating these features with Agglomerative Clustering, which builds a hierarchical cluster tree, allows for a nuanced understanding of user behaviour and preferences, thereby enhancing the accuracy and personalization of the travel recommendations.

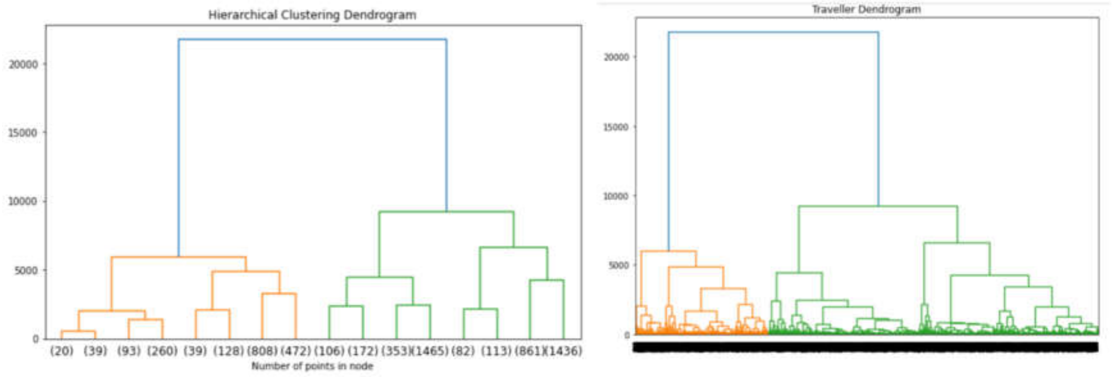


Figure 29 Agglomerative Clustering Dendrogram

The hierarchical agglomerative clustering process, depicted in Figure 29, proceeds through a series of stages focused on evaluating cluster similarity. The steps involved are as follows:

1. Build a proximity matrix utilizing one of the available distance metrics.

2. Treat each data point as its cluster initially.

3. Merge the clusters based on the selected similarity metric, thereby combining data points that are closely related.

4. Update the distance metrics to reflect the new clustering configuration.

5. Repeat steps 2, 3, and 4, continually merging clusters and recalculating distances, until only a single, comprehensive cluster remains.

In content-based collaborative filtering, additional features are integrated into the recommendation process, extending beyond the initial four factors used in phase 1. This approach considers the count of user likes, shares, and reactions to a specific travelogue or travel review, in conjunction with the previously considered four features. By doing so, user preferences for destinations are more closely aligned with their demonstrated interests. Formulated a proximity matrix to represent user affinities for specific destinations, assigning numerical values that reflect the relationship between users and destinations. Figure 30 demonstrates the interclass pairwise distances, which guided in selecting cosine similarity as the most appropriate metric for this study. Although explored the Euclidean distance, which calculates the straight-line distance between points in space using the Pythagorean theorem and found that cosine similarity provided more accurate results. Consequently, employed six distinct categories of travel behaviour and compared interclass pairwise distances using cosine similarity, Euclidean distance, and Manhattan/taxicab/city block distance methods, opting for cosine similarity as the optimal choice.

Assuming two points p and q, and d(p,q)/d(q,b) is the Euclidean distance between the two points, the Euclidean distance can be calculated by using the Equation (7).

$$d(\mathbf{p},\mathbf{q}) = d(\mathbf{q},\mathbf{p}) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \cdots + (q_n - p_n)^2} \\ = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}$$ Equation (7)

Cosine similarity is a metric used to measure how similar two vectors are. Essentially, it computes the cosine of the angle between the two vectors. This similarity measure ranges from -1 to 1, with a value of 1 indicating that the vectors are identical in orientation and a value of -1 indicating that they are diametrically opposed. A value of 0 means the vectors are orthogonal, or at a 90-degree angle to each other, reflecting no similarity.

The mathematical formula to calculate the cosine similarity between two points A and B is given by equation (8):

$$\text{similarity} = \cos(\theta) = \frac{\mathrm{A} \cdot \mathrm{B}}{\|\mathrm{A}\|\|\mathrm{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}}$$ Equation (8)

The city block distance, also known as Manhattan distance or L1 distance, calculates the distance between two points in space as the sum of the absolute differences of their coordinates. It's called the city block distance because it measures the distance a taxi would have to drive in a grid-like street layout, moving along the grid lines like you would in a typical city with perpendicular streets.

The mathematical formula for the city block distance between two points. In a plane with P at coordinate (x1, y1) and Q at (x2, y2). The city block distance between P and Q is determined by Equation (9).

$$M = |x1 - x2| + |y1 - y2|$$ Equation (9)

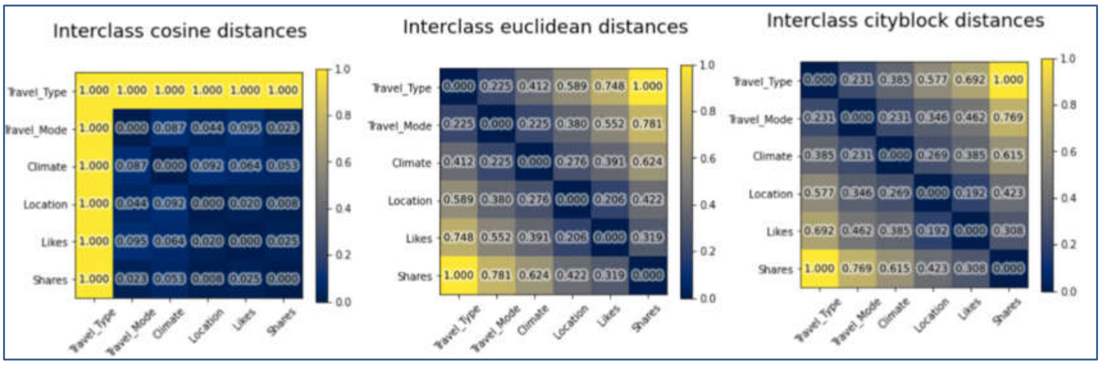


Figure 30 The interclass pairwise distances.

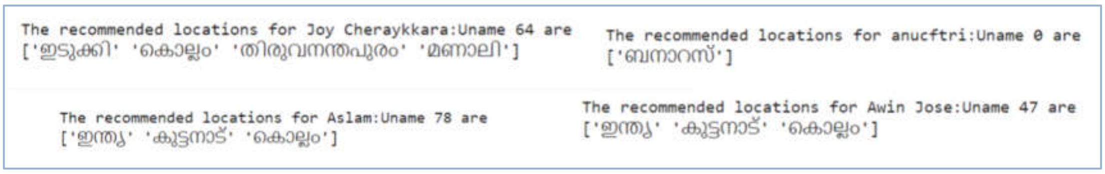
7.6.1 Experimental Results

The Recommendation model employing hierarchical agglomerative clustering in conjunction with content-based filtering achieved a performance accuracy of 85.01% and F1 score, precision and recall obtained as shown in Table 15. This is a notable result, given the complexity and rich texture of the data. In this approach, the recommendations are not solely based on conventional travel behaviours such as travel mode or location preferences. Instead, it extends to more dynamic and interactive features, including user interactions like likes and shares on specific travelogues or travel reviews.

Table 15 Observed result of HAC Clustering

Observed result	
Accuracy	85.01 %
F1 Score	84.25 %
Precision	84.15 %
Recall	84.35 %

Figure 31, as referred to, illustrates sample recommendations made by the recommendation system (RS). Here, the destinations are not arbitrarily selected but are the result of an intricate understanding of the user's travel patterns, combined with their active engagement with various travel content.



```
The recommended locations for Joy Cheraykkara:Uname 64 are
['ഇടുക്കി' 'കൊല്ലം' 'തിരുവനന്തപുരം' 'മണാലി']

The recommended locations for anucftri:Uname 0 are
['ബനാറസ്']

The recommended locations for Aslam:Uname 78 are
['ഇന്ത്യ' 'കുട്ടനാട്' 'കൊല്ലം']

The recommended locations for Awin Jose:Uname 47 are
['ഇന്ത്യ' 'കുട്ടനാട്' 'കൊല്ലം']
```

Figure 31 Sample recommendation in hierarchical agglomeration

7.7 Comparative Analysis and Performance Evaluation

The evaluation of the clustering method employed in this model reveals several notable characteristics: Efficiency, as the clustering process was achieved within two hours, making it a faster approach compared to building models with neural network architectures, tailored specifically for the dataset. Scalability is worth noting, as clustering larger datasets might require more time, indicating the need to consider this factor with more substantial volumes of data. The multi-step complexity of converting unstructured data into a structured format is essential but remains a time-consuming task. Adaptability is a key feature, allowing the model to learn new information from the web through regular retraining, ensuring it stays up to date. The practical output, illustrated in Table 15, contains personalized travel recommendations for various users, a testament to the model's ability to translate complex data analysis into actionable insights. In sum, clustering method is marked by its efficiency, scalability, intricacy in data transformation, adaptability, and tangible recommendations, underlining its practical utility in personalized travel recommendation systems.

Table 16 Recommendation to the users from two different approaches

Users	Preferences	Recommended Locations Using Collaborative filtering-based K-Means Clustering	Content Filtering-based Hierarchical Agglomerative Clustering
Anucfri	TM: friends, family TT: road, trek, train LC: Summer, Winter L: Leh, Humpi and Idukki	Paithal, Beypore, Mahe, Thirunelveli, Pune, North India, Zoo, Pondicherry, Vayalada	Banaras
Awin Jose	TM: friends, family TT: bus LC: winter L: Delhi	Parambikkulam, Vaga, Bonakkad, Malakkappara, Thalassery, varikkassery, Thenmala	Delhi, Kuttanad, Kollam
Joy Cheraykkara	TM: solo, friends TT: road, bike LC: -Summer L: wagamon, Idukki	Lakshadweep, Mumbai	Idukki, Kollam, Thiruvananthpuram, Manali
Aslam	TM: friend TT: train, boat LC: -rainy L: India, jog	Alappuzha	Delhi, Kuttanad, Kollam

The effectiveness of the two recommendation systems (RSs) developed in this research was assessed through key metrics, including accuracy, F1 score, Precision, and Recall. These measurements offer a comprehensive perspective on how well the systems perform in recommending personalized travel experiences. Table 16 encapsulates the detailed performance evaluation of both RSs, highlighting their capability and efficiency in aligning with the study's objectives.

The precision of the clustering is computed by summing the maximum number of objects in each cluster and then dividing by the total number of objects clustered. This metric can be expressed for an m x n matrix and is calculated as outlined in Equation (10). It provides a measure of how well the clustering algorithm has classified the objects into their respective groups.

$$\text{Precision} = \frac{\sum_m max_n\{a_{mn}\}}{\sum_m \sum_n a_{mn}} \qquad \text{Equation (10)}$$

The recall is calculated by selecting the cluster with the maximum number of objects assigned, summing the maximum number of objects for each cluster, and then dividing by the total number of both clustered and un-clustered objects. This measure, which can be expressed for an m x n matrix, is detailed in Equation (11). It provides an insight into the proportion of relevant objects that are successfully retrieved by the clustering algorithm.

$$\text{Recall} = \frac{\sum_n max_m\{a_{mn}\}}{(\sum_m \sum_n a_{mn}+U)} \qquad \text{Equation (11)}$$

The F1 Score is a harmonic mean of precision and recall, providing a balanced measure of the clustering algorithm's performance in terms of both retrieval and relevance. It can be calculated using Equation (12), utilizing the previously computed values of precision and recall. This score serves as a single metric that combines the influence of both precision and recall, offering a comprehensive view of the system's effectiveness.

$$\text{F1} = 2\,X\ \frac{\text{Precision X Recall}}{\text{Precision+Recall}} \qquad \text{Equation (12)}$$

Table 17 Comparative analysis of performance of models

Methodology	Accuracy	F1-Score	Precision	Recall
Collaborative Filtering Using K-Means Clustering	91%	92.04%	91.5%	92.6%
Content Filtering Based Hierarchical Agglomerative Clustering	85%	84.25%	84.15%	84.35%

The recommendation system (RS) model is designed with a broad perspective, encompassing numerous well-known and hidden tourist destinations across India, with a particular focus on Kerala. This personalized feature enables travelers to discover unexplored local tourist spots, which might not be prominently featured on mainstream websites or mobile applications. By analysing travelogues, it's possible to retrieve the latest trends and live statuses of various tourist locations, offering travelers the chance to explore lesser-known destinations. This can lead to a more authentic, enriching, and economical travel experience. Moreover, by highlighting and promoting local tourism through this approach, the RS could foster positive economic effects for various stakeholders within the tourism industry, including hotels, resorts, and local businesses.

7.8 Conclusion

In conclusion, the development and implementation of personalized travel recommendation systems using clustering methods have been an engaging and productive endeavor. The research successfully utilized techniques such as K-means clustering and hierarchical agglomerative clustering to generate precise travel recommendations for users. By employing both collaborative and content-based filtering, the models were able to consider a set of factors, including user travel behaviour, preferences, likes, and shares. This multifaceted approach allowed for an accurate prediction of travel destinations, with an impressive 91% accuracy for the K-means model and 85% for the agglomerative clustering technique.

Additionally, the research demonstrated the efficacy of using clustering in comparison to neural network architectures. The transformation of unstructured data into a structured format and subsequent clustering was achieved in a relatively shorter timeframe. This has the potential for scalability and adaptability, allowing the model to be retrained to accommodate new information and trends on the web. Furthermore, the system's focus on promoting local, lesser-known spots showcases a novel approach towards more sustainable and authentic tourism experiences.

Finally, this study represents a significant contribution to the field of personalized travel recommendation, particularly within the context of Malayalam language travelogues. It not only offers travelers a more customized and engaging experience but also paves the way for future research and development in this domain. The methods and insights derived from this research could potentially be applied to other languages and regions, thereby broadening the scope of personalized travel recommendations. This study lays the groundwork for innovative strategies that cater to the evolving needs and desires of modern travelers, supporting a more connected and enriching travel landscape.

8 Bi-LSTM Recommender System

8.1 Introduction

In an era where information overload is prevalent, recommender systems have become an indispensable tool in guiding users through an overwhelming array of choices. By tailoring suggestions to individual preferences and past behaviours, these systems not only personalize the user experience but also increase efficiency in decision-making. In the travel industry, recommender systems are especially vital, providing targeted suggestions from countless destinations, accommodation options, travel modes, and more. Personalized travel recommendations offer travelers unique and satisfying experiences, catering to their specific interests, budgets, and needs. Moreover, for a multilingual and culturally diverse population, language-specific recommendation systems can bridge the gap and resonate more deeply with local preferences, making them an important avenue to explore. Malayalam, a highly inflectional and morphologically rich language spoken by over 38 million people [123], has seen limited exploration in recommender systems. The nuanced expressions and unique cultural context embedded in the Malayalam language demand a customized approach to capture the subtleties that influence travel preferences. By focusing on Malayalam, this study not only contributes to broadening the reach of recommender systems but also emphasizes the value of language-specific modeling.

Malayalam, spoken predominantly in the Indian state of Kerala, is a language rich in cultural heritage and literary tradition. Unlike widely studied languages, Malayalam presents unique challenges due to its highly inflectional and morphologically complex nature. The development of a recommender system specifically tailored to Malayalam travel reviews is not merely a technological advancement but also a cultural empowerment. It recognizes and addresses the unique subtleties, expressions, and preferences embedded within the Malayalam-speaking community. Such a tailored approach ensures a deeper connection between technology and users, moving beyond generic models to provide meaningful and culturally nuanced recommendations.

Bidirectional Long Short-Term Memory (Bi-LSTM) architecture[124] is a cutting-edge deep-learning model that processes sequential data by understanding past and future contexts. Unlike traditional models, Bi-LSTM can capture temporal dependencies in a sequence, making it highly effective in analyzing text, especially in languages with complex structures like Malayalam. The architecture consists of forward and backward information flows, enabling it to learn intricate patterns and relationships within data. In the context of Malayalam travel reviews, Bi-LSTM's ability to decipher nuanced expressions and sentiments makes it an optimal choice for this study [84]. Its application paves the way for a more comprehensive and insightful analysis of travel preferences, as expressed in native language travelogues.

This chapter encapsulates an ambitious endeavor to innovate in the field of recommender systems by creating a tailored solution for Malayalam travel reviews. It delineates the process of selecting and extracting features that capture essential aspects of travel preferences and describes the application of the Bi-LSTM architecture to process these features. The comprehensive dataset, containing a variety of travel experiences, is examined to understand its contribution to model accuracy. The chapter also highlights the impressive results achieved through the experimental evaluation, showcasing an accuracy of 83.65 percent. By focusing on the Malayalam language, this work takes a significant step towards linguistic inclusivity and resonates with a wider audience. It also sets the tone for the detailed exploration of methodology, experiments, and results that form the subsequent sections of this chapter, illustrating a novel convergence of technology, culture, and language.

8.2 Data Collection and Dataset Creation

Facebook's "Sanchari" group has emerged as a virtual gathering place for Malayalam-speaking travel enthusiasts, cultivating a community of over 700,000 members. This platform serves as more than a social media hub; it's a rich repository of travel experiences, opinions, and recommendations all penned in the Malayalam language. The travelogues shared within the group reflect a wide spectrum of journeys, destinations, travel modes, and personal insights. With over 50,000 travelogues shared, the content offers a unique opportunity to delve into the preferences and desires of Malayalam-speaking travelers. Given the specificity and richness of this data, "Sanchari" became an ideal source for developing a recommender system that could resonate with the Malayalam-speaking community on a deeper level.

Extracting relevant data from the "Sanchari" group required a specialized approach that could navigate the complexity and volume of the content within the group. The extraction process also captured associated metadata such as user details, dates, comments, likes, shares, and reactions, enriching the dataset with multifaceted insights. The customized nature of the tool ensured that the collected data aligned with the study's objectives, thereby laying a strong foundation for the subsequent analysis. The steps of development of Bi-LSTM based recommendation model are given in Figure 32.

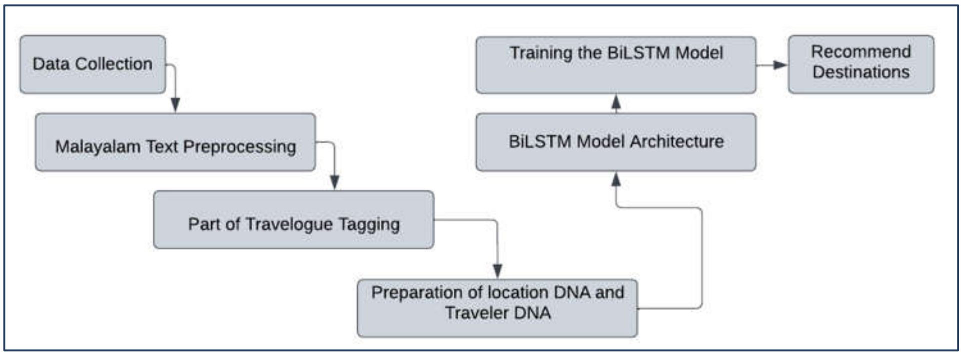

Data Collection
Malayalam Text Preprocessing
Part of Travelogue Tagging
Preparation of location DNA and Traveler DNA
BiLSTM Model Architecture
Training the BiLSTM Model
Recommend Destinations

Figure 32 Steps in Recommendation model using Bi-LSTM

8.3 Methodology

Once collected, the dataset underwent a meticulous preparation process to transform it into a structured format suitable for analysis. This stage involved cleaning the data by removing any inconsistencies, irrelevant information, or duplicates. Subsequently, the travelogues were parsed to extract essential features such as travel type, travel mode, location climate, and location type. The dataset was then segmented and annotated, aligning with the specific requirements of the Bi-LSTM model. Special attention was paid to maintaining the linguistic richness of the Malayalam language, preserving its inflectional and morphological characteristics.

8.3.1 Part of Travelogue Tagger

The creation of a dataset suitable for modeling in the context of Malayalam travelogues necessitated specific linguistic tools tailored to the rich and complex nature of the Malayalam language. With the help of a Part-of-Speech (POT) Tagger, each word in the travelogues was identified and annotated according to its grammatical role within the sentence.

This detailed tagging process allowed for a precise understanding of the syntax and semantics within the travel narratives. Alongside the POT Tagger, a specialized look-up dictionary was employed to associate words with specific travel-related features such as travel type, travel mode, location climate, and location type. The combination of the POT Tagger and the look-up dictionary-enabled the construction of a structured and semantically rich dataset, capturing the essence of the original travelogues while organizing them in a format conducive to the Bi-LSTM modeling process. This hybrid approach was instrumental in maintaining the linguistic integrity of the Malayalam language while facilitating the extraction and interpretation of key features, laying a robust groundwork for the subsequent stages of the recommendation system development.

8.3.2 Travel DNA and Location DNA

Travel DNA and Location DNA are vital concepts that play a key role in understanding and categorizing user preferences and travel destinations in the recommender system. Travel DNA refers to the unique set of characteristics, preferences, and behaviours exhibited by a traveler, such as the preferred travel mode, travel type, companions, and even preferences related to climate and specific experiences.

These attributes form a profile that can be used to predict and suggest personalized travel experiences. Location DNA, on the other hand, encapsulates the defining features of a travel destination, such as its climate, geography, types of attractions [125], culture, and more. It essentially captures the essence of a location and enables the recommendation system to match it with the Travel DNA of a user. By understanding both Travel DNA and Location DNA, the system can create a synergistic match between travelers and destinations, leading to more personalized and satisfying travel recommendations. The unique interplay between these two concepts is instrumental in delivering a nuanced and individualized user experience, effectively bridging the gap between travelers' desires and the most fitting travel experiences.

8.4 Bi-LSTM Introduction

The Bidirectional Long Short-Term Memory (Bi-LSTM) model architecture is a powerful advancement in the realm of Recurrent Neural Networks (RNNs). Unlike standard RNNs, Bi-LSTMs are equipped to learn temporal dependencies from both past and future states, making them well-suited for sequence prediction tasks such as the ones encountered in travel recommendations. Essentially, Bi-LSTMs consist of two LSTM (Long Short-Term Memory) layers that run in opposite directions, effectively capturing information from both the past and the future states of a sequence. This bidirectional approach allows for a richer representation of the input sequence and a more nuanced understanding of the context, leading to superior predictive performance.

In the context of the Malayalam travel recommender system, the Bi-LSTM architecture plays a pivotal role in processing and learning from the sequential data represented by travelogues. By capturing the complex dependencies and intricate patterns within the Malayalam language, Bi-LSTM facilitates the understanding of users' preferences and behaviours, which is essential for generating personalized recommendations. The two layers of LSTM operate in conjunction with the tailored feature extraction process, learning from the Travel DNA and Location DNA to understand the unique relationships between travel mode, travel type, location climate, and location type [126]. This comprehensive analysis enables the model to make precise and relevant travel recommendations, making the Bi-LSTM an integral part of the recommender system's success.

8.4.1 Recurrent Neural Network (RNN)

Recurrent Neural Networks (RNNs) are a class of artificial neural networks designed for handling sequential data [127]. Unlike traditional feedforward neural networks, RNNs possess memory to store previous outputs, allowing them to maintain a kind of "state" information. This capability makes them uniquely suitable for tasks involving sequences, such as time-series prediction, natural language processing, speech recognition, and, in the case of this research, understanding and interpreting travelogues.

An RNN operates by looping through sequence elements and maintaining a hidden state that captures information about previous steps in the sequence [128]. This hidden state serves as a contextual clue, linking past information to present decisions. However, standard RNNs often face challenges in dealing with long-term dependencies due to what is known as the vanishing gradient problem. This issue arises when the network is trained using gradient-based methods, leading to exponentially smaller gradients as the sequence length increases. As a result, the network struggles to learn from information early in the sequence, hindering its ability to understand and process long sequences effectively.

8.4.2 Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) networks are a specialized form of Recurrent Neural Networks (RNNs) that are designed to avoid the long-term dependency problem inherent in traditional RNNs. This ability to capture long-term dependencies in a sequence makes LSTMs particularly well-suited for sequence prediction tasks and various applications involving time-series data [129].

An LSTM unit consists of three gates - input gate, forget gate, and output gate - along with a cell state. These components work in conjunction to regulate the flow of information through the cell. Input Gate determines the extent to which a new value flows into the cell state. It uses a sigmoid activation function to compute a weight between 0 and 1 for the incoming input and the previous hidden state, which is then multiplied by a tanh-activated transformation of the same values. Forget gate determines how much of the previous cell state is retained or forgotten. Similar to the input gate, it computes a weight between 0 and 1 using a sigmoid activation function and then multiplies it with the previous cell state. Cell State is a sort of "memory" that carries information throughout the sequence processing. It's modified by the forget and input gates, allowing the LSTM to either retain or forget information as needed for the task at hand. A comparative diagram is given as Figure 33.

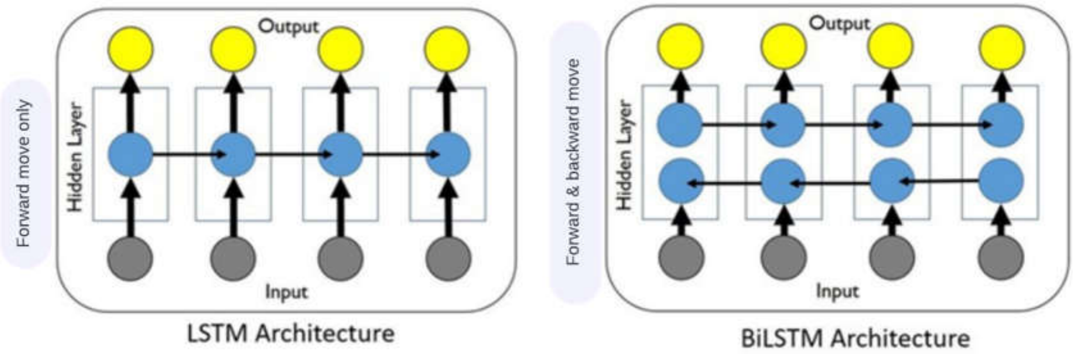


Figure 33 LSTM and Bi-LSTM

Output Gate is working based on the previous hidden state and current input, this gate determines what the next hidden state should be. It uses the previous hidden state and the input, passed through a sigmoid function, and multiplies that by the tanh of the (potentially modified) cell state. LSTMs have been successfully applied in various fields, including natural language processing[130], speech recognition, video analysis, and time-series forecasting, to name just a few.

8.5 Design of Bi-LSTM model Travel Recommender System

The design of the Bi-LSTM model Travel Recommender System was structured into a sequence of specialized stages. Input Encoding served as the foundational step, transforming the textual data into a numerical format suitable for processing. The Bi-LSTM Layer was then crafted with Forward and Backward LSTM units to capture the temporal relationships within the data, with the information from both directions concatenated for a more comprehensive representation. The Hidden State and Cell State were managed to preserve information across the sequences, while Dropout and Regularization [131] techniques were employed to prevent overfitting. A Time Distributed Layer was added to apply predictions to each time step, enabling multi-label classification. The model utilized a mean squared error Loss Function and was optimized through techniques like Adam. Hyperparameter Tuning[132] was conducted to fine-tune the model's settings, and finally, the model was trained and evaluated to gauge its effectiveness in predicting travel recommendations based on Malayalam travel reviews. The design of Bi-LSTM model Travel Recommender System is expressed in Figure 34.

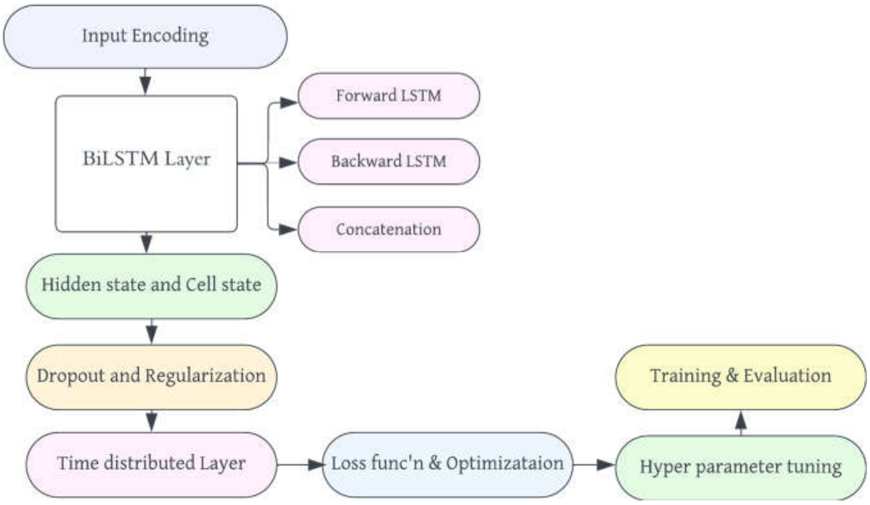


Figure 34 Design of Bi-LSTM model Travel Recommender System

8.5.1 Input Encoding.

The first phase of the Bi-LSTM model deals with converting raw textual data into a form that the network can understand [133]. Given that the dataset consists of Malayalam travelogues, it's paramount to translate this human-readable information into machine-readable numerical data. This is done by a series of operations such as Preprocessing and Feature Extraction. The next task is encoding. Each travel review, represented as a sequence of words, is then converted into numerical representations. This transformation is achieved through one-hot encoding, which creates a binary vector for each word in the vocabulary. The vector has a "1" in the position corresponding to the word's index in the vocabulary and "0" elsewhere. The encoded vectors are then assembled into a matrix that serves as the input to the Bi-LSTM model. This matrix is crafted in a way that each row represents a review, and each column within the row corresponds to a specific feature or word, encoded through one-hot encoding [134].

8.5.2 Bi-LSTM Layer

The Bi-LSTM layer represents the heart of the architecture, where the actual learning takes place. Comprising two LSTM networks, one processing the sequence forwards and the other backward, the Bi-LSTM layer is capable of recognizing patterns with temporal dependencies in both directions. Below is an explanation of how this layer operates.

8.5.2.1 Forward LSTM

The forward LSTM processes the input sequence from the first word to the last. Given an input sequence of length T, represented as x = (x_1, x_2, x3,x4), where each x_i represents the input at time i, the forward LSTM computes the hidden states $h^{t(0)}$ and cell states $c^{t(0)}$ using the forward propagation equations.

$$h^{t(0)} = \text{LSTM\_forward}(x^t, h^{t-1(0)}, c^{t-1(0)}) \quad \text{Equation (13)}$$

$$c^{t(0)} = \text{LSTM\_ forward\_cell}(x^t, h^{t-1(0)}, c^{t-1(0)}) \quad \text{Equation (14)}$$

8.5.2.2 Backward LSTM

Similarly, the backward LSTM computes the hidden states $h^{t(1)}$ and cell states $c^{t(1)}$ using the backward propagation equations:

$$h^{t(1)} = \text{LSTM\_backward}(x^t, h^{t+1(1)}, c^{t+1(1)}) \quad \text{Equation(15)}$$

$$c^{t(1)} = \text{LSTM \_backward\_cell}(x^t, h^{t+1(1)}, c^{t+1(1)}) \quad \text{Equation(16)}$$

8.5.2.3 Concatenation

In a bidirectional Long Short-Term Memory (Bi-LSTM) architecture, the output at any given time step t is typically formed by concatenating the hidden states from both the forward and backward LSTM. This allows the model to capture information from both past and future contexts in the sequence, making Bi-LSTMs particularly powerful for sequential data like text. Here's how it is usually formulated,

$$y^{t} = [h^{t(0)}, h^{t(1)}] \qquad \text{Equation(17)}$$

8.5.3 Hidden State and Cell State

Hidden State is a dynamic record of the information that has been seen by the LSTM so far in the sequence. It represents the "memory" of the LSTM, encapsulating all the relevant information from past inputs up to the current time step. In a Bi-LSTM, the forward LSTM maintains a hidden state that considers all the previous inputs, while the backward LSTM maintains a hidden state considering all the future inputs. When concatenated, they form a complete view of the contextual information around each point in the sequence.

Cell State, on the other hand, acts as an "internal memory" of the LSTM, regulating the flow of information within the unit. It can remember or forget certain parts of the sequence, based on the significance of the information. The LSTM's gating mechanisms, namely the forget gate, input gate, and output gate, control the updates to the cell state, enabling it to retain important information and discard irrelevant details.

8.5.4 Dropout and Regularization

Dropout and regularization are vital techniques used in training deep learning models, including Bi-LSTMs, to prevent overfitting. Dropout is a regularization technique that adds some form of noise or randomness during the training process to make the model more robust. By randomly setting a fraction of the input units to 0 at each update during training time, dropout helps prevent overfitting. The "dropout rate" is the fraction of the input units to drop, and it's a hyperparameter that needs to be chosen carefully. In the context of the Bi-LSTM layer, dropout can be applied to the connections between LSTM units or even between different layers. When applied, the model can no longer rely on any specific feature or connection, thereby forcing it to learn more generalized and robust representations. The key benefit of dropout in the Bi-LSTM is that it helps the model generalize better from the training data to unseen data, reducing the risk of overfitting. It pushes the model to learn more independent and distributed features, enhancing its overall predictive power.

Regularization is a technique that adds some form of penalty to the loss function, discouraging the model from fitting the training data too closely. This can be done through methods like L1 or L2 regularization[135], which add terms to the loss function that penalize large weights in the model. Regularization in the Bi-LSTM layer ensures that the model does not become overly complex and overfit the training data. By controlling the magnitude of the weights, regularization keeps the model simpler and more likely to generalize well to unseen data.

8.5.5 Time Distributed Layer

After the Bi-LSTM layer, a time-distributed layer was integrated into the model architecture. This layer enabled the application of the prediction layer to each time step within the sequence, allowing for independent predictions at every step. By doing so, it facilitated the complex task of multi-label classification required for predicting travel recommendations. The time-distributed layer acted as a critical bridge between the sequential understanding of the Bi-LSTM layer and the specific prediction requirements, translating the extracted features into actionable recommendations for each user, thus underscoring its importance in the model's overall functionality.

8.5.6 Loss function and Optimization

The model was trained to employ the mean squared error (MSE) loss function, a common measure used to calculate the average squared difference between predicted and actual values. During the training process, various optimization techniques, including stochastic gradient descent (SGD) [136], Adam, and RMSprop, were experimented with to iteratively update the model's parameters and minimize the loss function.

Among these, the Adam optimizer was chosen in conjunction with the mean_squared_error loss function to effectively minimize errors. This combination leverages the gradients of the MSE loss, allowing the Adam optimizer to adaptively update the model's parameters and achieve a more precise alignment with the Malayalam travel reviews, thus enhancing the recommender system's accuracy and robustness. The formula for MSE with Adam optimizer can be represented as in equation 18:

$$\text{MSE} = (1 / n) * \Sigma(\text{y\_pred} - \text{y\_actual})\text{\^{}}2 \qquad \text{Equation (18)}$$

Where n is the number of samples in the dataset, y_pred represents the predicted values and y_actual represents the actual values. The Adam optimizer adjusts the model's parameters during training by calculating the gradients of the MSE loss and applying appropriate updates based on the adaptive learning rates for each parameter.

8.5.7 Hyperparameter and Tuning

Hyperparameter tuning played a significant role in optimizing the performance of the Bi-LSTM model, addressing various influential parameters such as the learning rate, batch size, number of LSTM units, dropout rate, and number of epochs. Experimentation with these hyperparameters allowed the model to achieve a finely tuned balance between accuracy and generalization, tailored to the unique characteristics of the Malayalam travel reviews dataset. Careful adjustment and optimization of these hyperparameters were instrumental in the success of the model, ensuring that it captured the essential patterns and relationships within the data while avoiding overfitting. The construction of the neural network is represented in Figure 35.

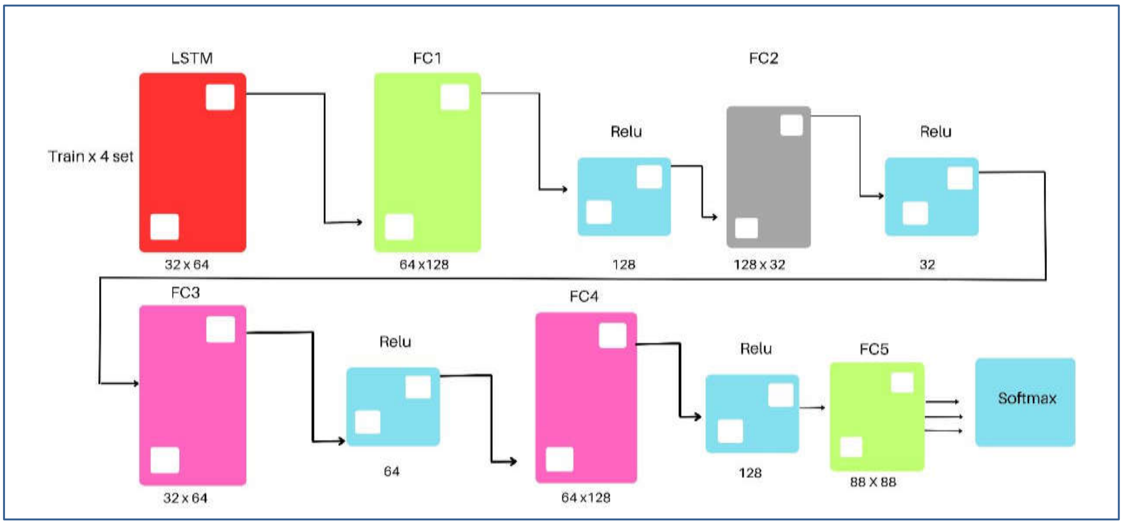


Figure 35 Architecture of Bi-LSTM Neural network

8.5.8 Training and Evaluation

The training and evaluation of the Bi-LSTM model were conducted by dividing the dataset into distinct training, validation, and testing sets. During the training phase, the model's parameters were iteratively adjusted to minimize the loss function specific to the training data. Simultaneously, the validation set was utilized to fine-tune the model and prevent overfitting, while the testing set offered an unbiased evaluation of the model's performance. The model's effectiveness in recommending travel options was assessed based on its accuracy in predicting the relevant features from the validation and testing sets, providing a comprehensive and robust evaluation of its capabilities in understanding and responding to user travel preferences.

The constructed recommender system harnessed the power of the Bi-LSTM, skillfully capturing the sequential characteristics inherent in the Malayalam travel reviews. The unique nature of bidirectional processing within the Bi-LSTM enabled it to learn and understand intricate patterns and relationships present in the data. Combined with the integration of regularization techniques like dropout, the model achieved a heightened ability to generate accurate travel recommendations. By meticulously aligning these aspects, the system ensured that the recommendations were based on the significant extracted features, thereby providing personalized travel suggestions that resonated with the unique preferences of individual users.

8.6 Experimental Results

The integration of a Bidirectional Long Short-Term Memory (Bi-LSTM) model provides a comprehensive and holistic perspective on the input sequence by leveraging information from both past and future contexts. This unique architectural design enhances the model's ability to accurately identify and interpret the distinctive features of textual travel patterns.

By capturing contextual information from both directions, the Bi-LSTM model excels in tasks related to pattern matching. During the experimentation phase, the architecture of the Bi-LSTM model underwent rigorous fine-tuning. Multiple combinations of optimizers, loss functions, and neural network architectures were explored to identify the most effective configuration, and only the refined and optimized version of the architecture is considered as the output of recommendation as shown in Table 18.

Table 18 Hyper parameter tuning of Bi-LSTM network.

	Optimizer	Loss function	epoch	Train Accuracy	Val. accuracy
Phase I	SGD	Categorical cross entropy	1000	46.76 %	43.87 %
			1500	56.87 %	51.63 %
Phase II	Adam	Mean squared error	800	66.71 %	58.14 %
			1500	83.65 %	69.41 %
			2000	80.01 %	63.17 %
Phase III	RMS_prop	Cat cross entropy	800	71.31 %	66.56 %
		MSE	1500	74.03 %	67.65 %

The effectiveness of the model was evaluated using test and validation accuracies, which served as metrics to assess its performance in each phase of the research. To construct the Bi-LSTM architecture, several key parameters were carefully selected. The initial learning rate was set to 0.01, the loss function used was mean_squared_error, and the activation functions employed were Relu and softmax. The optimizer chosen was Adam, known for its efficiency in optimizing deep learning models. The model was trained for a total of 1500 epochs, ensuring sufficient iterations for convergence and improved performance. This extensive training phase allowed the model to learn the underlying patterns and features of the Malayalam travel reviews, enabling it to make precise predictions and interpretations.

By employing this optimized Bi-LSTM architecture, the research aimed to achieve the highest level of accuracy and performance in predicting and interpreting travel patterns based on the extracted features from the Malayalam travel reviews. The chosen parameter settings and fine-tuning process aimed to maximize the model's potential and deliver reliable and robust results. Figure 36 shows the accuracy and loss of the Bi-LSTM model performed according to the parameters discussed above, summarizing the successful outcome of this experimental setup.

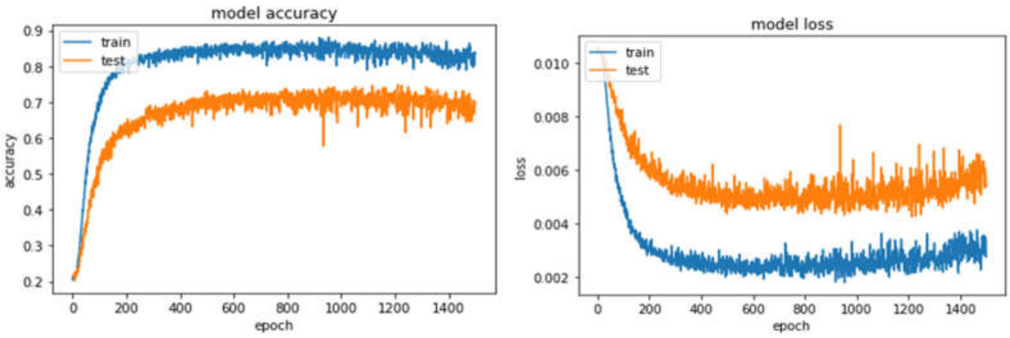


Figure 36 The learning curves of the experiment

8.7 Results and Discussion

The objective of this study was to predict travel destinations by considering significant input variables such as travel type, mode of travel, location type, and location climate. These variables are key factors in determining user preferences for different destinations based on various combinations. To address this task, a Bi-LSTM model was constructed with multiple layers and activation functions, including ReLU and Softmax.

During the training phase, the Bi-LSTM model demonstrated promising results, achieving a training accuracy of 83.65% and a training loss of 0.004. These metrics indicate that the model successfully learned the underlying patterns and relationships between the input variables and the corresponding travel destinations. The high training accuracy suggests that the model effectively captures the complexities of the data and performs well in predicting destinations based on the provided input.

In the subsequent validation phase, the model achieved a validation accuracy of 69.41% and a validation loss of 0.006. Although slightly lower than the training accuracy, this result demonstrates the model's ability to generalize and make accurate predictions on unseen data. Table 19 represents the performance of training and validation accuracies and loss. The validation accuracy indicates that the model can effectively apply its learned patterns to new combinations of input variables, thereby providing reliable travel destination recommendations.

Table 19 Performance evaluation of Bi-LSTM model

Metrics	Training	Validation
Accuracy	83.65%	69.41%
Loss	0.004	0.006

The performance of the Bi-LSTM model highlights its efficacy in capturing user preferences based on various input combinations. However, further analysis and experimentation are necessary to understand the factors contributing to the lower validation accuracy compared to the training accuracy. Future research can focus on enhancing the model's generalization capabilities and addressing potential limitations to improve the accuracy and reliability of travel destination predictions.

8.8 Conclusion

In conclusion, this research aimed to develop a recommender system for travel destinations using Malayalam travel reviews. The study utilized a Bi-LSTM model to capture the underlying patterns and relationships between various input variables, including travel type, mode of travel, location type, and location climate. The results demonstrated the effectiveness of the Bi-LSTM model in predicting travel destinations based on the provided inputs. During the training phase, the model achieved a high accuracy of 83.65%, successfully learning the complexities of the data. The validation phase further confirmed the model's ability to generalize its learned patterns, achieving a validation accuracy of 69.41%. These findings highlight the potential of the Bi-LSTM model in personalized travel recommendation systems.

By considering key input variables, the model can provide accurate and tailored travel destination suggestions to users. The integration of the Bi-LSTM architecture allows for the capture of contextual information from both past and future contexts, enhancing the model's capability to interpret travel patterns effectively. The developed system holds the potential for providing personalized and accurate travel destination recommendations, facilitating enhanced user experiences and satisfaction. By leveraging the power of deep learning techniques, such as the Bi-LSTM architecture, the study contributes to the field of recommender systems by demonstrating the applicability of the model in the context of Malayalam travel reviews.

While the results are promising, it is essential to address the observed difference between training and validation accuracies, indicating the scope for further improvements. Future research can focus on enhancing the model's generalization capabilities, exploring additional features, and incorporating user feedback to refine the travel recommendation system.

9 RS based on Deep Autoencoders

9.1 Introduction

Travel and tourism have emerged as critical drivers for cultural understanding, economic development, and personal exploration. In today's interconnected world, social media platforms, including Facebook travel groups, have grown into invaluable repositories of travel experiences and recommendations. These platforms allow travelers to pen down fascinating travelogues, share firsthand experiences, and offer insights that cater to different travel preferences. This research paper zeroes in on the Malayalam language, tapping into a rich collection of 13458 travelogues from travel blogs and Facebook's travel communities, to craft a personalized travel recommender system employing deep learning algorithm using autoencoders [88] and a set of machine learning algorithms.

The Malayalam language, native to the Indian state of Kerala and the Lakshadweep Islands, presents distinct challenges for text and speech processing. Its complex morphological structure, agglutinative nature, spelling inconsistencies, and the absence of benchmark datasets add layers of complexity to developing effective models for Malayalam text processing. In response to these challenges, this research meticulously shapes a structured dataset, encapsulating key features like Travel Type (TT), Travel Mode (TM), Location Type (LT), Location Climate (LC), Users (U), and specific destinations (L). The dataset serves as the backbone for the innovative personalized travel recommender model introduced in this experiment.

At the core of this methodology stands the autoencoder neural network, a powerful deep learning architecture designed to extract and compress essential patterns [89] from Malayalam travelogues. This novel approach enables the construction of compact and meaningful representations of travel data, facilitating various machine learning models' training. The work meticulously details the autoencoder's architecture, data preprocessing, training processes, and achieves an impressive validation accuracy of 95.84%. The significant contributions of this research include the development of a specialized autoencoder for Malayalam travel data, comparative analysis of several machine learning models such as logistic regression, decision tree classifier, SVM, random forest, KNN, SGD, and MLP, and achieving enhanced travel recommendation accuracy. By weaving these elements together, this research not only pioneers a path in Malayalam language processing but also promises to enhance travel experiences with personalized, culturally rich recommendations.

Key contributions of this approach include,

- *Development of an Autoencoder Model for Malayalam Travelogues*: This research marks the introduction of a unique autoencoder model tailored for processing Malayalam travelogues, facilitating unsupervised learning by capturing underlying patterns and features within the travel data.

- *Comprehensive Data Analysis with and without Compression*: The study incorporates detailed analysis utilizing autoencoder architecture, comparing results with and without data compression to understand the model's effectiveness.

- *Robust Evaluation of Compressed Data Performance*: A systematic evaluation of the autoencoder model's performance in compressing travel data demonstrates its practicality and efficiency for the chosen application.

- *Achievement of Enhanced Travel Recommendation Accuracy*: The encoded representations produced by the autoencoder have been instrumental in training various machine learning models, significantly enhancing the accuracy of travel recommendations.

- *Comparative Analysis of Diverse Machine Learning Models*: This research involves an extensive comparison of several machine learning algorithms, including logistic regression, decision tree classifier, SVM, random forest, KNN, SGD, and MLP. These models were trained using the encoded travel representations, offering insights into their relative performances.

- *Potential Extension to Other Indian Languages*: The methodology and insights gained from this research holds potential for adaptation and implementation in other low-resourced Indian languages, paving the way for further advancements in personalized recommendations across diverse linguistic landscapes.

9.2 Significance of Autoencoder in Recommendation Model

Autoencoders have emerged as a powerful tool in recommendation systems, making significant contributions to the field [137]. Their unique capability to perform dimensionality reduction allows them to capture essential characteristics of users' preferences without the noise often present in high-dimensional data. This compression technique not only emphasizes key patterns that drive users' choices but also filters out irrelevant information, leading to more accurate and personalized recommendations. Furthermore, as an unsupervised learning algorithm, autoencoders can discover underlying patterns and similarities without requiring labeled data, which is often a challenging aspect of building recommendation systems [96], [138].

The ability of autoencoders to capture complex non-linear relationships sets them apart from many traditional dimensionality reduction techniques. In recommendation systems, where relationships between users, items, and preferences are often intricate, this capability is paramount. The flexibility and adaptability of autoencoders allow them to be tailored to suit specific recommendation tasks, handling diverse types of data and objectives. This customization, combined with their efficiency in dealing with large datasets, makes them a suitable choice for modern, scalable recommendation systems.

In addition to the aforementioned benefits, autoencoders also provide enhanced robustness against noise and anomalies and can be integrated with collaborative filtering techniques to further refine recommendations. By learning from the core features of the data and combining user-item interactions with additional content features, autoencoders offer a nuanced understanding of user preferences[139]. The significant role they play in recommendation systems, thus, extends to improving both the quality and computational efficiency of recommendations, establishing them as an indispensable component in the field of personalized travel recommendations and beyond.

9.3 Structure of Autoencoder Algorithm

An autoencoder is a type of artificial neural network used for unsupervised learning of efficient coding, primarily for the purpose of dimensionality reduction and feature learning. The network consists of two main parts: the encoder, which compresses the input into a latent-space representation, and the decoder, which reconstructs the input data from this internal representation. Essentially, the network is trained to copy its input to its output, but it must learn this mapping by compressing the data through a narrow-hidden layer. This process forces the autoencoder to engage in data-specific learning, capturing relevant features in the compressed representation. The reduced dimensionality often makes the learned relationships in the data more tractable and is particularly useful for tasks like anomaly detection, denoising, and recommendation systems.

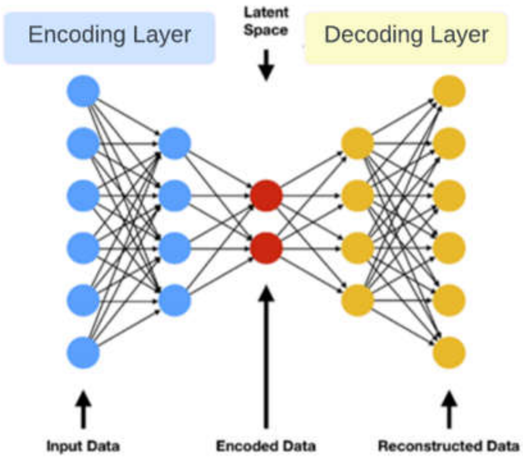


Figure 37 Structure of autoencoder network

The architecture of an autoencoder network as shown in figure 37, consists of three main components: the input layer, the hidden layer, and the output layer. The input layer receives the raw data and passes it to the hidden layer, which typically contains a smaller number of neurons, enforcing a compressed representation of the data; batch normalization and activation functions within the hidden layers further assist in efficient training by normalizing the input and introducing non-linearities. The hidden layer then connects to the output layer, aiming to reconstruct the original input from the compressed internal representation, thereby learning the salient features of the data that allow for such reconstruction.

9.4 Methodology

The methodology for travel recommendation using an autoencoder starts with data collection, where vast amounts of travel-related information are gathered from various sources, such as social media platforms[140]. The collected data then undergoes a series of preprocessing steps, including cleaning, normalization, and filtering, to ensure quality and consistency. Feature extraction is performed to identify and select the most relevant attributes, followed by dataset preparation, where the processed data is organized. The core of the methodology lies in the building and training of the autoencoder model, a deep learning technique that learns a compressed representation of the travelogue data, effectively capturing essential patterns and features. Figure 38 shows the steps involved in the autoencoder recommender model.

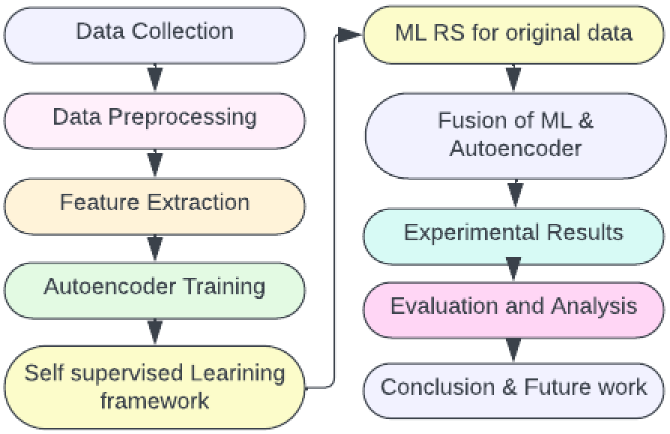


Figure 38 steps involved in autoencoder recommender model.

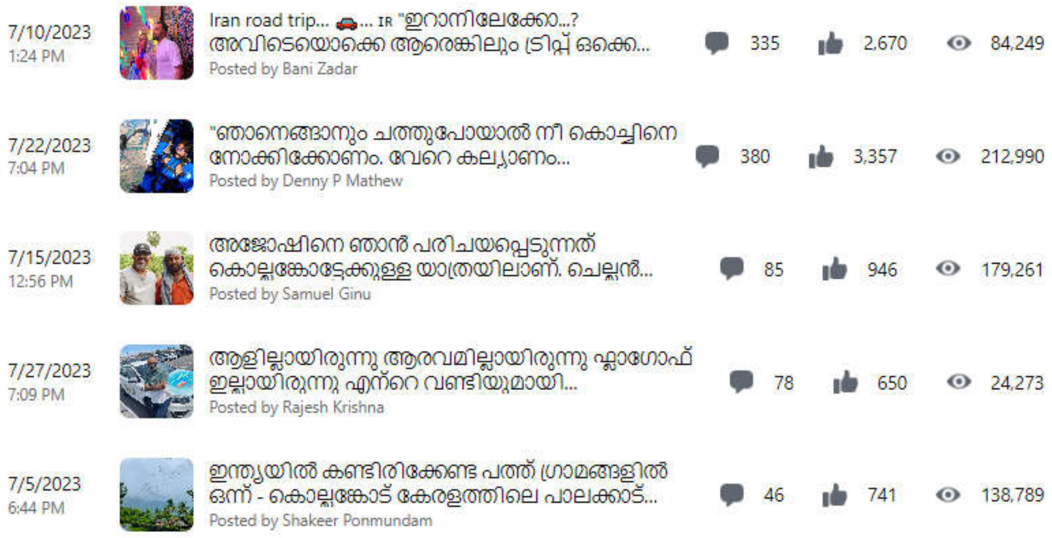
A self-supervised [141] learning framework [142], [143] is implemented to facilitate the training of the model without extensive labeled data [144]. The machine learning recommender system with the original data is then built, an ensembled model using machine learning algorithms, and the autoencoder model allows for enhanced travel recommendation accuracy [145]. The experimental results are obtained, thoroughly analysed, and compared to evaluate the overall performance of the system. The combination of the autoencoder's powerful data compression ability with various machine learning models provides a robust and effective solution to personalized travel recommendations, culminating in a system capable of delivering precise and tailored travel suggestions.

9.5 Data Collection

The data collection phase of this study was an intricate process that revolved around obtaining travel-related information from various online sources.

The primary focus was on 'Sanchari,' the largest Malayalam travel group on Facebook, as well as different travel blogs containing valuable reviews and travelogues. The information was gathered through a method that automatically extracts data from web pages.

Figure 39 illustrates the structure of a list of posts, containing travelogues in Malayalam, along with corresponding reactions, usernames, profile pictures, the total count of people who viewed the post, and the date and time it was posted. These posts are meticulously retrieved and compiled into a Spreadsheet, serving as the foundational base for both data collection and dataset preparation.



7/10/2023 1:24 PM	Iran road trip... 🚗... IR "ഇറാനിലേക്കോ...? അവിടെയൊക്കെ ആരെങ്കിലും ട്രിപ്പ് ഒക്കെ... Posted by Bani Zadar	335	2,670	84,249
7/22/2023 7:04 PM	"ഞാനെങ്ങാനും ചത്തുപോയാൽ നീ കൊച്ചിനെ നോക്കിക്കോണം. വേറെ കല്യാണം... Posted by Denny P Mathew	380	3,357	212,990
7/15/2023 12:56 PM	അജോഷിനെ ഞാൻ പരിചയപ്പെടുന്നത് കൊല്ലങ്കോട്ടേക്കുള്ള യാത്രയിലാണ്. ചെല്ലൻ... Posted by Samuel Ginu	85	946	179,261
7/27/2023 7:09 PM	ആളില്ലായിരുന്നു ആരവമില്ലായിരുന്നു ഫ്ലാഗോഫ് ഇല്ലായിരുന്നു എന്റെ വണ്ടിയുമായി... Posted by Rajesh Krishna	78	650	24,273
7/5/2023 6:44 PM	ഇന്ത്യയിൽ കണ്ടിരിക്കേണ്ട പത്ത് ഗ്രാമങ്ങളിൽ ഒന്ന് - കൊല്ലങ്കോട് കേരളത്തിലെ പാലക്കാട്... Posted by Shakeer Ponmundam	46	741	138,789

Figure 39 Structure of posts in Facebook group admin panel insight

Figure 40 shows the admin insight[146] graph of the Facebook group which focuses on the growth and engagement of users and interactions. It describes the total number of posts in the scheduled interval, details of the travelogue, posted time, URL of the post, username, Facebook profile link, total reactions, comments, likes, and shares.

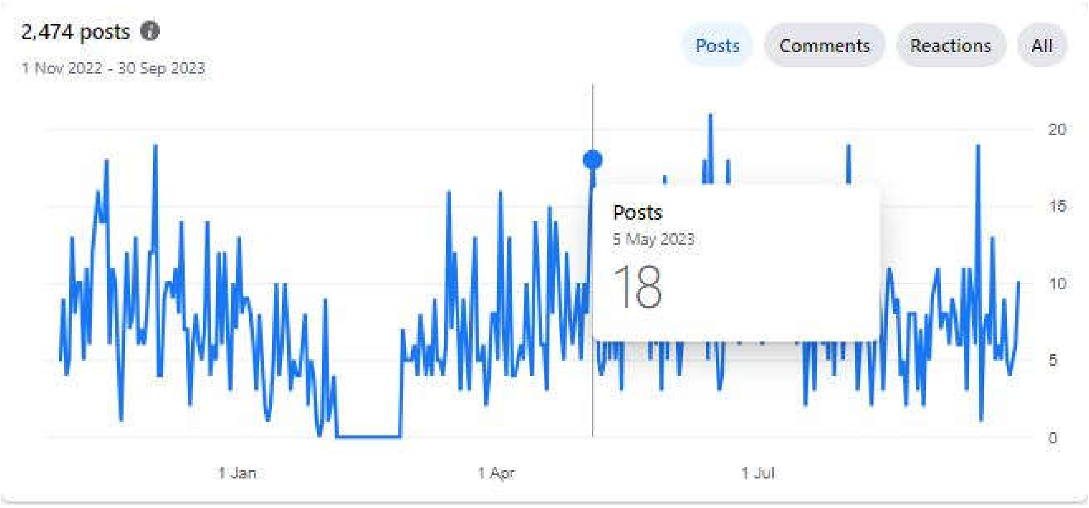


Figure 40 Admin insight of growth and engagements of Facebook group.

The collected data was inherently unstructured, noisy, and inconsistent. It included diverse write-ups from various users, reflecting a wide array of individual experiences and opinions. To maintain structure and organization, each travelogue was stored in a separate individual file, named after the respective traveler. This method ensured easy retrieval and management of data, facilitating its future use. Additionally, the system was designed to accommodate input from new users, seamlessly incorporating their unique travel-related text into the dataset.

The initial dataset contained a substantial amount of 13,458 unstructured travelogues. Rigorous preprocessing was carried out to transform this unorganized information into a structured tabular format.

9.6 Travelogue Preprocessing

Text preprocessing forms an essential part of readying the unstructured travelogues for the construction of personalized travel recommender system. This process starts with tokenization, splitting the text into individual words or tokens, allowing for more precise manipulation. Next, the data is cleaned to eliminate irrelevant characters, symbols, or special elements, making the text more suitable for subsequent examination. Alongside this, stopwords—common words without substantial meaning—are removed to minimize noise in the data, and some examples of Malayalam stopwords are provided in Table 20. To maintain language uniformity, either stemming or lemmatization is applied to reduce words to their root or essential forms, helping in grouping different forms of a word. Additionally, specific words are mapped to standardized representations from a predefined dictionary, resolving synonyms and related words, and thereby cutting down ambiguity. The Part of Travelogue Tagger (POTT) is used to annotate and identify travel-related aspects like locations, activities, and means of transport, incorporating domain-specific knowledge into the process. The final tagged data is then systematically saved in CSV or spreadsheet formats, ensuring ease of access and further analysis.

Table 20 Sample Stop words, Tokens, Root word in Malayalam.

Stop words		**Tokens and Root word in Malayalam and English**		
Malayalam	**English**	**Tokens**	**Root word**	**English**
അത്	that	പോയിരിക്കും	പോകുക	Go
അങ്ങനെ	so	കുടുംബമായി	കുടുംബം	Family
മതി	enough	കാറിലേക്ക്	കാർ	Car
എങ്കിൽ	if	തണുപ്പിന്റെ	തണുപ്പ്	Cool
മറ്റു	other	മരത്തിന്റെ	മരം	tree

9.7 Feature Extraction

The feature extraction phase is crucial in converting lengthy, noisy, and code-mixed unstructured travelogues into a coherent and valuable dataset. Within an intricate preprocessing pipeline, every token in the travelogues is labeled utilizing specially developed Part of Travelogue Tagger (POTT), a tool fashioned specifically for this research. This tagging operation classifies each token into one of the identified feature classes, such as Travel Type (TT), Travel Mode (TM), Location Climate (LC), Location Type (LT), and particular destinations (L). Through marking each token with the corresponding feature class, developed a systematic dataset encapsulating vital information tied to varied travel experience facets. Table 21 illustrates the fundamental structure of the dataset with these essential features. This information serves as the base for instructing the personalized travel recommender model, thereby equipping it to furnish precise and customized suggestions grounded on users' tastes and inclinations. Utilizing the POTT tool guarantees that the extracted elements are consistent with the context of travelogues in the Malayalam language, which amplifies the model's comprehension of the intricacies and subtleties of travel-related information. In summation, the feature extraction phase eases the transformation of unstructured travelogues into an insightful and methodically organized dataset, fortifying model's ability to provide tailored and pertinent travel advice for users.

Table 21 Structure of dataset prepared from the unstructured travelogue.

Sl. No	Climate	Travel_type	Location_type	Travel_mode	Locations
1	മഴ	തനിയെ	സാഹസികം	കാർ	മണാലി
2	വെയിൽ	സോളോ	സിറ്റി	ബസ്	ദൽഹി
3	മഞ്ഞ്	കുടുംബം	ഹൈറേഞ്ച്	ബൈക്ക്	ഇടുക്കി
4	തണുപ്പ്	കൂട്ടുകാർ	സിറ്റി	തീവണ്ടി	കോഴിക്കോട്
5	ചൂട്	ഫ്രണ്ട്സ്	പ്രകൃതി	വിമാനം	കശ്മീർ
6	സമ്മർ	ചങ്ങാതി	സാഹസികം	ഫ്ലൈറ്റ്	ലഡാക്ക്
7	വിന്റർ	സഹപ്രവർത്തകർ	പ്രകൃതി	കപ്പൽ	കോവളം
7	ശൈത്യം	ഓഫീസ്	സാഹസികം	റോഡ്	ഗോവ
8	വസന്തം	ഭാര്യ	ഹൈറേഞ്ച്	കടൽ	വയനാട്
9	മഴ	ഭർത്താവ്	ചരിത്രം	ബോട്ട്	കാസർഗോഡ്
10	വെയിൽ	സഹോദരങ്ങൾ	പ്രകൃതി	സൈക്കിൾ	ആലപ്പുഴ
11	മഞ്ഞ്	അമ്മ	സിറ്റി	നടന്ന്	ബാംഗ്ലൂർ
12	തണുപ്പ്	അച്ഛൻ	തീർത്ഥാടനം	സ്കൂട്ടർ	മൈസൂർ

9.8 Modeling Autoencoder Architecture

The custom autoencoder architecture utilized in this research consists of two main components: the encoder and the decoder, each performing a specific role in processing the pre-processed Malayalam travelogue data.

9.8.1 Encoder

The encoder's function is to transform the high-dimensional input features derived from the Malayalam travelogues into a condensed latent space. It accomplishes this by utilizing a series of layers (often including fully connected layers, batch normalization, and activation functions) that systematically reduce the dimensionality of the data.

This reduction captures the essential information and patterns within the data in a smaller and more manageable form. The encoded representation serves as a form of data compression, allowing for more efficient processing while preserving critical information about the original travelogues. The encoding part of the architecture consists of two fully connected layers. The first layer utilizes a dense configuration (4x8), followed by batch normalization (8x8) and LeakyReLU activation. This layer sequence compresses the input data and identifies meaningful patterns. The second layer repeats the configuration, further condensing the data into a compact form. The encoder's output serves as a reduced-dimensional representation of the original input, capturing essential characteristics.

9.8.2 Decoder

The decoding part of the architecture mirrors the encoder but works in the opposite direction. The first layer within the decoder includes a dense configuration, batch normalization, and LeakyReLU activation, serving to expand the compressed data. The second layer follows the same pattern, further enhancing the expanded representation.

The final output layer, consisting of a dense layer with linear activation, reconstructs the original input data, reflecting the information captured in the encoded form. Once the data is compressed into the latent space by the encoder, the decoder's task is to reconstruct the original data from this compact form. It essentially mirrors the architecture of the encoder but in reverse, progressively expanding the compressed information through successive layers.

The aim of this reconstruction is not merely to recreate the original data but to understand and learn the fundamental structures and relationships within the travelogues. By comparing the original input with the reconstructed output, the autoencoder's training process can fine-tune the weights and biases of the network, optimizing the representation in the latent space. The construction of the Autoencoder without compression is given in Figure 41.

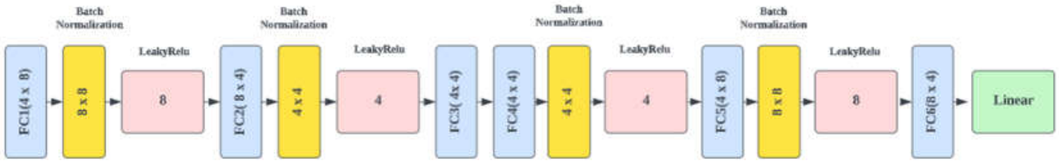


Figure 41 Construction of Autoencoder without compression

9.8.3 Construction and Training of the Autoencoder

The model construction employs encoded data, with the encoder comprising two dense layers incorporating LeakyReLU activation and batch normalization. The dimensionality is successively reduced, with the bottleneck layer having half the neurons of the input, forming the compressed representation. The decoder is constructed to mirror the encoder but expands the dimensions. The output layer utilizes a linear activation function to reflect the original input features. The model is trained through 50 epochs with a batch size of 16, minimizing the difference between the original input and the reconstructed output. An additional encoder model allows for the extraction of the compressed data representation.

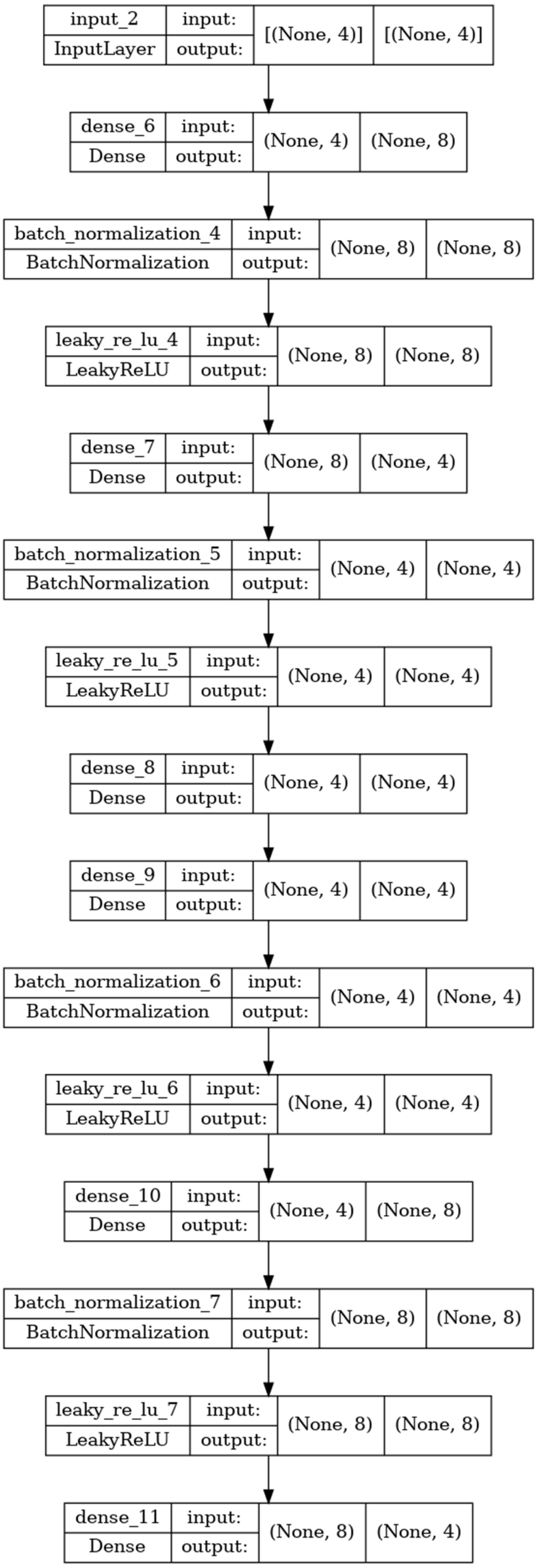

input_2 | InputLayer | input: | output: | [(None, 4)] | [(None, 4)]
dense_6 | Dense | input: | output: | (None, 4) | (None, 8)
batch_normalization_4 | BatchNormalization | input: | output: | (None, 8) | (None, 8)
leaky_re_lu_4 | LeakyReLU | input: | output: | (None, 8) | (None, 8)
dense_7 | Dense | input: | output: | (None, 8) | (None, 4)
batch_normalization_5 | BatchNormalization | input: | output: | (None, 4) | (None, 4)
leaky_re_lu_5 | LeakyReLU | input: | output: | (None, 4) | (None, 4)
dense_8 | Dense | input: | output: | (None, 4) | (None, 4)
dense_9 | Dense | input: | output: | (None, 4) | (None, 4)
batch_normalization_6 | BatchNormalization | input: | output: | (None, 4) | (None, 4)
leaky_re_lu_6 | LeakyReLU | input: | output: | (None, 4) | (None, 4)
dense_10 | Dense | input: | output: | (None, 4) | (None, 8)
batch_normalization_7 | BatchNormalization | input: | output: | (None, 8) | (None, 8)
leaky_re_lu_7 | LeakyReLU | input: | output: | (None, 8) | (None, 8)
dense_11 | Dense | input: | output: | (None, 8) | (None, 4)

Figure 42 Autoencoder architecture without compression

The custom-designed autoencoder model, illustrated in Figure 42, focuses on the unique task of compressing textual data in the Malayalam language, utilizing pre-processed travelogues. In the encoder phase, the model reduces the dimensionality through two dense layers with leaky ReLU activation and batch normalization, retaining vital language features.

The bottleneck layer, containing half the neurons of the input, forms a compressed representation, encapsulating essential linguistic elements. The decoder mirrors the encoder, reconstructing the original data, and is optimized using Adam with the MSE loss function and accuracy as the metric. MinMaxScaler is employed to enhance efficiency and handle variations in textual input. This architecture's significance lies in its ability to understand complex linguistic features such as syntax, semantics, and context specific to the Malayalam language.

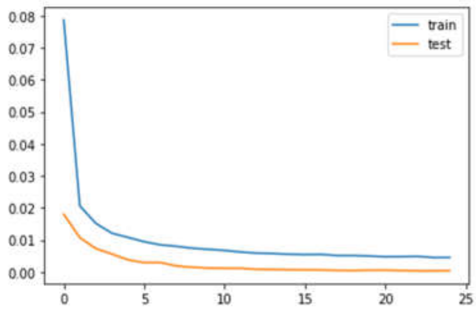
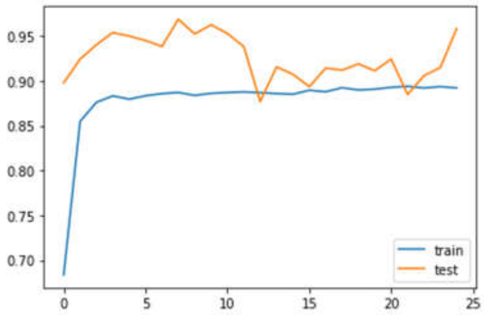
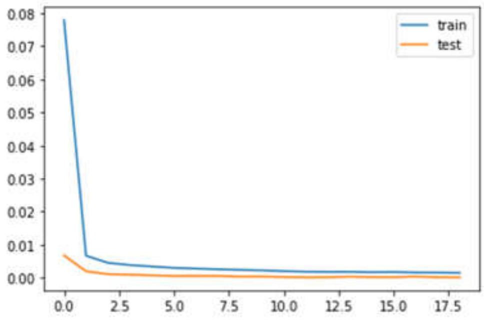
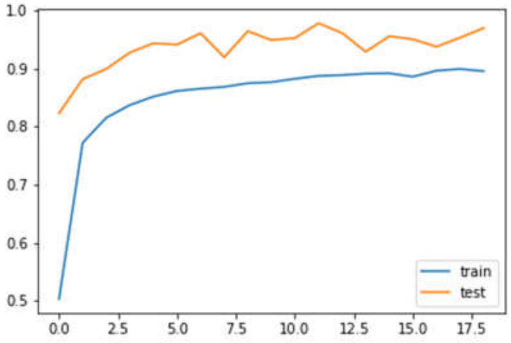
9.8.4 Performance Evaluation Of Autoencoder With And Without Compression

Table 22 Performance evaluation of autoencoder

Sl. No	Methodology	Accuracy
1	Encoder Without Compression	95.84 %
2	Encoder With Compression	96.96 %

Table 22 presents a comparison between two methodologies: using an encoder without compression and using an encoder with compression. Remarkably, the encoder with compression yields a slightly higher accuracy rate of 96.96%, compared to the 95.84% accuracy obtained without compression. This analysis indicates that the compression technique implemented within the encoder not only effectively reduces the dimensionality of the data but also enhances the model's performance. This suggests that the specific compression technique employed in this work can retain significant features and information needed for the task, leading to an improvement in accuracy. Therefore, the application of compression within the encoder can be considered a valuable approach in the context of this specific study.

Table 23 Performance evaluation of autoencoder models

Autoencoder without compression	
Loss – Training loss and validation	Training accuracy and validation accuracy
	
Autoencoder with compression	
Loss – Training loss and validation	Training accuracy and validation accuracy
	

The analysis of the Encoder with Compression methodology reveals a remarkable performance, achieving an accuracy of 96.96%. This method involves the autoencoder further reducing the dimensionality of the input data, leading to a more condensed and compressed representation in the bottleneck layer. Unlike the uncompressed methodology, this approach focuses on retaining vital information while eliminating less important details. This precise compression results in a more efficient representation of the data, with higher accuracy showcasing the effectiveness of this approach. Table 23 illustrates the accuracy and loss curves for both the compressed and uncompressed algorithms, providing a detailed numerical comparison.

Figure 43 visually complements this analysis, offering a clear diagrammatic representation of the results and underlying the superiority of the compression methodology in this context.

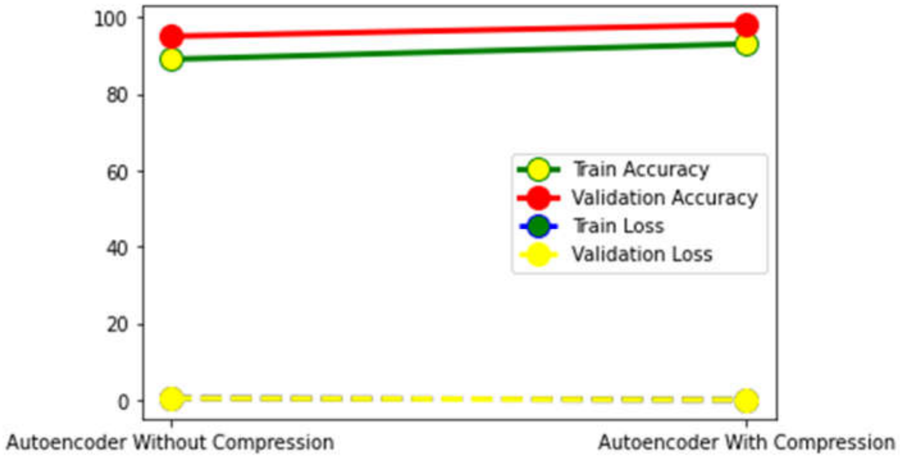


Figure 43 Autoencoder - compressed and not.

9.9 Ensembled Model Using Autoencoder With ML Models

The fusion of autoencoder with different machine learning approaches as shown in figure 44, offers a robust method for travel recommendation, leveraging the strengths of both deep learning [147] and traditional algorithms. In the first stage, the autoencoder's encoder component is used to reduce the dimensionality of the input data, creating a compact and representative feature set. By capturing essential information from the textual travelogues in Malayalam, the encoder produces a compressed representation that retains vital characteristics related to travel experiences, destinations, and preferences. This condensed form of data serves as a foundation, providing a more manageable and expressive input for subsequent machine learning models.

The second stage involves applying various machine learning algorithms to the compressed data. Techniques such as Support Vector Machines (SVM) [148], Random Forest[149], Gradient Boosting, and K-Nearest Neighbors (K-NN)[150] can be employed, each bringing unique strengths and methodologies to the task. The combination of the autoencoder with these methods allows the model to combine deep learning's ability to automatically extract relevant features with traditional machine learning algorithms' efficiency and interpretability. The result is an enhanced travel recommender system that can provide personalized and accurate recommendations, leveraging insights from both neural networks and machine learning techniques. This hybrid approach not only increases the model's performance but also ensures its adaptability and applicability across various travel-related contexts and user preferences.

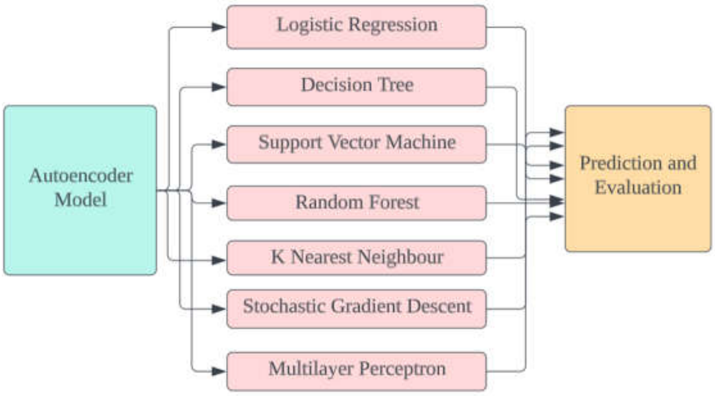


Figure 44 Fusion of ML- Autoencoder Model Architecture

9.9.1 Logistic Regression

In the context of this work, logistic regression serves as one of the machine learning techniques used in conjunction with the autoencoder to create a powerful travel recommender system. The compressed representation of the travelogues, obtained from the encoder part of the autoencoder, acts as the input to the logistic regression model[151]. This approach simplifies the high-dimensional Malayalam language data into a format that's more amenable to modelling. Logistic regression, being a statistical method for analysing a dataset in which there are one or more independent variables that determine an outcome, is employed to predict the probability of a particular travel preference or category.

It aligns well with the project's goal of personalizing travel recommendations, as it can handle binary classification problems and provide probabilities that can be translated into actionable insights. The combination of autoencoder and logistic regression leverages the deep learning capabilities to handle complex language processing tasks with a statistical method known for its interpretability and efficiency, resulting in a nuanced and targeted recommender system. The structure of Logistic regression is represented in figure 45.

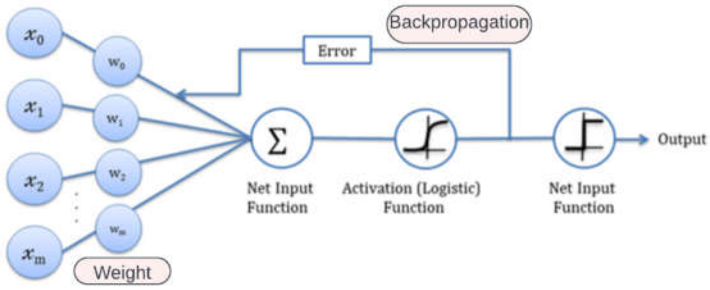


Figure 45 Diagrammatic representation of Logistic regression.

The performance metrics for the Logistic Regression model utilized in this research for personalized travel recommendations based on Malayalam language travelogues. With an accuracy of 86.96%, the model demonstrates a robust ability to predict travel preferences correctly. Precision, which measures the proportion of true positive predictions among the total predicted positives, is 90%, indicating a high level of reliability in its positive classifications. The F1 Score, which balances the precision and recall, is at 86%, further testifying to the model's solid performance. The recall of 86%, representing the proportion of actual positives that were correctly classified, also signifies that the model is adept at recognizing the relevant data points. These metrics collectively affirm that the fusion of autoencoder with Logistic Regression in this context is highly effective, enabling precise and trustworthy personalized travel recommendations.

9.9.2 Support Vector Machine

The Support Vector Machine (SVM) plays a critical role in classifying travel-related features. By mapping the input data into a higher-dimensional space, SVM constructs a hyperplane that optimally separates the classes, thus aiding in the discernment of travel patterns and preferences. This methodology's integration, in conjunction with autoencoders, allows the model to capture complex relationships and nuances within the travelogues, enhancing the overall accuracy and relevance of the travel recommendations provided.

The Support Vector Machine (SVM) model in this study achieved an accuracy of 83%, with precision as 84 %, F1 Score, and recall all similarly positioned at 83 %. This performance illustrates the model's balanced classification ability in the context of Malayalam travelogues, effectively capturing and differentiating travel-related features, although there may still be room for optimization to enhance its predictive capacity.

9.9.3 Decision Tree

Decision Tree algorithm was applied as part of the fusion of machine learning approaches with the autoencoder for travel recommendation in the Malayalam language. Utilizing a tree-like graph of decisions, this algorithm facilitated the understanding of the complex relationships between different travel-related features extracted from the travelogues. Its hierarchical structure and comprehensible visualization made it a valuable tool for both prediction and interpretability within the personalized travel recommender system, providing insights into the underlying patterns and preferences in the data.

The Decision Tree model achieved commendable results in the context of this travel recommendation project, demonstrating an accuracy of 87%, a precision of 85%, and an F1 Score and Recall of 85% each. These metrics reflect the model's ability to accurately classify and predict travel preferences, providing a solid basis for personalized recommendations, and its balanced performance in both precision and recall indicates a robust alignment with the specific nuances of the Malayalam language travelogues.

9.9.4 Random Forest

The Random Forest model, an ensemble learning method comprising multiple decision trees, was implemented in this study to boost the performance and reliability of the travel recommender system. By combining the predictions from various decision trees, the Random Forest approach provided a more robust and generalized model, effectively handling the complexity and intricacy of the Malayalam language travelogues. The integration of Random Forest within the recommendation framework contributed to enhanced stability and accuracy, offering a more nuanced understanding of user preferences and interests related to travel experiences. Graphical representation of Random Forest model is shown in figure 46.

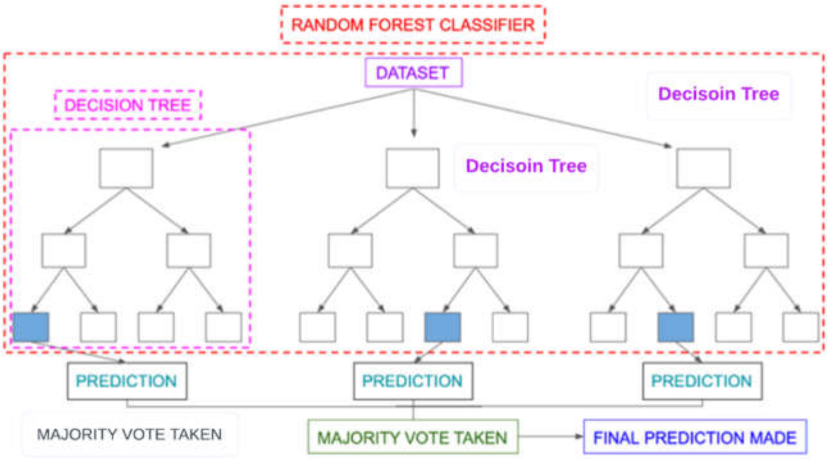


Figure 46 Graphical representation of Random Forest model

In this work, the Random Forest model exhibited an accuracy of 86.66%, a precision of 87%, an F1 score of 86%, and a recall rate of 87%. These metrics demonstrate the model's solid performance in accurately classifying and understanding travel preferences within the Malayalam travelogues. The balanced scores across these key performance indicators underline the Random Forest's efficiency in capturing the nuanced relationships in the data, thereby confirming its suitability and effectiveness for this specific application in the personalized travel recommender system.

9.9.5 K-Nearest Neighbour

The K-Nearest Neighbours (KNN) algorithm was utilized in this research to further explore the Malayalam travelogues for personalized recommendations. By leveraging its non-parametric and instance-based learning nature, the KNN model analysed similarities within the dataset, efficiently grouping travel preferences based on specific features such as location, travel mode, and travel type. This utilization of the KNN method has proven instrumental in enhancing the recommender system's capability to provide more accurate and contextually relevant suggestions, tailored to individual users' tastes and preferences. The K-Nearest Neighbours (KNN) model was implemented in the study, achieving an accuracy of 86%, precision of 88%, F1 Score of 86%, and recall of 88%. These metrics highlight the efficacy of KNN in identifying and categorizing travel preferences within the Malayalam travelogues.

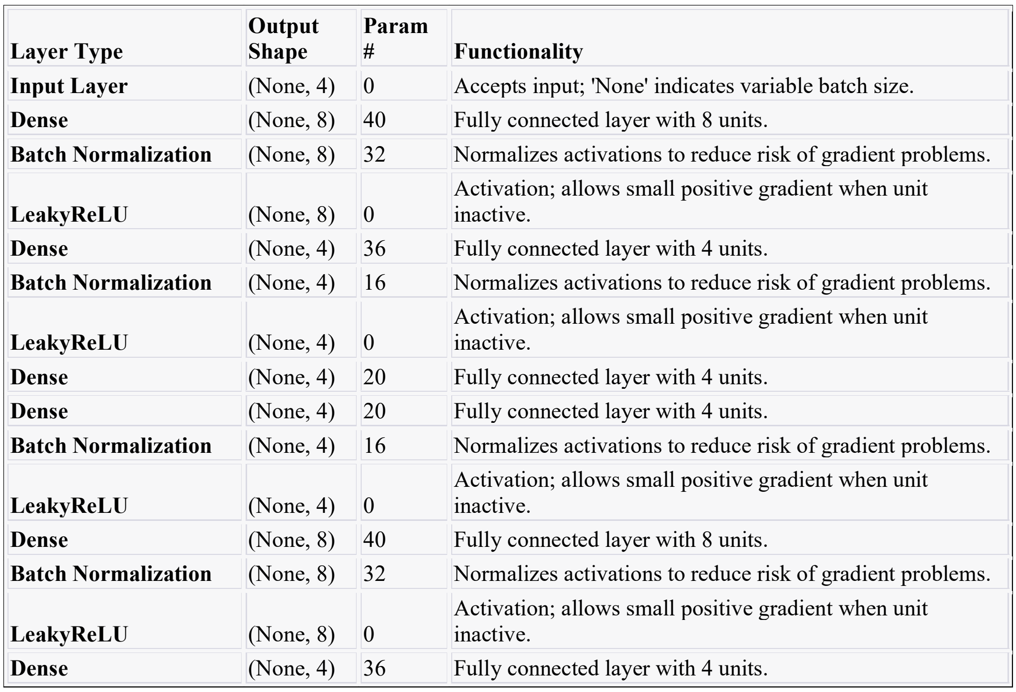
9.9.6 Stochastic Gradient Descent

Stochastic Gradient Descent (SGD) was utilized as an optimization method, contributing to the efficiency of the model training process. By incrementally updating the model parameters using a subset of the Malayalam travelogue data, SGD enabled faster convergence and fine-tuning of the model. The application of SGD in this work proved to be a strategic choice, facilitating optimal performance in the development of the personalized travel recommender system. Stochastic Gradient Descent (SGD) was employed, achieving an accuracy of 85%, a precision of 87%, an F1 Score of 85%, and a recall of 85%. These metrics reflect the model's balanced performance, with SGD playing a vital role in the optimization process, guiding the model towards an optimal solution efficiently. The results further demonstrate the effectiveness of SGD in handling the complex and morphologically rich Malayalam language data, contributing to the success of the recommendation system.

9.9.7 Multi-Layer Perceptron

The Multi-Layer Perceptron (MLP) was also integrated into the personalized travel recommender system, designed specifically to work with the Malayalam language data. It was instrumental in capturing the complex relationships and patterns within the data, resulting in improved modelling of user preferences and travel experiences. The application of MLP within the system illustrates the power of neural network architectures in processing linguistically rich and intricate data, contributing to the model's ability to make accurate and personalized travel recommendations.

The utilization of a Multi-Layer Perceptron (MLP) in the personalized travel recommender system yielded an accuracy of 86.66%, with a precision of 90%, an F1 Score of 87%, and a recall rate of 87%. The MLP's architecture and training process were tailored to the Malayalam language, enabling the model to effectively discern complex patterns in the data. The achieved results highlight the MLP's effectiveness in delivering personalized travel recommendations, balancing various performance metrics, and exemplifying its role in enhancing the overall system. The construction of neural network for deep autoencoder is represented in Figure 47.



Layer Type	Output Shape	Param #	Functionality
Input Layer	(None, 4)	0	Accepts input; 'None' indicates variable batch size.
Dense	(None, 8)	40	Fully connected layer with 8 units.
Batch Normalization	(None, 8)	32	Normalizes activations to reduce risk of gradient problems.
LeakyReLU	(None, 8)	0	Activation; allows small positive gradient when unit inactive.
Dense	(None, 4)	36	Fully connected layer with 4 units.
Batch Normalization	(None, 4)	16	Normalizes activations to reduce risk of gradient problems.
LeakyReLU	(None, 4)	0	Activation; allows small positive gradient when unit inactive.
Dense	(None, 4)	20	Fully connected layer with 4 units.
Dense	(None, 4)	20	Fully connected layer with 4 units.
Batch Normalization	(None, 4)	16	Normalizes activations to reduce risk of gradient problems.
LeakyReLU	(None, 4)	0	Activation; allows small positive gradient when unit inactive.
Dense	(None, 8)	40	Fully connected layer with 8 units.
Batch Normalization	(None, 8)	32	Normalizes activations to reduce risk of gradient problems.
LeakyReLU	(None, 8)	0	Activation; allows small positive gradient when unit inactive.
Dense	(None, 4)	36	Fully connected layer with 4 units.

Figure 47 Configuration of deep autoencoder with layer details

9.10 Evaluation of Performance Matrix

As shown in Table 24, the evaluation of performance observed after experiment and analysis of various machine learning models used in the study offers insights into their performance and suitability for the personalized travel recommender system. Logistic Regression shines with a top Precision score of 90% and strong Accuracy at 86.96%, making it a standout for predicting specific travel recommendations. MLP also shows excellent Precision at 90%, with an overall balanced performance, highlighting its versatility.

Decision Tree, although slightly lagging in Precision, offers consistent results across all metrics, suggesting its robustness. Random Forest and KNN exhibit reliable prediction capabilities, with Accuracy levels above 86%, demonstrating their efficiency in handling the complex dataset. Conversely, SVM registers the lowest Accuracy of 83%, indicating some challenges in capturing the intricacies of the Malayalam travelogues. SGD, although performing reasonably well, shows slightly inconsistent results across different metrics. Overall, the analysis emphasizes the importance of choosing the right model, considering the specific characteristics of the data and research objectives. The results underscore the strengths of Logistic Regression and MLP in providing accurate and precise travel recommendations, with their strong performance in essential evaluation metrics, setting them apart in this context.

Table 24 Comparative analysis of ML - Autoencoder models

<table>
<tr><th>Sl.No.</th><th>Methodology</th><th>Accuracy</th><th></th><th></th><th></th></tr>
<tr><td>1.</td><td>Encoder Without Compression</td><td>95.84</td><td></td><td></td><td></td></tr>
<tr><td>2</td><td>Encoder With Compression</td><td>96.96</td><td></td><td></td><td></td></tr>
<tr><th colspan="6">Machine Learning Models with Input as Autoencoder model with compression</th></tr>
<tr><th>Sl.No.</th><th>ML Model</th><th>Accuracy</th><th>Precision</th><th>F1 Score</th><th>Recall</th></tr>
<tr><td>1</td><td>Logistic Regression</td><td>86.96</td><td>90</td><td>86</td><td>86</td></tr>
<tr><td>2</td><td>Decision Tree</td><td>87</td><td>85</td><td>85</td><td>85</td></tr>
<tr><td>3</td><td>SVM</td><td>83</td><td>84</td><td>83</td><td>83</td></tr>
<tr><td>4</td><td>Random Forest</td><td>86.66</td><td>87</td><td>86</td><td>87</td></tr>
<tr><td>5</td><td>KNN</td><td>86</td><td>88</td><td>86</td><td>88</td></tr>
<tr><td>6</td><td>SGD</td><td>85</td><td>87</td><td>85</td><td>85</td></tr>
<tr><td>7</td><td>MLP</td><td>86.66</td><td>90</td><td>87</td><td>87</td></tr>
</table>

9.11 Discussion and Comparative Analysis

Among the different algorithms evaluated, MLP leads with the highest F1 Score of 87%, symbolizing an equilibrium between precision and recall. The remaining models exhibit F1 Scores within the range of 83% to 86%, reflecting their overall balanced performance. KNN, with a recall score of 88%, excels in correctly identifying positive instances, while SVM falls short with the lowest recall of 83%. This comparative analysis of accuracy, precision, recall, and F1 Score illustrates the distinct performances of each methodology and the various models. These detailed observations are collectively represented in Figure 48.

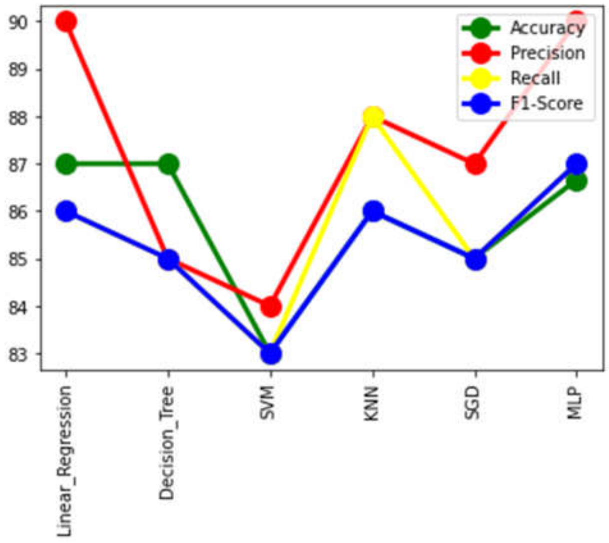


Figure 48 Comparative analysis of Machine learning algorithms

9.12 Conclusion

In the conclusion of this chapter, the significant contributions of the research in building a personalized travel recommender system for the Malayalam-speaking community have been thoroughly summarized and highlighted. The innovative approach of utilizing an autoencoder, in conjunction with various machine learning models, has enabled the extraction and compression of essential features from unstructured Malayalam travelogues. The Encoder with Compression methodology, achieving an impressive accuracy of 96.96%, stands out as an effective method that retains vital information without losing the unique characteristics of the Malayalam language. The comparative analysis of different models provides valuable insights, emphasizing the potential and applicability of this work in enhancing user-centric travel recommendations.

The research marks a pioneering effort in Malayalam travel recommendation, paving the way for future studies and improvements. It demonstrates how personalized travel recommendations can be generated from complex and culturally rich data, providing meaningful suggestions that resonate with individual preferences. The findings of this chapter contribute to the evolving field of personalized recommender systems and emphasize the importance of linguistic understanding in the development of efficient and nuanced models. This work serves as a strong foundation and an inspiration for researchers aiming to explore and innovate in the realm of travel planning and recommendation in regional languages.

10 Results and Discussions

Experiment 1 - The Rule-Based Cosine Similarity RS model.

This model managed to achieve an accuracy rate of 75% for its primary list of recommendations and an impressive 80% for the secondary list. Despite its relatively straightforward approach, this model demonstrated decent performance.

Table 25 Experimental Result from Rule Based Model

Recommendation Type	Average testing count	Correct Recommendations	Total Recommendations	Accuracy
Primary list	50	3	4	75%
Secondary list	50	4	5	80%

Based on Travel DNA, Location DNA and feature vector mapping, the cosine similarity measure calculate the most significant destinations to users based on their past travel histories and preferences. The data received through prompt window will pass to user's clusters and destination cluster to compute the best match and the suggestions are passed through two recommendation lists, primary list and secondary list. Implementation and model configuration details given in Chapter 6.

Experiment 2 – RS based on Clustering techniques.

The use of clustering techniques like Collaborative Filtering based K-Means Clustering and Content Based Filtering using Hierarchical Agglomerative Clustering has shown enhanced performance in the travel recommender system. With a remarkable 91% of suggested destinations featuring in the top three positions when using K-means clustering and 85% using hierarchical agglomerative clustering, these models exhibit a high degree of precision. Implementation and model configuration details given in Chapter 7.

The corresponding F1 score measures further validate their effectiveness, standing at 92.04% and 84.25% respectively, making them some of the most reliable approaches examined.

Table 26 Experimental result from Clustering techniques

Methodology	Accuracy	F1-Score	Precision	Recall
Collaborative Filtering Using K-Means Clustering	91.01%	92.04%	91.5%	92.6%
Content Filtering Based Hierarchical Agglomerative Clustering	85.01%	84.25%	84.15%	84.35%

Experiment 3 –The RS model based on Bi-LSTM neural network.

This model excelled during the training phase with a high accuracy of 83.65% from comparing several combinations of values continuous hyper parameter tuning. The result obtained when the optimizer as adam, loss function opted as mean squared error and epoch as 1500. Other combinations are tested but didn’t give better result as this deep network. However, the validation phase saw a drop to 69.41%, which signifies that while the model is capable of learning complex relationships in the training data, it has some limitations in generalizing these relationships to new, unseen data. Validation loss observed as 0.006 and training loss as 0.004. Despite the decrease in validation accuracy, the model's strong training performance cannot be overlooked, suggesting that with some refinements, it can become even more reliable. Implementation and model configuration details given in Chapter 8.

Table 27 Experimental result from Bi-LSTM model

	Optimizer	Loss function	epoch	Train Accuracy	Val. accuracy
Phase I	SGD	Categorical cross entropy	1000	46.76 %	43.87 %
			1500	56.87 %	51.63 %
Phase II	Adam	Mean squared error	800	66.71 %	58.14 %
			1500	**83.65 %**	**69.41 %**
			2000	80.01 %	63.17 %
Phase III	RMS_prop	Cat cross entropy	800	71.31 %	66.56 %
		MSE	1500	74.03 %	67.65 %

The validation accuracy indicates that the model can effectively apply its learned patterns to new combinations of input variables, thereby providing reliable travel destination recommendations.

These experimental metrics demonstrate that the model has effectively grasped the underlying patterns and connections between the input variables and the associated travel destinations. The high training accuracy implies that the model adeptly captures the intricacies of the data and excels in predicting destinations based on the given input.

Experiment 4 - The RS using deep Autoencoder architecture.

The model set a new benchmark by achieving an astounding accuracy rate of 96.96% with compression, compared to 95.84% without compression. The model's performance indicates that it is not only efficient but also highly accurate, making it a compelling choice for future work in this area.

Table 28 Result from deep Autoencoder architecture.

Sl. No	Methodology	Accuracy
1	Encoder Without Compression	95.84 %
2	Encoder With Compression	96.96 %

This method involves the autoencoder further reducing the dimensionality of the input data, leading to a more condensed and compressed representation in the bottleneck layer. Unlike the uncompressed methodology, this approach focuses on retaining vital information while eliminating less important details. This precise compression results in a more efficient representation of the data, with higher accuracy showcasing the effectiveness of this approach. Implementation and model configuration details given in Chapter 9.

In the model construction, the encoder consists of two dense layers that incorporate LeakyReLU activation and batch normalization. Progressively reduced the dimensionality, and the bottleneck layer has half the neurons of the input, creating a compressed representation. The decoder is designed to mirror the encoder but expands the dimensions. The output layer uses a linear activation function to match the original input features. Then trained the model for 50 epochs with a batch size of 16, aiming to minimize the difference between the original input and the reconstructed output. Additionally, an encoder model that used to extract the compressed data representation.

Experiment 5 – Ensembled model of Autoencoder and ML algorithms.

While performing an Ensembled model of recommendation model using Autoencoder with different ML algorithms, it is shown that Logistic

Regression shines with a top Precision score of 90% and strong Accuracy at 86.96%, making it a standout for predicting specific travel recommendations.

Table 29 Result from Ensembled model of Autoencoder and ML algorithms

Machine Learning Models with Input as Autoencoder model with compression					
Sl.No.	ML Model	Accuracy	Precision	F1 Score	Recall
1	Logistic Regression	**86.96**	**90**	86	86
2	Decision Tree	87	85	85	85
3	SVM	83	84	83	83
4	Random Forest	86.66	87	86	87
5	KNN	86	88	86	88
6	SGD	85	87	85	85
7	MLP	**86.66**	**90**	87	87

MLP also shows excellent Precision at 90%, with an overall balanced performance. Decision Tree, although slightly lagging in Precision, offers consistent results across all metrics, Random Forest and KNN exhibit reliable prediction capabilities, with Accuracy levels above 86%, demonstrating their efficiency in handling the complex dataset. Conversely, SVM registers the lowest Accuracy of 83%, indicating some challenges in capturing the intricacies of the Malayalam travelogues.

Experiment 6 – Opinion Mining from the Primary crowdsourced Data.

Based on the data collected through google from, the primary and authentic data undergo a set of analysis of users travel taste and preferences. A study conducted with the objectives to identify the most visited tourist destinations by Keralite's. To determine the travelling behaviour of people based on the Gender factor, to analyse the travelling behaviour of people based on the Age factor and to report most rated/ranked destinations by the travelers based on their experience. For the analysis purpose, the study has employed percentage analysis, cross tabulation, independent sample t-test, and analysis of variance, ANOVA in SPSS software.

1. Mode of Travel

Table 30 Analysis of Travel Mode of user.

Mode of Travel

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Solo	260	13.0	13.0	13.0
	Friends	924	46.1	46.1	59.0
	Family	822	41.0	41.0	100.0
	Total	2006	100.0	100.0	

Table 30 show the user's preferences on mode of travel. It shows 46.1% of people prefer to hangout with friends, then comes with family about 41.0%.

2. Mode of Travel * Medium of Travel Crosstabulation

While considering information given in Table 31 cross tabulation between TM and Medium of travel, travelling with family on road comes in higher position, joining with friends for the road trip and Train journey is also in the preferred combination of travel by many users. Solo travellers prefer bike trip as their favourite medium and a few the travellers shown interested in walking.

Table 31 Cross tabulation of TM and TT

Mode of Travel * Medium of Travel Crosstabulation

Count

		Medium of Travel								Total
		Bike	Road	Sea	Train	Flight	Trecking	Cycle	Walk	
Mode of Travel	Solo	84	51	5	61	44	5	7	3	260
	Friends	153	383	17	275	54	30	7	5	924
	Family	52	490	15	130	129	6	0	0	822
Total		289	924	37	466	227	41	14	8	2006

3. Gender * Mode of Travel Crosstabulation

Table 32 represents another cross tabulation between TM and gender, where male prefer to select friends as their travel companions where female seems comfortable to explore the locations with their family. Analysing at solo travellers, male shown higher tendency than females.

Table 32 Cross tabulation between TM and Gender

Gender * Mode of Travel Crosstabulation

Count

		Mode of Travel			Total
		Solo	Friends	Family	
Gender	Male	198	670	391	1259
	Female	62	254	431	747
Total		260	924	822	2006

11 Conclusion

11.1 Summary of Existing Work

Travel recommendation model developed in Malayalam language using machine learning and clustering techniques lacks prior works. The language complexities, unavailability of data and hurdles in practical implications are the big challenges in this domain. Few similar works done in English and other languages which having benchmark dataset. Very few works reported in low resourced Indian Languages such as Malayalam. All available works noted in area, discussed in literature review chapter.

11.2 Summary of Work Done and Findings

In conclusion, this research endeavour marks a significant milestone in the domain of personalized travel recommender systems for the Malayalam language. The extraction of travelogues from social media websites and travel groups, shedding light on the unique and diverse experiences of travelers in the Malayalam-speaking world. This wealth of unstructured text was meticulously processed using state-of-the-art Natural Language Processing (NLP) tools and advanced packages, overcoming the challenges posed by the highly inflectional and morphologically rich Malayalam language. The creation of a customized Part of Travelogue (POT) Tagger further added depth to this research by annotating Malayalam travel reviews, enabling the extraction of meaningful insights from the text. A lookup dictionary was thoughtfully constructed to consolidate and converge tokens, laying the foundation for a structured dataset exclusive to the Malayalam travel domain. The development of Travel DNAs and Location DNA is a testament to the dedication of this research in understanding and categorizing travel experiences in a way that is intuitive and user centric.

The findings of this research work represent a comprehensive exploration of diverse recommender models aimed at transforming the way tourist destinations are recommended to users based on their unique tastes and personal preferences. The utilization of a rule-based Cosine Similarity model provided a strong foundation for generating recommendations by measuring the similarity of users' preferences and destination characteristics. Collaborative Filtering using K-Means Clustering offered a novel approach, enhancing the quality of recommendations by grouping users with similar interests. The CB Filtering using HAC brought a new dimension to the recommendation process, enabling the clustering of destinations based on shared features, thereby delivering more refined suggestions. The incorporation of advanced machine learning, such as the Bidirectional Long Short-Term Memory model and RS using Autoencoders, demonstrated the adaptability and predictive power of these cutting-edge techniques. These findings offer invaluable insights into the performance of each approach, which can inform the future development and deployment of personalized travel recommender systems. The fusion of Autoencoders with traditional machine learning algorithms represents a promising approach to RS.

11.3 Research Contributions

The following are the major contributions of the thesis.

1) Structuring Malayalam travelogues to benchmark dataset.

2) Preparation of customised Part of Travelogue Tagger (POT Tagger)

3) Preparation of Travel DNA and Location DNA

4) Rule based cosine similarity recommender system

5) RS based on clustering techniques

6) Bi-LSTM recommender system

7) RS based on Deep Autoencoders

11.4 Practical Implications

Entire phases of this work done in advanced computing machine. To process Malayalam text, many software packages are available in Linux operating systems. Few customised algorithms took hours to process unstructured travelogue to a structured dataset. Identifying and configuring apt algorithm for this task and constructing neural network to execute dataset also done in varying timeframe. The models are theoretically developed to produce results which shall be converted to display in a handheld device with appropriate tools and packages.

11.5 Limitations and Future of the proposed work

The unavailability of benchmark dataset in Malayalam for travel domain is the biggest challenge. Availability of quality travelogue from social media is the limitation to prepare benchmark dataset. The NLP tools to process Malayalam text is still in infant stage is another limitation as complications of the Language. Unavailability of prior works to refer is another challenge. Quality of recommendation depends on the quality and quantity of dataset, which has to be improved in this work. The future study of the current model focuses on three main areas of improvement. First, by optimizing the Part of Travelogue Tagger, the tagging accuracy can be enhanced, leading to more relevant recommendations. Second, enlarging the corpus capacity of Travel DNA and Location DNA, along with implementing advanced techniques for stemming and preprocessing, will enrich the quality and representativeness of the dataset, providing more nuanced and reflective travel recommendations. Detailed explanations about future scope given in recommendations chapter.

12 Recommendations

The future scope of work in the realm of personalized travel recommender systems is extensive with opportunities for further enhancement and innovation. First and foremost, expanding the dataset to encompass a more diverse range of geographical locations and a larger corpus of travelogues is crucial. This will not only broaden the system's knowledge base but also enable it to cater to an even wider spectrum of traveller preferences. The inclusion of less-explored destinations and a greater diversity of travel experiences will offer travelers more choices and enrich their exploration.

To ensure greater efficiency and accuracy, advanced methodologies should be incorporated into the preprocessing stage of dataset curation. Leveraging cutting-edge Natural Language Processing (NLP) techniques and machine learning algorithms can lead to improved data cleaning, feature extraction, and sentiment analysis. This will ultimately enhance the quality of the structured dataset and refine the recommendations made to users.

Real-time data extraction is another avenue to explore, allowing for dynamic updates to the travelogue database. Incorporating up-to-the-minute information on destinations, user reviews, and travel experiences will ensure that the system remains current and relevant. The travel landscape is constantly evolving, and the ability to provide real-time recommendations will be a valuable addition to the system's capabilities.

Expanding the model to cater to other Indian low-resourced languages is a promising direction. India is a land of diverse languages, and tourists from various regions would benefit from recommendations in their native languages. This will require the development of language-specific models and NLP tools, opening new avenues for research and technology development.

Furthermore, the integration of multi-modal data sources such as images and videos can greatly enhance the system's performance and user experience. Travelers often rely on visual content to make decisions about their destinations. By combining text-based travelogues with multimedia elements, the system can offer a more immersive and informative experience, enabling users to gain a deeper understanding of the places they plan to visit.

Lastly, advanced techniques in deep learning and NLP, including transformers and attention mechanisms, offer exciting opportunities for improving recommendation accuracy. These methods have shown tremendous potential in various fields, and their application to personalized travel recommendations can lead to more precise and context-aware suggestions. By harnessing the power of these state-of-the-art technologies, the system can continuously evolve and provide travelers with recommendations that align seamlessly with their evolving preferences and expectations.

References

[1] M. Nilashi, O. Ibrahim, E. Yadegaridehkordi, S. Samad, E. Akbari, and A. Alizadeh, "Travelers decision making using online review in social network sites: A case on TripAdvisor," *J Comput Sci*, vol. 28, pp. 168–179, Sep. 2018, doi: 10.1016/j.jocs.2018.09.006.

[2] Q. Zhang, J. Lu, and Y. Jin, "Artificial intelligence in recommender systems," *Complex and Intelligent Systems*, vol. 7, no. 1, pp. 439–457, Feb. 2021, doi: 10.1007/s40747-020-00212-w.

[3] M. V. K. and K. P. M. Basheer, "The evolution of travel recommender systems: A comprehensive review," *Malaya Journal of Matematik*, vol. 8, no. 4, pp. 1777–1785, 2020, doi: 10.26637/mjm0804/0075.

[4] J. He and W. W. Chu, "A Social Network-Based Recommender System (SNRS)," 2010, pp. 47–74. doi: 10.1007/978-1-4419-6287-4_4.

[5] F. Alyari and N. Jafari Navimipour, "Recommender systems: A systematic review of the state of the art literature and suggestions for future research," *Kybernetes*, vol. 47, no. 5. Emerald Group Holdings Ltd., pp. 985–1017, May 02, 2018. doi: 10.1108/K-06-2017-0196.

[6] J. He, H. Liu, and H. Xiong, "SocoTraveler: Travel-package recommendations leveraging social influence of different relationship types," *Information and Management*, vol. 53, no. 8, pp. 934–950, Dec. 2016, doi: 10.1016/j.im.2016.04.003.

[7] M. P. Sebastian and G. Santhosh Kumar, "Malayalam Natural Language Processing: Challenges in Building a Phrase-Based Statistical Machine Translation System," *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 22, no. 4, Apr. 2023, doi: 10.1145/3579163.

[8] S. Renjit and S. Idicula, "Natural language inference for Malayalam language using language agnostic sentence representation," *PeerJ Comput Sci*, vol. 7, pp. 1–25, 2021, doi: 10.7717/PEERJ-CS.508.

[9] M. K. Basheer and A. Professor, "An Intelligent Travel Recommender System By Mining Behavioral Attributes From Online Travelogues In Malayalam-A Low Resourced Language Section A-Research paper Eur," *Chem. Bull*, vol. 2023, pp. 4435–4444, doi: 10.48047/ecb/2023.12.si5a.0349.

[10] H. Ko, S. Lee, Y. Park, and A. Choi, “A Survey of Recommendation Systems: Recommendation Models, Techniques, and Application Fields,” *Electronics (Switzerland)*, vol. 11, no. 1. MDPI, Jan. 01, 2022. doi: 10.3390/electronics11010141.

[11] J. Coelho, P. Nitu, and P. Madiraju, “A Personalized Travel Recommendation System Using Social Media Analysis,” in *Proceedings - 2018 IEEE International Congress on Big Data, BigData Congress 2018 - Part of the 2018 IEEE World Congress on Services*, Institute of Electrical and Electronics Engineers Inc., Sep. 2018, pp. 260–263. doi: 10.1109/BigDataCongress.2018.00046.

[12] H. Pan and Z. Zhang, “Research on Context-Awareness Mobile Tourism E-Commerce Personalized Recommendation Model,” *J Signal Process Syst*, vol. 93, no. 2–3, pp. 147–154, Mar. 2021, doi: 10.1007/s11265-019-01504-2.

[13] F. Lorenzi, S. Loh, and M. Abel, “PersonalTour: A recommender system for travel packages,” in *Proceedings - 2011 IEEE/WIC/ACM International Conference on Intelligent Agent Technology, IAT 2011*, 2011, pp. 333–336. doi: 10.1109/WI-IAT.2011.69.

[14] Muneer VK, “Collaborative Travel Recommender System Based on Malayalam Travel Reviews.” 2022, Artificial Intelligence and Speech Technology. AIST 2021. Communications in Computer and Information Science, vol 1546. Springer, Cham. https://doi.org/10.1007/978-3-030-95711-7_53

[15] K. Manohar, A. R. Jayan, and R. Rajan, “Quantitative Analysis of the Morphological Complexity of Malayalam Language.” [Online]. Available: https://en.wikipedia.org/wiki/Malayalam

[16] D. Hee Park, I. Young Choi, H. Kyeong Kim, and J. Kyeong Kim, “A Review and Classification of Recommender Systems Research.”

[17] A. Anandhan, L. Shuib, M. A. Ismail, and G. Mujtaba, “Social Media Recommender Systems: Review and Open Research Issues,” *IEEE Access*, vol. 6, pp. 15608–15628, Feb. 2018, doi: 10.1109/ACCESS.2018.2810062.

[18] J. Han and H. Lee, “Adaptive landmark recommendations for travel planning: Personalizing and clustering landmarks using geo-tagged social media,” *Pervasive Mob Comput*, vol. 18, pp. 4–17, Apr. 2015, doi: 10.1016/j.pmcj.2014.08.002.

[19] B. Batrinca and P. C. Treleaven, "Social media analytics: a survey of techniques, tools and platforms," *AI Soc*, vol. 30, no. 1, pp. 89–116, Feb. 2015, doi: 10.1007/s00146-014-0549-4.

[20] S. Stieglitz, M. Mirbabaie, B. Ross, and C. Neuberger, "Social media analytics – Challenges in topic discovery, data collection, and data preparation," *Int J Inf Manage*, vol. 39, pp. 156–168, Apr. 2018, doi: 10.1016/j.ijinfomgt.2017.12.002.

[21] Muneer VK, "A Collaborative Destination Recommender Model in Dravidian Language by Social Media Analysis." [Online]. Available: www.facebook.com/groups/teamsanchari.

[22] P. Forbrig, "Personalized Sightseeing Tours Support Using Mobile Devices," 2010. Doi : https://doi.org/10.1007/978-3-642-15231-3_35.

[23] D. Schneider and K. Harknett, "What's to Like? Facebook as a Tool for Survey Data Collection," *Sociol Methods Res*, vol. 51, no. 1, pp. 108–140, Feb. 2022, doi: 10.1177/0049124119882477.

[24] S. A. Salloum, M. Al-Emran, A. A. Monem, and K. Shaalan, "A survey of text mining in social media: Facebook and Twitter perspectives," *Advances in Science, Technology and Engineering Systems*, vol. 2, no. 1, pp. 127–133, 2017, doi: 10.25046/aj020115.

[25] L. C. Dewi, Meiliana, and A. Chandra, "Social media web scraping using social media developers API and regex," in *Procedia Computer Science*, Elsevier B.V., 2019, pp. 444–449. doi: 10.1016/j.procs.2019.08.237.

[26] M. Zimmer, "'But the data is already public': On the ethics of research in Facebook," *Ethics Inf Technol*, vol. 12, no. 4, pp. 313–325, Dec. 2010, doi: 10.1007/s10676-010-9227-5.

[27] J. He and W. W. Chu, "Design Considerations for a Social Network-Based Recommendation System (SNRS)," in *Community-Built Databases*, Springer Berlin Heidelberg, 2011, pp. 73–106. doi: 10.1007/978-3-642-19047-6_4.

[28] G. J. Chen *et al.*, "Realtime data processing at Facebook," in *Proceedings of the ACM SIGMOD International Conference on Management of Data*, Association for Computing Machinery, Jun. 2016, pp. 1087–1098. doi: 10.1145/2882903.2904441.

[29] G. R. Devi, P. V. Veena, M. A. Kumar, and K. P. Soman, "Entity Extraction for Malayalam Social Media Text Using Structured Skip-gram Based Embedding Features from Unlabeled Data," in *Procedia Computer Science*, Elsevier B.V., 2016, pp. 547–553. doi: 10.1016/j.procs.2016.07.276.

[30] M. R. Kalideeen, I. Safeek, and M. R. Kalideen, "PREPROCESSING ON FACEBOOK DATA FOR SENTIMENT ANALYSIS," 2017. [Online]. Available: https://www.researchgate.net/publication/328315374

[31] M. K. Basheer and A. Muhaimin, "Online Malayalam Script Assortment And Pre-Processing For Building Recommender Systems." [Online]. Available: https://developers.facebook.com/

[32] A. P. Ajees, K. J. Abrar, M. I. Sumam, and M. Sreenathan, "A Deep Level Tagger for Malayalam, a Morphologically Rich Language," *Journal of Intelligent Systems*, vol. 30, no. 1, pp. 115–129, Jan. 2020, doi: 10.1515/jisys-2019-0070.

[33] by G. Gayathri and V. M. Sarma, "Malayalam Morphosyntax: Inflectional Features and their Acquisition," 2019.

[34] B. Premjith, K. P. Soman, and M. A. Kumar, "A deep learning approach for Malayalam morphological analysis at character level," in *Procedia Computer Science*, Elsevier B.V., 2018, pp. 47–54. doi: 10.1016/j.procs.2018.05.058.

[35] M. Subhash, "A Rule Based Approach for Root Word Identification in Malayalam Language," *International Journal of Computer Science and Information Technology*, vol. 4, no. 3, pp. 159–166, Jun. 2012, doi: 10.5121/ijcsit.2012.4313.

[36] Y. Y. Chen, A. J. Cheng, and W. H. Hsu, "Travel recommendation by mining people attributes and travel group types from community-contributed photos," *IEEE Trans Multimedia*, vol. 15, no. 6, pp. 1283–1295, 2013, doi: 10.1109/TMM.2013.2265077.

[37] F. Alyari and N. Jafari Navimipour, "Recommender systems: A systematic review of the state of the art literature and suggestions for future research," *Kybernetes*, vol. 47, no. 5. Emerald Group Holdings Ltd., pp. 985–1017, May 02, 2018. doi: 10.1108/K-06-2017-0196.

[38] K. Chaudhari and A. Thakkar, "A Comprehensive Survey on Travel Recommender Systems," *Archives of Computational Methods in Engineering*, vol. 27, no. 5, pp. 1545–1571, Nov. 2020, doi: 10.1007/s11831-019-09363-7.

[39] Y. L. Hsueh and H. M. Huang, "Personalized itinerary recommendation with time constraints using GPS datasets," *Knowl Inf Syst*, vol. 60, no. 1, pp. 523–544, Jul. 2019, doi: 10.1007/s10115-018-1217-7.

[40] J. C. Hung, V. Hsu, and M. M. Weng, "A Study for Task based Recommendation System for Travel Navigation." 2013, doi : 10.1109/ICAwST.2013.6765488.

[41] Y. Zheng, L. Zhang, Z. Ma, X. Xie, and W. Y. Ma, "Recommending friends and locations based on individual location history," *ACM Transactions on the Web*, vol. 5, no. 1, Feb. 2011, doi: 10.1145/1921591.1921596.

[42] H. Yu Yao Dan Luo Jing Zhang Mu, "Research on Personalized Recommender System for Tourism Information Service," 2013. [Online]. Available: www.iiste.org

[43] W. Xu-yin, H. Xiang-pei, and L. Wei-guo, "An Urban Tourism Intelligent Recommendation System Based on WebGIS I1 2 i2 1,2."

[44] P. Brusilovsky, "LNCS 4321 - Hybrid Web Recommender Systems," 2007. [Online]. Available: http://www.google.com

[45] A. Bin Hossain, W. U. Hasan, K. T. Zaman, and K. Howlader, "Integrated Music Recommendation System Using Collaborative and Content Based Filtering, and Sentiment Analysis," 2023, pp. 162–172. doi: 10.1007/978-3-031-34622-4_13.

[46] Z. Yu, H. Xu, Z. Yang, and B. Guo, "Personalized Travel Package with Multi-Point-of-Interest Recommendation Based on Crowdsourced User Footprints," *IEEE Trans Hum Mach Syst*, vol. 46, no. 1, pp. 151–158, Feb. 2016, doi: 10.1109/THMS.2015.2446953.

[47] N. Hazrati and F. Ricci, "Choice models and recommender systems effects on users' choices," *User Model User-adapt Interact*, 2023, doi: 10.1007/s11257-023-09366-x.

[48] R. SujithraKanmani and B. Surendiran, "Boosting credibility of a Recommender System using Deep Learning Techniques - An Empirical

Study," *International Journal of Engineering Trends and Technology*, vol. 69, no. 10, pp. 235–243, Oct. 2021, doi: 10.14445/22315381/IJETT-V69I10P230.

[49] A. Fareed, S. Hassan, S. B. Belhaouari, and Z. Halim, "A collaborative filtering recommendation framework utilizing social networks," *Machine Learning with Applications*, vol. 14, p. 100495, Dec. 2023, doi: 10.1016/j.mlwa.2023.100495.

[50] P. Srikanth, E. Ushitaasree, S. M. Bhargav Bhattaram, and G. Paavaianand, "Movie Recommendation System Using Deep Autoencoder," in *Proceedings of the 5th International Conference on Electronics, Communication and Aerospace Technology, ICECA 2021*, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 1059–1064. doi: 10.1109/ICECA52323.2021.9675960.

[51] K. Singh, H. K. Shakya, and B. Biswas, "Clustering of people in social network based on textual similarity," *Perspect Sci (Neth)*, vol. 8, pp. 570–573, Sep. 2016, doi: 10.1016/j.pisc.2016.06.023.

[52] S. Jiang, X. Qian, J. Shen, Y. Fu, and T. Mei, "Author topic model-based collaborative filtering for personalized POI recommendations," *IEEE Trans Multimedia*, vol. 17, no. 6, pp. 907–918, Jun. 2015, doi: 10.1109/TMM.2015.2417506.

[53] S. Jiang, X. Qian, T. Mei, and Y. Fu, "Personalized Travel Sequence Recommendation on Multi-Source Big Social Media," *IEEE Trans Big Data*, vol. 2, no. 1, pp. 43–56, Mar. 2016, doi: 10.1109/tbdata.2016.2541160.

[54] P. B. Thorat, R. M. Goudar, and S. Barve, "Survey on Collaborative Filtering, Content-based Filtering and Hybrid Recommendation System," 2015.

[55] I. Karabila, N. Darraz, A. El-Ansari, N. Alami, and M. El Mallahi, "Enhancing Collaborative Filtering-Based Recommender System Using Sentiment Analysis," *Future Internet*, vol. 15, no. 7, Jul. 2023, doi: 10.3390/fi15070235.

[56] A. Anandhan, L. Shuib, M. A. Ismail, and G. Mujtaba, "Social Media Recommender Systems: Review and Open Research Issues," *IEEE Access*, vol. 6, pp. 15608–15628, Feb. 2018, doi: 10.1109/ACCESS.2018.2810062.

[57] D. Roy and M. Dutta, "A systematic review and research perspective on recommender systems," *J Big Data*, vol. 9, no. 1, Dec. 2022, doi: 10.1186/s40537-022-00592-5.

[58] B. Bhatt, P. J. Patel, H. Gaudani, and A. Professor, "A Review Paper on Machine Learning Based Recommendation System," 2014. [Online]. Available: www.ijedr.org

[59] Y. Zheng, L. Zhang, Z. Ma, X. Xie, and W. Y. Ma, "Recommending friends and locations based on individual location history," *ACM Transactions on the Web*, vol. 5, no. 1, Feb. 2011, doi: 10.1145/1921591.1921596.

[60] P. Serdyukov and M. J. T. Reinders, "Personalised Travel Recommendation based on Location Co-occurrence Genetic Network Modeling View project Intelligent Molecular Diagnostic System View project Maarten Clements," 2011. [Online]. Available: http://www.facebook.com/

[61] S. P. R. Asaithambi, R. Venkatraman, and S. Venkatraman, "A Thematic Travel Recommendation System Using an Augmented Big Data Analytical Model," *Technologies (Basel)*, vol. 11, no. 1, Feb. 2023, doi: 10.3390/technologies11010028.

[62] J. Aravind and R. Parvathi, "A personalized location recommendation system based on probability and proximity," *International Journal of Recent Technology and Engineering*, vol. 8, no. 2, pp. 714–718, Jul. 2019, doi: 10.35940/ijrte.B1723.078219.

[63] Y. Zheng, L. Zhang, Z. Ma, X. Xie, and W. Y. Ma, "Recommending friends and locations based on individual location history," *ACM Transactions on the Web*, vol. 5, no. 1, Feb. 2011, doi: 10.1145/1921591.1921596.

[64] G. Adomavicius and A. Tuzhilin, "Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions."

[65] R. Kitamura and T. Itoh, "Tourist spot recommendation applying generic object recognition with travel photos," in *Information Visualisation - Biomedical Visualization, Visualisation on Built and Rural Environments and Geometric Modelling and Imaging, IV 2018*, Institute of Electrical and Electronics Engineers Inc., Dec. 2018, pp. 1–5. doi: 10.1109/iV.2018.00011.

[66] X. Luo, Y. Xia, and Q. Zhu, "Incremental Collaborative Filtering recommender based on Regularized Matrix Factorization," *Knowl Based Syst*, vol. 27, pp. 271–280, Mar. 2012, doi: 10.1016/j.knosys.2011.09.006.

[67] X. Luo, Y. Xia, and Q. Zhu, "Incremental Collaborative Filtering recommender based on Regularized Matrix Factorization," *Knowl Based Syst*, vol. 27, pp. 271–280, Mar. 2012, doi: 10.1016/j.knosys.2011.09.006.

[68] B. Bhatt, P. J. Patel, H. Gaudani, and A. Professor, "A Review Paper on Machine Learning Based Recommendation System," 2014. [Online]. Available: www.ijedr.org

[69] H. Wang, N. Wang, and D. Y. Yeung, "Collaborative deep learning for recommender systems," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Association for Computing Machinery, Aug. 2015, pp. 1235–1244. doi: 10.1145/2783258.2783273.

[70] P. S. Singh, "A Review on Travel Recommendation Techniques," *Int J Sci Eng Res*, vol. 9, no. 10, 2018, [Online]. Available: http://www.ijser.org

[71] S. Glaret Shirley, K. Subrahmanyam, D. Susrija, and P. Akhila, "K-Means Algorithm and Clustering Technique for A Recommender System," *International Journal of Application on Sciences, Technology and Engineering*, vol. 1, no. 1, pp. 302–312, Feb. 2023, doi: 10.24912/ijaste.v1.i1.302-312.

[72] Y. L. Hsueh and H. M. Huang, "Personalized itinerary recommendation with time constraints using GPS datasets," *Knowl Inf Syst*, vol. 60, no. 1, pp. 523–544, Jul. 2019, doi: 10.1007/s10115-018-1217-7.

[73] J. Coelho, P. Nitu, and P. Madiraju, "A Personalized Travel Recommendation System Using Social Media Analysis," in *Proceedings - 2018 IEEE International Congress on Big Data, BigData Congress 2018 - Part of the 2018 IEEE World Congress on Services*, Institute of Electrical and Electronics Engineers Inc., Sep. 2018, pp. 260–263. doi: 10.1109/BigDataCongress.2018.00046.

[74] W. Wei, H. Wu, and H. Ma, "An autoencoder and LSTM-based traffic flow prediction method," *Sensors (Switzerland)*, vol. 19, no. 13, Jul. 2019, doi: 10.3390/s19132946.

[75] C. Te Li, H. Y. Chen, R. H. Chen, and H. P. Hsieh, "On route planning by inferring visiting time, modeling user preferences, and mining representative trip patterns," *Knowl Inf Syst*, vol. 56, no. 3, pp. 581–611, Sep. 2018, doi: 10.1007/s10115-017-1106-5.

[76] H. Sen Chiang and T. C. Huang, "User-adapted travel planning system for personalized schedule recommendation," *Information Fusion*, vol. 21, no. 1, pp. 3–17, 2015, doi: 10.1016/j.inffus.2013.05.011.

[77] W. C. and Mobile Computing, "Retracted: Route Planning of Health Care Tourism Based on Computer Deep Learning," *Wirel Commun Mob Comput*, vol. 2023, pp. 1–1, Aug. 2023, doi: 10.1155/2023/9857094.

[78] D. Z. Liu and G. Singh, "A Recurrent Neural Network Based Recommendation System." [Online]. Available: https://www.yelp.com/dataset_challenge

[79] L. Gao and J. Li, "E-Commerce Personalized Recommendation Model Based on Semantic Sentiment," *Mobile Information Systems*, vol. 2022, 2022, doi: 10.1155/2022/7246802.

[80] H. Barzan Abdalla, A. Ahmed, B. Mehmed, M. Gheisari, M. Cheraghy, and H. B. Abdalla, "An Efficient Recommendation System in E-commerce using Passer learning optimization based on Bi-LSTM." [Online]. Available: http://arxiv.org/abs/2308.00137

[81] Q. Li, X. Zheng, and X. Wu, "Neural Collaborative Autoencoder," Dec. 2017, [Online]. Available: http://arxiv.org/abs/1712.09043

[82] F. Lorenzi, S. Loh, and M. Abel, "PersonalTour: A recommender system for travel packages," in *Proceedings - 2011 IEEE/WIC/ACM International Conference on Intelligent Agent Technology, IAT 2011*, 2011, pp. 333–336. doi: 10.1109/WI-IAT.2011.69.

[83] D. Z. Liu and G. Singh, "A Recurrent Neural Network Based Recommendation System." [Online]. Available: https://www.yelp.com/dataset_challenge

[84] G. Liu and J. Guo, "Bidirectional LSTM with attention mechanism and convolutional layer for text classification," *Neurocomputing*, vol. 337, pp. 325–338, Apr. 2019, doi: 10.1016/j.neucom.2019.01.078.

[85] Y. Zhang, "Music Recommendation System and Recommendation Model Based on Convolutional Neural Network," *Mobile Information Systems*, vol. 2022, 2022, doi: 10.1155/2022/3387598.

[86] P. S. Singh, "A Review on Travel Recommendation Techniques," *Int J Sci Eng Res*, vol. 9, no. 10, 2018, [Online]. Available: http://www.ijser.org

[87] J. of Sensors, "Retracted: Implementation of Personalized Information Recommendation Platform System Based on Deep Learning Tourism," *J Sens*, vol. 2023, pp. 1–1, Aug. 2023, doi: 10.1155/2023/9857129.

[88] D. Ferreira, S. Silva, A. Abelha, and J. Machado, "Recommendation system using autoencoders," *Applied Sciences (Switzerland)*, vol. 10, no. 16, Aug. 2020, doi: 10.3390/app10165510.

[89] Q. Li, X. Zheng, and X. Wu, "Neural Collaborative Autoencoder," Dec. 2017, [Online]. Available: http://arxiv.org/abs/1712.09043

[90] Y. Liu and M. Lapata, "Text Summarization with Pretrained Encoders," Aug. 2019, [Online]. Available: http://arxiv.org/abs/1908.08345

[91] O. Kuchaiev and B. Ginsburg, "Training Deep AutoEncoders for Collaborative Filtering," Aug. 2017, [Online]. Available: http://arxiv.org/abs/1708.01715

[92] S. Zhang, L. Yao, A. Sun, and Y. Tay, "Deep learning based recommender system: A survey and new perspectives," *ACM Computing Surveys*, vol. 52, no. 1. Association for Computing Machinery, Feb. 01, 2019. doi: 10.1145/3285029.

[93] O. Prakash Verma and N. Beniwal, "A Novel Recommender System Using PSO, FCM and, Autoencoder," 2022, doi: 10.21203/rs.3.rs-2125860/v1.

[94] X. Zhang, X. Wang, H. Li, S. Sun, and F. Liu, "Monthly runoff prediction based on a coupled VMD-SSA-BiLSTM model," *Sci Rep*, vol. 13, no. 1, p. 13149, Dec. 2023, doi: 10.1038/s41598-023-39606-4.

[95] F. Strub, R. Gaudel, and J. Mary, "Hybrid recommender system based on autoencoders," in *ACM International Conference Proceeding Series*, Association for Computing Machinery, Sep. 2016, pp. 11–16. doi: 10.1145/2988450.2988456.

[96] Q. Guo, J. Jia, G. Shen, L. Zhang, L. Cai, and Z. Yi, "Learning robust uniform features for cross-media social data by using cross autoencoders," *Knowl Based Syst*, vol. 102, pp. 64–75, Jun. 2016, doi: 10.1016/j.knosys.2016.03.028.

[97] C. Chen, X. Meng, Z. Xu, and T. Lukasiewicz, "Location-aware personalized news recommendation with deep semantic analysis," *IEEE Access*, vol. 5, pp. 1624–1638, 2017, doi: 10.1109/ACCESS.2017.2655150.

[98] H. Kori, S. Hattori, T. Tezuka, and K. Tanaka, "LNCS 4351 - Automatic Generation of Multimedia Tour Guide from Local Blogs."

[99] Z. Zhang and Y. Morimoto, "Collaborative Hotel Recommendation based on Topic and Sentiment of Review Comments," 2017.

[100] M. R. Lyu, I. King, C. Cheng, and H. Yang, "Where you like to go next: Successive point-of-interest recommendation." [Online]. Available: http://statspotting.com/2012/04/foursquare-statistics-20-

[101] V. P. Abeera *et al.*, "LNCS 6411 - Morphological Analyzer for Malayalam Using Machine Learning," 2012.

[102] K. Manohar, A. R. Jayan, and R. Rajan, "Quantitative analysis of the morphological complexity of malayalam language," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Springer Science and Business Media Deutschland GmbH, 2020, pp. 71–78. doi: 10.1007/978-3-030-58323-1_7.

[103] B. Premjith, K. P. Soman, and M. A. Kumar, "A deep learning approach for Malayalam morphological analysis at character level," in *Procedia Computer Science*, Elsevier B.V., 2018, pp. 47–54. doi: 10.1016/j.procs.2018.05.058.

[104] K. Manohar, A. R. Jayan, and R. Rajan, "Quantitative analysis of the morphological complexity of malayalam language," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Springer Science and Business Media Deutschland GmbH, 2020, pp. 71–78. doi: 10.1007/978-3-030-58323-1_7.

[105] D. M. N. Mubarak, "A New Approach to Parts of Speech Tagging in Malayalam," *International Journal of Computer Science and Information Technology*, vol. 7, no. 5, pp. 121–130, Oct. 2015, doi: 10.5121/ijcsit.2015.7509.

[106] J. P. Jayan and R. R. R, "Parts Of Speech Tagger and Chunker for Malayalam-Statistical Approach", [Online]. Available: www.iiste.org

[107] M. Gayathidevi, Cs. Reddy, and A. Professor, "A Framework for Tourist Recommendation System Exploiting Geo-Tagged Photos."

[108] Q. Zhang, Y. Liu, L. Liu, S. Lu, Y. Feng, and X. Yu, "Location identification and personalized recommendation of tourist attractions based on image processing," *Traitement du Signal*, vol. 38, no. 1, pp. 197–205, Feb. 2021, doi: 10.18280/TS.380121.

[109] D. Gunawan, C. A. Sembiring, and M. A. Budiman, "The Implementation of Cosine Similarity to Calculate Text Relevance between Two Documents," in *Journal of Physics: Conference Series*, Institute of Physics Publishing, Mar. 2018. doi: 10.1088/1742-6596/978/1/012120.

[110] A. Rizqi Lahitani, A. Erna Permanasari, and N. Akhmad Setiawan, "Cosine Similarity to Determine Similarity Measure: Study Case in Online Essay Assessment."

[111] B. Li and L. Han, "LNCS 8206 - Distance Weighted Cosine Similarity Measure for Text Classification," 2013.

[112] R. Chen, Q. Hua, Y. S. Chang, B. Wang, L. Zhang, and X. Kong, "A survey of collaborative filtering-based recommender systems: from traditional methods to hybrid methods based on social networks," *IEEE Access*, vol. 6, pp. 64301–64320, 2018, doi: 10.1109/ACCESS.2018.2877208.

[113] Z. Zhang, H. Pan, G. Xu, Y. Wang, and P. Zhang, "A context-awareness personalized tourist attraction recommendation algorithm," *Cybernetics and Information Technologies*, vol. 16, no. Specialissue6, pp. 146–159, 2016, doi: 10.1515/cait-2016-0084.

[114] A. Yassine, L. Mohamed, and M. Al Achhab, "Intelligent recommender system based on unsupervised machine learning and demographic attributes," *Simul Model Pract Theory*, vol. 107, Feb. 2021, doi: 10.1016/j.simpat.2020.102198.

[115] S. Sathasivam and A. M. Sagir, "Comparative Study of Euclidean and City Block Distances in Fuzzy C-Means Clustering Algorithm," *International Journal of Computational and Electronic Aspects in Engineering*, vol. 1, no. 1, Dec. 2014, doi: 10.26706/ijceae.1.1.20141203.

[116] N. Vara, M. Mirzabeigi, H. Sotudeh, and S. M. Fakhrahmad, "Application of k-means clustering algorithm to improve effectiveness of the results recommended by journal recommender system," *Scientometrics*, vol. 127, no. 6, pp. 3237–3252, Jun. 2022, doi: 10.1007/s11192-022-04397-4.

[117] Y. Rani[1] and H. Rohil, "A Study of Hierarchical Clustering Algorithm," 2013. [Online]. Available: http://www.irphouse.com/ijict.htm

[118] E. Seo and H. J. Choi, "Movie recommendation with K-means clustering and Self-Organizing Map methods," in *ICAART 2010 - 2nd International Conference on Agents and Artificial Intelligence, Proceedings*, 2010, pp. 385–390. doi: 10.5220/0002737603850390.

[119] B. Bai, Y. Fan, W. Tan, and J. Zhang, "DLTSR: A Deep Learning Framework for Recommendations of Long-Tail Web Services," *IEEE Trans Serv Comput*, vol. 13, no. 1, pp. 73–85, Jan. 2020, doi: 10.1109/TSC.2017.2681666.

[120] D. Marutho, S. Hendra Handaka, and E. Wijaya, "The Determination of Cluster Number at k-mean using Elbow Method and Purity Evaluation on Headline News."

[121] G. Boo, E. Darin, D. R. Thomson, and A. J. Tatem, "A grid-based sample design framework for household surveys," *Gates Open Res*, vol. 4, p. 13, Jan. 2020, doi: 10.12688/gatesopenres.13107.1.

[122] E. Umargono, J. E. Suseno, and V. Gunawan, "K-Means Clustering Optimization Using the Elbow Method and Early Centroid Determination Based on Mean and Median Formula," 2020.

[123] S. Thara and P. Poornachandran, "Social media text analytics of Malayalam–English code-mixed using deep learning," *J Big Data*, vol. 9, no. 1, Dec. 2022, doi: 10.1186/s40537-022-00594-3.

[124] C. Zhao, J. You, X. Wen, and X. Li, "Deep Bi-LSTM networks for sequential recommendation," *Entropy*, vol. 22, no. 8, Aug. 2020, doi: 10.3390/E22080870.

[125] E. Pantano, C. V. Priporas, and N. Stylos, "'You will like it!' using open data to predict tourists' response to a tourist attraction," *Tour Manag*, vol. 60, pp. 430–438, Jun. 2017, doi: 10.1016/j.tourman.2016.12.020.

[126] H. B. Aji and E. B. Setiawan, "Detecting Hoax Content on Social Media Using Bi-LSTM and RNN," *Building of Informatics, Technology and Science (BITS)*, vol. 5, no. 1, Jun. 2023, doi: 10.47065/bits.v5i1.3585.

[127] A. Noorian, A. Harounabadi, and M. Hazratifard, "A sequential neural recommendation system exploiting BERT and LSTM on social media posts," *Complex and Intelligent Systems*, 2023, doi: 10.1007/s40747-023-01191-4.

[128] W. Yin, K. Kann, M. Yu, and H. Schütze, "Comparative Study of CNN and RNN for Natural Language Processing," Feb. 2017, [Online]. Available: http://arxiv.org/abs/1702.01923

[129] R. L. Abduljabbar, H. Dia, and P. W. Tsai, "Development and evaluation of bidirectional LSTM freeway traffic forecasting models using simulation data," *Sci Rep*, vol. 11, no. 1, Dec. 2021, doi: 10.1038/s41598-021-03282-z.

[130] B. Jang, M. Kim, G. Harerimana, S. U. Kang, and J. W. Kim, "Bi-LSTM model to increase accuracy in text classification: Combining word2vec CNN and attention mechanism," *Applied Sciences (Switzerland)*, vol. 10, no. 17, Sep. 2020, doi: 10.3390/app10175841.

[131] B. H. Pansambal and A. B. Nandgaokar, "Integrating Dropout Regularization Technique at Different Layers to Improve the Performance of Neural Networks." [Online]. Available: www.ijacsa.thesai.org

[132] V. Ganesh and M. Kamarasan, "Parameter Tuned Bi-Directional Long Short Term Memory Based Emotion With Intensity Sentiment Classification Model Using Twitter Data."

[133] Y. Zhang, J. Zheng, Y. Jiang, G. Huang, and R. Chen, "A text sentiment classification modeling method based on coordinated CNN-LSTM-attention model," *Chinese Journal of Electronics*, vol. 28, no. 1, pp. 120–126, Jan. 2019, doi: 10.1049/cje.2018.11.004.

[134] P. Cerda, G. Varoquaux, and B. Kégl, "Similarity encoding for learning with dirty categorical variables," Jun. 2018, [Online]. Available: http://arxiv.org/abs/1806.00979

[135] M. Shirdel, R. Asadi, D. Do, and M. Hintlian, "Deep Learning with Kernel Flow Regularization for Time Series Forecasting," Sep. 2021, [Online]. Available: http://arxiv.org/abs/2109.11649

[136] Y. Tian, Y. Zhang, and H. Zhang, "Recent Advances in Stochastic Gradient Descent in Deep Learning," *Mathematics*, vol. 11, no. 3. MDPI, Feb. 01, 2023. doi: 10.3390/math11030682.

[137] K. Mecheri, R. Mamadji, S. Klai, and L. Souici-Meslati, "Enhanced Deep Autoencoder based Recommender System," in *Proceedings of the 2022 1st International Conference on Big Data, IoT, Web Intelligence and Applications,*

BIWA 2022, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 31–36. doi: 10.1109/BIWA57631.2022.10038127.

[138] D. Charte, F. Charte, S. García, M. J. del Jesus, and F. Herrera, "A practical tutorial on autoencoders for nonlinear feature fusion: Taxonomy, models, software and guidelines," *Information Fusion*, vol. 44, pp. 78–96, Nov. 2018, doi: 10.1016/j.inffus.2017.12.007.

[139] S. Zhang, Y. Yao, J. Hu, Y. Zhao, S. Li, and J. Hu, "Deep autoencoder neural networks for short-term traffic congestion prediction of transportation networks," *Sensors (Switzerland)*, vol. 19, no. 10, May 2019, doi: 10.3390/s19102229.

[140] Q. Wang, H. Jiang, M. Qiu, Y. Liu, and D. Ye, "TGAE: Temporal Graph Autoencoder for Travel Forecasting," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 8, pp. 8529–8541, Aug. 2023, doi: 10.1109/TITS.2022.3202089.

[141] Q. Gao, W. Wang, K. Zhang, X. Yang, C. Miao, and T. Li, "Self-supervised representation learning for trip recommendation," *Knowl Based Syst*, vol. 247, Jul. 2022, doi: 10.1016/j.knosys.2022.108791.

[142] T. Yao *et al.*, "Self-supervised Learning for Large-scale Item Recommendations," in *International Conference on Information and Knowledge Management, Proceedings*, Association for Computing Machinery, Oct. 2021, pp. 4321–4330. doi: 10.1145/3459637.3481952.

[143] J. Yu, H. Yin, X. Xia, T. Chen, J. Li, and Z. Huang, "Self-Supervised Learning for Recommender Systems: A Survey," Mar. 2022, [Online]. Available: http://arxiv.org/abs/2203.15876

[144] W. Wei, C. Huang, L. Xia, and C. Zhang, "Multi-Modal Self-Supervised Learning for Recommendation," in *ACM Web Conference 2023 - Proceedings of the World Wide Web Conference, WWW 2023*, Association for Computing Machinery, Inc, Apr. 2023, pp. 790–800. doi: 10.1145/3543507.3583206.

[145] P. Brusilovsky, "LNCS 4321 - Hybrid Web Recommender Systems," 2007. [Online]. Available: http://www.google.com

[146] K. M. Houk and K. Thornhill, "Using Facebook Page Insights Data to Determine Posting Best Practices in an Academic Health Sciences Library,"

Journal of Web Librarianship, vol. 7, no. 4, pp. 372–388, 2013, doi: 10.1080/19322909.2013.837346.

[147] T. Alashkar, S. Jiang, S. Wang, and Y. Fu, "Examples-Rules Guided Deep Neural Network for Makeup Recommendation." [Online]. Available: www.aaai.org

[148] M. M. Aziz, M. D. Purbalaksono, and A. Adiwijaya, "Method comparison of Naïve Bayes, Logistic Regression, and SVM for Analyzing Movie Reviews," *Building of Informatics, Technology and Science (BITS)*, vol. 4, no. 4, Mar. 2023, doi: 10.47065/bits.v4i4.2644.

[149] Y. Liu, Y. Wang, and J. Zhang, "LNCS 7473 - New Machine Learning Algorithm: Random Forest," 2012.

[150] S. Uddin, I. Haque, H. Lu, M. A. Moni, and E. Gide, "Comparative performance analysis of K-nearest neighbour (KNN) algorithm and its different variants for disease prediction," *Sci Rep*, vol. 12, no. 1, Dec. 2022, doi: 10.1038/s41598-022-10358-x.

[151] C. Y. J. Peng, K. L. Lee, and G. M. Ingersoll, "An introduction to logistic regression analysis and reporting," *Journal of Educational Research*, vol. 96, no. 1, pp. 3–14, 2002, doi: 10.1080/00220670209598786.

[152] P. Srikanth, E. Ushitaasree, S. M. Bhargav Bhattaram and G. PaavaiAnand, "Movie Recommendation System Using Deep Autoencoder," *2021 5th International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, Coimbatore, India, 2021, pp. 1059-1064, doi: 10.1109/ICECA52323.2021.9675960.

[153] Ferreira, Diana & Silva, Sofia & Abelha, António & Machado, José. (2020). Recommendation System Using Autoencoders. Applied Sciences. 10. 5510. 10.3390/app10165510.

[154] Braja Gopal Patra, Dipankar Das, and Sivaji Bandyopadhyay. 2017. Retrieving Similar Lyrics for Music Recommendation System. In *Proceedings of the 14th International Conference on Natural Language Processing (ICON-2017)*, pages 290–297. https://aclanthology.org/W17-7536/

[155] Wohiduzzaman, Kazi and Sabir Ismail. "Recommendation System for Bangla News Article with Anaphora Resolution." *2018 4th International Conference on*

Electrical Engineering and Information & Communication Technology (iCEEiCT) (2018): 467-472. DOI:10.1109/CEEICT.2018.8628075

[156] Sanzida Akter, Aanan Ehsan Siam, Bengali Movie Recommendation System using K Nearest Neighbor and Cosine Similarity. In Proceedings of the 2023 9th International Conference on Computer Technology Applications (ICCTA '23). Association for Computing Machinery, New York, NY, USA, 25–29. https://doi.org/10.1145/3605423.3605432

[157] Prabir Mondal, Pulkit Kapoor, Siddharth Singh, Task-Specific and Graph Convolutional Network based Multi-modal Movie Recommendation System in Indian Setting, Procedia Computer Science, Volume 222, 2023, Pages 591-600, ISSN 1877-0509, https://doi.org/10.1016/j.procs.2023.08.197.

Publications of the Author

[1] **Muneer V. K.,** Mohamed Basheer, K. P. (2020). The evolution of travel recommender systems: A comprehensive review. Malaya Journal of Matematik, 8(4), 1777–1785. (October 2020), https://doi.org/10.26637/mjm0804/0075 ISSN: 2319-3786 (**UGC CARE**)

[2] **Muneer V. K**., Mohamed Basheer K. P., Collaborative Travel Recommender System based on Malayalam Travel reviews, (Jan 2022). In Springer eBooks (pp. 651–659). Communications in Computer and Information Science (CCIS,volume 1546), https://doi.org/10.1007/978-3-030-95711-7_53. Online ISBN: 978-3-030-95711-7 (**Scopus Indexed**)

[3] **Muneer V.K,** Mohamed Basheer K.P. (2023). A Comparative study of Collaborative Filtering and Content-Based Approaches for improving the Accuracy of travel recommender Systems for Malayalam language. May 2023, International Journal of Advanced Networking and Applications, 14(06), 5717–5721. https://doi.org/10.35444/ijana.2023.14608. ISSN: 0975-0282. (**Peer Reviewed**)

[4] **Muneer VK,** Mohamed Basheer KP, Thandil RK. (2023) Convolutional Neural Network-Based Automatic Speech Emotion Recognition System for Malayalam. Indian Journal of Science and Technology.16(46):4410-4420. December 2023, https://doi.org/10.17485/IJST/v16i46.2090. P-ISSN 0974-6846. (**Web of Science**).

[5] **Muneer V. K**., Mohamed Basheer K. P., Rizwana K. T., & Muhaimin, Online Malayalam script assortment and preprocessing for building recommender systems., A. W. (April 2022). In Smart Innovation, Systems and Technologies (pp. 57–66). https://doi.org/10.1007/978-981-16-9669-5_5. E ISSN: 2190-3026 (**Scopus Indexed**)

[6] **Muneer V.K.**, Mohamed Basheer KP, Thandil RK. Utilizing BiLSTM For Fine-Grained Aspect-Based Travel Recommendations Using Travel Reviews In Low Resourced Language, Journal of Electrical Systems, Vol. 20 No. 2s, https://doi.org/10.52783/jes.1133, ISSN:1112-5209 (**Scopus Indexed**)

[7] **Muneer V.K**, Mohamed Basheer K.P, A Hybrid Travel Recommender Model Based on Deep Level Autoencoder And Machine Learning Algorithms, (Dec 2023), https://doi.org/10.53555/jaz.v44i5.3571, Vol. 44 No. 5 (2023), ISSN: 0253-7214. (**Web of Science**)

[8] **Muneer V.K**, Mohamed Basheer K.P., A Collaborative Destination Recommender Model in Dravidian Language by Social Media Analysis, Lecture Notes in Networks and Systems (LNNS,volume 572) (March 2023), pp 541–551, https://doi.org/10.1007/978-981-19-7615-5_45. (**Scopus Indexed**)

Conferences

[9] Presented a paper titled - A Collaborative Destination Recommender Model in Dravidian Language by Social Media Analysis, 3rd International Conference on Data Analytics and Management (ICDAM-2022), organized jointly by The Karkonosze University of Applied Sciences, Poland in association with the University of Craiova Romania on 25th – 26th June 2022.

[10] Presented a paper titled - Online Malayalam Script Assortment and Pre-Processing For Building Recommender Systems. 5th International Conference on Smart Computing and Informatics (SCI-2021) September 17 - 18, 2021, Vasavi College of Engineering, Hyderabad, India.

[11] Presented a paper titled – Natural Language Processing of Malayalam Text for Predicting its Authencity. International Conference on Emerging Trends in Signal Processing September 24-26, 2021.

[12] Presented a paper titled - Customized Datamining and Part of Travelogue Tagging of Malayalam Texts for a Recommender Model, International Conference on Innovations and Recent Trends in Computer Science, 25 March 2022.

[13] Presented a paper titled - E2E Accent-Robust ASR for Low Resourced Malayalam Language: A Feature-Based Investigation of LSTM-RNN and ML Approaches presented at the International Conference on Computing and Communication Networks (ICCCN-2022) jointly organized by Manchester Metropolitan University, Manchester, United Kingdom, 20th November 2022.

[14] Presented a paper titled - Collaborative Travel Recommender Model Based on Malayalam Travel Reviews. 3rd International conference on Artificial Intelligence and Speech Technology AIST2021. 13 Nov 2021, IGDTUW, Delhi.