TEXT SUMMARIZATION FOR ROMAN URDU

Researcher

Student Name: Laraib Kaleem
Enrollment No: 1-249212-005

Supervisors

Supervisor: Dr. Arif-ur-Rahman
Co-Supervisor: Dr.Momina Moetesum

A thesis submitted in fulfillment of the requirements for the award of a degree of Masters of Science (Computer Science).

Department of Computer Science

BAHRIA UNIVERSITY, ISLAMABAD

OCTOBER 2023

Approval of Examination

Scholar Name:	**Laraib Kaleem**
Registration Number:	**75939**
Enrollment:	**1-249212-005**
Program of Study:	**Ms Data Science**
Thesis Title:	**Text Summarization for Roman Urdu**

It is to certify that the above scholar's thesis has been completed to my satisfaction and, to the best of my belief, its standard is appropriate for submission for examination. I have also conducted a plagiarism test for this thesis using HEC-prescribed software and found a similarity index **17%**. that is within the permissible limit set by the HEC for the MS/M.Phil degree thesis. I have also found the thesis in a format recognized by the BU for the MS/M.Phil thesis.

Principal Supervisor Name: **Dr.Arif-ur-Rahman**
Principal Supervisor Signature:
Date: **October 24, 2023**

Author's Declaration

I, **Laraib Kaleem** hereby state that my MS/M.Phil thesis titled is my own work and has not been submitted previously by me for taking any degree from Bahria University or anywhere else in the country/world. At any time if my statement is found to be incorrect even after my graduation, the University has the right to withdraw/cancel my MS/M.Phil degree.

Name of Scholar: **Laraib Kaleem**

Date: **October 24, 2023**

Plagiarism Undertaking

I, solemnly declare that the research work presented in the thesis titled **Text Summarization for Roman Urdu** is solely my research work with no significant contribution from any other person. Small contribution/help wherever taken has been duly acknowledged and that complete thesis has been written by me. I understand the zero-tolerance policy of the Higher Education Commission (HEC) and Bahria University towards plagiarism. Therefore I as an Author of the above-titled thesis declare that no portion of my thesis has been plagiarized and any material used as reference is properly referred to/cited. I undertake that if I am found guilty of any formal plagiarism in the above-titled thesis even after the award of MS/M.Phil degree, the university reserves the right to withdraw/revoke my MS/M.Phil degree and that Higher Education Commission (HEC) and the University has the right to publish my name on the Higher Education Commission (HEC) / University website on which names of scholars are placed who submitted plagiarized thesis.

Name of Scholar: **Laraib Kaleem**

Date: **October 24, 2023**

Dedication

I, **Laraib Kaleem** dedicate this artistic endeavor to my cherished parents, family, friends, and my present supervisor, **Dr.Arif-ur-Rahman**, as well as my co-Supervisor, **Dr.Momina Moetesum**

Acknowledgements

At the outset, I was thankful to Allah for the wisdom and perseverance that he has bestowed upon me and for the strength and prospect of completing this thesis.

I would certainly like to acknowledge my family, parents, and siblings for their unwavering financial and psychological assistance throughout my degree. I have no words to express my gratitude to my beloved parents, who supported me at every step. Without their moral and emotional commitment, I would not have been able to wrap up my master's dissertation.

Regards, in particular, to my friends, who helped and supported me constantly during the entire year.

Furthermore, other individuals, including juniors, professors, academicians, researchers, and my fellow graduates, provided their thoughts, vision, and collaboration, which greatly aided me.

I express my gratitude to Dr. Momina Moetesum, my co-supervisor, for her encouragement and assistance during my thesis research.

I would like to acknowledge the present supervisor, Dr. Arif-ur-rahman, for their assistance, instructions, and advice during the research.

Finally, with the acknowledgment of all, I have accomplished this astonishingly demanding task. Especially for those whose prayers have been the decisive source of accomplishment for me.

Abstract

Recent research has shown that multilingual languages are used in roman form over generations. Due to this complex challenge, we are working on a Roman Urdu (RU) in terms of Abstractive Text Summarization (ATS). Roman Urdu (RU) is gathered from news articles. This paper restricts ground truth for Roman-Urdu summaries. Therefore, we used two ways to achieve different tactics. The first was a manual approach to transliterating the dataset into Roman Urdu (RU) by using tools, and for achieving baseline, we approached Google Bard to generate baseline summaries. After that, evaluate the outcomes. The second approach uses transform-based models T5-small and Bert-base-uncased with fine-tuned pretrained models for State-of-the-Art (SOTA) summarization models. For performance evaluation, there are three ways we explored, such as finding similarity to generate baseline results and using the feature extraction Term Frequency-Inverse Document Frequency (TF-IDF) technique to identify performance. And for Natural Language Processing (NLP) phases, we are using tokenization, then punctuation, and after that, loanwords are converted into the desired format to use in the models. However, as a predicted model, accuracy is not the best approach to evaluate, so for this purpose, we also identify intrinsic [1] and extrinsic [2] evaluations to find out the predicted fallout and also identify the model's training and testing losses.

Keywords: *Baseline, Roman Urdu (RU), Natural Language Processing (NLP) , Abstractive Text Summarization (ATS), State-of-the-Art (SOTA).*

[1]Intrinsic: It measures the quality of the summary without considering how the summary is used.

[2]Extrinsic: It measures the quality of the summary based on how it is used.

TABLE OF CONTENTS

LIST OF TABLE

LIST OF FIGURE

LIST OF ACRONYMS

AI Artificial Intelligence

ATS Abstractive Text Summarization

ANLP Advanced Neural Language Processing

BBC British Broadcasting Corporation

BLEU Bilingual Evaluation Understudy

BERT Bidirectional Encoder Representations from Transformers

DL Deep Learning

DeBERTa Decoding-enhanced BERT with Disentangled Attention

ETS Extractive Text Summarization

ELECTRA Efficiently Learning an Encoder that Classifies Token Replacements Accurately

GPT Generative Pre-trained Transformer

HEC Higher Education Commission

LLM Large Language Model

ML Machine Learning

MT Machine Translation

mBART Multilingual Bidirectional Encoder

METEOR Metric for Evaluation of Translation with Explicit Ordering

NLP Natural Language Processing

POS Part-of-Speech

QA Question Answer

RU Roman Urdu

RoBERTa Robustly Optimized BERT

ROUGE Recall Oriented Understudy of Gisting Evaluation.

SA Sentiment Analysis

SOV Subject Object Verb

SOTA State-of-the-Art

Seq2Seq Sequence to Sequence

SQuAD Stanford Question Answering Dataset

SQuAD1.0 Stanford Question Answering Dataset

TS Text Summarization

T5 Text-to-Text Transfer Transformer

TF-IDF Term Frequency-Inverse Document Frequency

UQuAD Urdu Question Answering Dataset

UQuAD1.0 Urdu Question Answering Dataset

CHAPTER 1

INTRODUCTION

1.1 Background Summary

Text summarization is the process of shortening the length of publications and documents. It is a common problem-solving technique. Extractive Text Summarization (ETS) techniques concatenate essential sentences or paragraphs without understanding the meaning of those sentences. This approach merely identifies the sentences and phrases in the articles that provide important, beneficial information about the main area mentioned in the content. However, this methodology is unfavorable since its performance differs significantly from the methods employed by humans to compress and analyse various papers and articles [1]. Abstractive summarization is the generation of a meaningful summary. On the other hand, it produces a human-like summary that includes selecting, rearranging, and summarizing phrases. However, this methodology is challenging for robots to accomplish automatically and alone. This approach uses more Advanced Neural Language Processing (ANLP) techniques to generate new sentences by learning from the original text. It is a complex task and requires heavy computing power, such as a GPU [2].

Language corpora are beneficial for a diversity of natural language processing strategies. The intricacy of natural language structures complicates this undertaking. Up to this point, the majority of analysis has concentrated on expensive resource languages; similarly, past work on Abstractive Text Summarization (ATS) has concentrated on high-resource languages such as English, because of a shortage of datasets for low- and middle-resource languages.

The Roman script uses English language characters. In South Asian nations such as Pakistan and India, Roman Urdu (RU) are often used on numerous social media sites and messaging applications. Roman Urdu and Urdu are two scripts for writing Urdu on social networking, respectively. However, Roman Urdu is a resource-constrained language, which implies that there is no sin-

gle corpus, tool, or approach for creating large pre-trained language models and performing out-of-the-box Natural Language Processing (NLP) tasks on Roman Urdu (RU) [3].

Roman Urdu uses the alphabet and characters of the English language, making it more versatile than Urdu in terms of reading, writing, and comprehension. Anyone with a basic understanding of English may read Roman Urdu content.

Additionally, no authoritative lexicon of Roman Urdu can be used to establish if a vocabulary is legitimate or incorrect. In a similar vein, there are no rules for proper sentence construction. The Urdu script is more challenging to write and comprehend than Roman Urdu since it has its alphabet, lexicon, and syntax. Nearly identical Roman script that billions of individuals in Pakistan, India, and other areas of the world can read and understand. A comparison of both scripts is given in Figure 1.1 [4].

Urdu is an Indo-Aryan language widely spoken in South Asia. It is widely spoken around the world because of the vast South Asian Diaspora. Urdu is spoken by millions of people worldwide. It is written in a modified Perso-Arabic script from right to left. To be properly viewed, it requires appropriate rendering. It is usually written in nastalique, a very sophisticated and context-sensitive writing technique. It has a complex morphology that incorporates grammatical forms and vocabulary from Arabic, Persian, and other South Asian native languages.

Features	**Roman Urdu**	**Urdu**
Alphabets characters	26 as the English	38
Font style	English	Nastaleeq
Grammars	No	Yes
Dictionary	No	Yes
Word order	No	Yes
Easy to type	Yes	No
Easy to read and understand	Yes	No

Figure 1.1: Comparison of Urdu with Roman Urdu language text.

Urdu does not use capitalization. This makes it harder to discern proper names, titles, acronyms, and abbreviations. Diacritics (vowels) are scarce in the text, and words are assumed based on the context of adjacent words. It features a free word order in terms of syntax Subject Object Verb (SOV).

Despite being spoken by millions of people; Urdu is a language with limited resources [5] [6].

For understanding examples in Urdu with English and Roman Urdu translations are shown in Figure 1.2. Urdu language characters are utilized in the Urdu script. Urdu is the national language of Pakistan and the official language of six Indian states. Previously, academics ignored Urdu due to its unique morphology, distinctive traits, and scarcity of linguistic resources. The writing script is the primary distinction between Urdu and Hindi. The Roman scripts of both languages, however, are identical. The written form of the Korean language is very comparable to the syntax of Urdu.

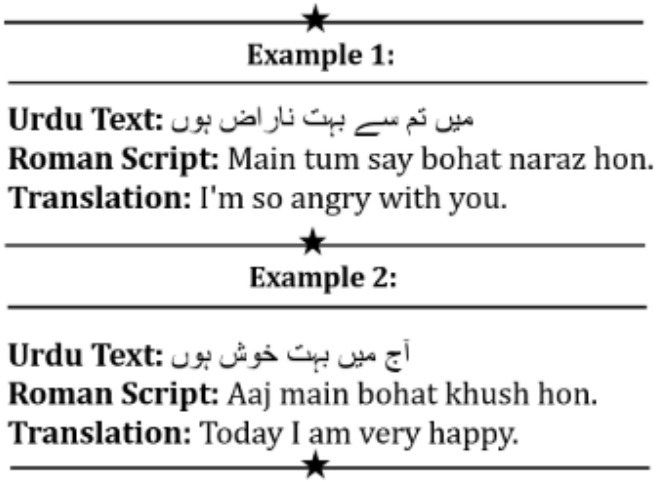


Figure 1.2: Differentiate between Transliteration and Translation.

In research mainly focus is on Abstractive Text Summarization (ATS) and also summaries based on monolingual form. While working the is also cross-lingual involved because the dataset is in Urdu.

1.2 Problem Analysis

In a nutshell, the motivation for this project sprang from the reality that there is a wealth of material in high-resource languages but few references in Urdu. The Roman Urdu (RU) situation is likewise the same as Urdu. Unfortunately, only a small amount of information published in Roman Urdu (RU) uses Abstractive Text Summarization (ATS), such as social media evaluations, comments, or discussions. We are, however, dealing with a massive volume of

material, including essays, films, articles, dissertations, news items, reports, and cases, among other things.

The research challenges in Roman Urdu are that it does not have a standard script; because of this, several Natural Language Processing (NLP) tasks do not perform well, including text summarization. Our research aims to create a technique that can summaries the Roman Urdu language using information obtained from multimedia sources.

1.3 Research Objectives

The research objectives of this dissertation are:

- To generate a Abstractive Text Summarization (ATS) dataset for Roman Urdu (RU), which is our ground truth.
- To assess the performance of existing State-of-the-Art (SOTA) summarization models on our data and identify baseline results.
- To design a model that can be effectively capable of generating an Abstractive Text Summarization (ATS) for Roman Urdu (RU).
- To evaluate the performance of our models and compare it with the State-of-the-Art (SOTA) models.

1.4 Significance of the Study

The research was carried out to understand how this application will be used by future generations. The study is significant since the present generation is involved in this research, which will have a tremendous influence on the following generation.

This perception helped us to scrutinize the fact that future generations are not going to give any appreciation for their national languages; they used easy ways to communicate and solve problems without facing challenges. As for our majority mindset, we gave preference to English languages more than to our native language. Due to this problematic situation, we concluded that we should work on this research.

1.5 Thesis structure

Thesis Structure The remaining portions of the thesis are structured as follows:

In Chapter 2 (Related Work), we provided details regarding all prior research publications, journals, and conferences in which we produced a comprehensive examination of Roman Urdu and ATS and emphasized the essential work that was going on in this research.

In Chapter 3 (Methodology), we propose the methodology of existing state-of-the-art techniques and different approaches according to their requirements, with an explanation.

In Chapter 4 (Analysis and Results), we demonstrate our experimenters and outcomes based on the way they performed, which we examine.

In Chapter 5 (Conclusion and Future Work), we precise our overall workings, presumptions, judgments, and future evolutions.

CHAPTER 2

RELATED WORK

2.1 Literature Review

Numerous studies over the past eight centuries have concentrated on a variety of domains, including Sequence to Sequence (Seq2Seq) language modeling, Question Answer (QA), Sentiment Analysis (SA), reading comprehension, polarity detection, text formation, summarization, and so on. Our domain, which specializes in text summarization, is also centered on one of the fields that has been successful for decades. Diverse portions, including document, extraction, abstraction, and extreme, exist for this endeavor. As well as current trends that have an emphasis on several summarizations that are monolingual [1], bilingual [2], multilingual [3] and cross-lingual [4].

2.1.1 Text Summarization

Last decades, accessible datasets for text summarization challenges have been formed; we describe them in this section. Some of the datasets are LCTS (Hu, 2016) [7], which introduces a sizable dataset of publicly available Chinese short text summarization datasets derived from the Sine Weibo microblogging platform. More than 2 million authentic short Chinese texts are included in this dataset, each with a summary from the author. Furthermore, 10,666 summaries' relevance was manually tagged with the corresponding short texts.

English Wikipedia article generation may be viewed as a multi-document summary of source papers, as shown in the WikiSum article (Saleh, 2018) [8]. They

[1]Monolingual individual understands or can utilize just one language

[2]Bilingual refers to an author's ability to negotiate in two languages.

[3]Multilingual ability to converse in several than bilinguals, or (of an object) narrated either spoken in enough than bilinguals

[4]Cross-lingual means creating a summary in one language (for example, English) for the provided document(s) in another language (e.g., Chinese)

created the article using a neural abstractive model and extractive summarization to broadly identify salient information. In addition, Newsroom (Grusky, 2020) [9] made use of between 1998 and 2017, writers and editors submitted 1.3 million articles and summaries. To develop extractiveness measurements and apply them to split data into extractive, mixed, and abstractive groups. They focused on both text summarization methods. The BookSum dataset (Rajani, 2021) [10] is a data resource collection for long-form narrative summarizing in English. The hierarchical nature of the dataset, which contains aligned paragraph, chapter, and book-level data, makes it a potential target for single-document and multi-document summarization algorithms. Their dataset will help to progress the field of automatic text summarization. We are particularly interested in abstract text summarizing since it focuses on producing a summary of the input text in a paraphrased fashion that takes into account all information. This is considerably different from what we observe with extractive summarizing; as abstractive summarization produces a succinct summary of everything rather than a paragraph made up of each "best phrase."

2.1.2 Abstractive Text Summarization for Other Linguistic

Datasets that are relevant for Abstractive text summarization challenges have emerged globally, and we will explore them in this section. CNN/Daily Mail (Nallapati, 2016) [11] is a dataset for text summarizing, according to research. This is the case. Human-produced abstractive summary bullets were constructed from CNN and Daily Mail online news items as questions (with one of the entities obscured), and stories as the relevant sections from which the system is supposed to answer the fill-in-the-blank inquiry. The programmers that crawl, extract, and produce pairs of excerpts and questions from these websites were released by the authors. This study introduces extreme summarization, a novel single-document summary challenge that does not favor extractive tactics and instead asks for an abstractive modeling approach XSum (Narayan, 2018) [12]. Their purpose is to develop a one-sentence news summary that responds to the inquiry, "What is the article about?". For this assignment, they gather a real-world, large-scale dataset by gathering internet articles from the British Broadcasting Corporation (BBC). WikiHow (Koupaee, 2018) [13], this research presents a dataset of over 230,000 article and summary pairs collected and built from an online knowledge base produced by various human writers. The papers cover a wide variety of themes and hence represent a wide range of styles. Dataset for Summarization on arXiv (Cohan, 2018) [14]

Neural abstractive summarization methods have generated encouraging results when evaluating relatively brief materials. It introduced the first abstractive summarization methodology for single, longer-form publications (e.g., research papers). On Reddit, TIFU (Kim, 2019) [15] took on the subject of abstractive summarization from two perspectives: by offering a fresh dataset and a new model. To begin, gather the Reddit TIFU dataset, which consists of 120K Reddit online discussion forum entries. and use such unstructured crowd-generated postings as text sources, in contrast to existing datasets, which mostly use formal documents, such as news items, as sources.

Corpus of Urdu Synthesis (Humayoun, 2016) [5] Due to this issue, there are inadequate resources (under-resourced) to develop a benchmark corpus. They obtained the dataset from internet sources with a handwritten summary by selected volunteers who had no constraints. It concentrated solely on abstractive tokenization and developed two versions of the dataset. Pn-summary (Farahani, 2020) [16] document serves as a basis for future Persian language study. They are working on the Abstractive Text Summarization (ATS) framework for the Persian language to attain their aim because there are no acceptable Persian text datasets accessible for this assignment. They propose two ways of dealing with the pn-summary dataset in Persian abstractive text summarization. There are a few datasets available for Text Summarization shown in Table 2.1

They introduced the first multilingual summarization dataset on a big scale, MLSUM (Scialom, 2020) [17]. It has 1.5 million or more articles or summary pairs taken from internet newspapers in five different languages: French, German, Spanish, Russian, and Turkish. The topic of this work was abstractive text summarization. They propose a large-scale Multilingual Summarizing (MLSUM) dataset to solve this gap for the automated summarization job. Global Voices (Nguyen, 2020) [18] was a multilingual dataset created to test cross-lingual summarizing algorithms. By omitting social network descriptors from Global Voices news items, assessment data for into-English and from-English summarization in 15 languages will be gathered. 15 languages, including Romance, Barito, Indic, Slavic, Semitic, Greek, Germanic, Japanese, and Bantoid, are currently supported, representing nine language families and nine language genera (Indo-European, Austronesian, Japanese, Niger-Congo, Afro-Asiatic). Voting outcomes are used to detect languages. WikiLingua (Ladhak, 2020) [19] is a large-scale, multilingual dataset that may be used to evaluate cross-lingual abstractive summarization methods. WikiHow, a high-quality, crowdsourced collection of how-to guides authored by human writers

on a wide range of topics, was used to extract article and summary pairings in 18 languages.

Table 2.1: Dataset for Text Summarization

References	Year	Datasets	Languages	Samples	Focus on	Model	Performance Evaluation
[20]	2022	AHS ANA	Arabic	300k 265k	ATS	AraBART	AHS: R-1: 34.74 R-2: 17.50 R-L: 34.08 R-LSUM: 34.08 ANA: R-1: 85.83 R-2: 70.90 R-L: 85.01 R-LSUM: 85.01
[1]	2021	Roman Urdu	Roman Urdu	30K	ETS	Fuzzy Logic	Recall: 0.99 BLEU: 0.45 Precision: 0.98 F-measure: 0.76
[16]	2021	Pn-Summary	Persian	93207	ATS	BERT	R-1: 44.01 R-2: 25.07 R-L: 37.76
[21]	2021	XL-Sum	Multilingual(44)	1M	ATS	mT5	HR: R-1: 36.99 R-2: 15.18 R-L: 29.64 LR: R-1: 44.55 R-2: 21.35 R-L: 34.43
[10]	2021	BookSum	English	155882	ATS ETS	BART T5	BART: R-1: 29.97 R-2: 6.02 R-L: 10.97 T5: R-1: 39.46 R-2: 7.69 R-L: 13.77
[19]	2020	WikiLingua	Multilingual(18)	770K	ATS	BART	Spanish-En: R-1: 37.16 R-2: 14.25 R-L: 31.04 Turkish-En: R-1: 41.06 R-2: 17.72 R-L: 34.53
[17]	2020	MLSUM	Multilingual(5)	1.5M	ATS	BERT	Spanish: R-L: 20.44 METEOR: 14.92 Turkish: R-L: 32.94 METEOR: 26.26 English: R-L: 35.41 METEOR: 22.16
[15]	2019	Reddit TIFU	English	120K	ATS	MMN [5]	TIFU-short: R-1:20.2 R-2: 7.4 R-L: 19.8 TIFU-long: R-1: 19.0 R-2: 3.7 R-L: 15.1
[14]	2018	ArXiv	English	1314000	ATS	BiLSTMs	R-1: 35.80 R-2: 11.05 R-3: 3.62 R-L: 31.80
[13]	2018	WikiHow	English	230000	ATS	Seq-to-seq	R-1: 22.04 R-2: 6.27 R-L: 20.87 METEOR: 10.06
[8]	2018	WikiSum	English	232998	ATS, ETS	LSTM	R-L: 12.7
[12]	2018	XSum	English	226711	ATS	Seq2Seq	LEAD: R-1: 16.30 R-2: 1.61 R-L: 11.95
[7]	2016	LCSTS	Chinese	2M	TS	RNN	Word: R-1: 0.177, R-2: 0.085, R-L: 0.158 Char: R1: 0.215, R2: 0.089, R-L: 0.186

It generates gold-standard cross-language article summary alignments by matching the graphics used to depict each how-to step in an article. English, French, Spanish, German, Russian, Turkish, Czech, Chinese, Korean, Hindi, Thai, Japanese, Arabic, Vietnamese, Italian, Dutch, Indonesian, and Portuguese are among them.

Due to the scarcity of datasets for low- and mid-resource languages, XL-Sum (Hasan, 2021) [21] is an abstractive text summary research that has mostly targeted higher source languages such as English. It provides a massive and diverse dataset of 1 million professionally annotated article-summary pairs obtained from the British Broadcasting Corporation (BBC) using a series of well-defined algorithms. The collection includes 44 languages ranging from low to high resource, with many without a dataset. By human and intrinsic judgment, it is very abstract, concise, and of excellent quality. These are English, French, Spanish, Chinese, Bengali, Japanese, Russian,

[5]Multi-level Memory Networks (MMN)

Portuguese, Amharic, Arabic, Hindi, Indonesian, Korean, Marathi, Persian, Scottish Gaelic, Serbian, Tamil, Telugu, Thai, Turkish, Ukrainian, Urdu, Vietnamese, Welsh, Yoruba, Swahili, Azerbaijani, Gujarati, Hausa, Igbo, Kyrgyz, Burmese, Nepali (macrolanguage), Oromo, Punjabi, Central Pashtun, Cingalese, Somali, Tigrinya, Pidgin, Kirundi and Uzbek. some of the challenging time-consuming to extract vital information from. In [20] automatic text summarizing techniques used lengthy texts to preserved their key information. They are using Arabic Headline Summary (AHS) and Arabic News Articles (ANA) dataset to ever-increasing the demands of textual data. This study explores five State-Of-The-Art (SOTA) Arabic deep Transformer-based Language Models (TLMs) in the task of text summarization by adapting various text summarization datasets dedicated to Arabic. And they compare against deep learning and machine learning-based baseline models has also been conducted. Their Experimental results reveal the superiority of TLMs, specifically the PEAGASUS family, against the baseline approaches, with an average F1-score of 90% on several benchmark datasets

2.1.3 Roman Urdu Linguistic

A growing body of research has geared on a wide range of areas that engage in text summarization and are more interested in anglicized datasets. In linguistics, romanization, or Latinization, is the transfer of text from another writing system to the Roman (Latin) script, or a mechanism for doing so. Transliteration, for expressing written text; transcription, for portraying spoken speech; and mixtures of these are romanization methods. Some are explained in this next section.

Roman Urdu (2017, Rahman) [22] they did a comparative assessment of how social media writing has been standardized to achieve consistency in diverse languages such as Chinese, Arabic, Japanese, Polish, Bangla, Dutch, and Roman Urdu. Based on the lexical normalization of Roman Urdu text using our analytical approach. Sentimental analysis was the focus of their work. This effort is a precursor to a larger undertaking that involves sentiment analysis based on conversation using Roman Urdu datasets [6]. To achieve this goal, they were required to first collect a big data corpus in Roman Urdu (RU) from social media networks. Following that, the raw data was cleaned, and lexically standardized for standard word representation, Part-of-Speech (POS) tagging was conducted so that the words could be tokenized meaningfully, and lastly, the existence or absence of a discourse element was discovered. They are now pre-

[6] `https://archive.ics.uci.edu/ml/datasets/Roman+Urdu+Data+Set`

pared to do Neural Network-based sentiment Analysis on Roman Urdu (RU) (2018, Sharf) [23] datasets using conversation. Due to the ramifications for an inclusive society regarding race, gender, and religion, RUSHOLD (Rizwan, 2020) [24] was a technique that automatically identified hate speech and profanity in social media content. The vast bulk of research in this area, however, is done in English, which limits its applicability to some cultures. Even though Roman Urdu (RU) is commonly used, there aren't enough annotated datasets, language models, or language resources available for this project. The goals of this research are to (1) offer a Roman Urdu lexicon of hateful phrases; and (2) produce the annotated dataset RUHSOLD, which comprises 10, 012 tweets in Roman Urdu and is divided into coarse- and fine-grained categories of hate speech and offensive language.

2.1.4 Urdu Linguistic

A rapidly rising dataset of research has adopted a variety of categories that participate in text summarization and are particularly interested in Latinized Urdu datasets. Some linguistic concepts are discussed in this section. The visual representations of some corpora in Table 2.2 for Urdu.

The Urdu Sentiment Corpus (USC) (Khan, 2020) [25], a dataset made up of tweets that foster rivalry between two distinct political parties and the Pakistani government, is one of the many Urdu linguistic paradigms. This study discusses visual insights into literary similarities, multiple learning, and other topics from document level to word level. This research also presents Part-of-Speech (POS) wise analysis and a straightforward method for extracting sentiment lexicons from corpora. With the identification of average textual similarities using the Sorensen-Dice coefficient and Tanimoto similarity with the Tversky index as a parameter, they propose sentiment analysis and classification in Urdu. This little dataset uses the Romanized form of the Urdu language. The Romanized version of the Urdu language is used in this little dataset. The innovative Urdu dataset for fake news detection, together with a baseline classification and evaluation of it, are described in Bend the Truth (Amjada, 2020) [26].

The challenge of quickly identifying fake news in multilingual digital media becoming increasingly pressing as Internet usage grows around the globe and the impact created by the availability of confusing information significantly increases. To evaluate automatic fake news detection techniques in Urdu, they provide a human-collected and validated dataset of 900 news articles, 500 of which have been classified as real and 400 as fraudulent. The news

articles in the authentic subset were manually examined to make sure they came from trustworthy news sources. The acknowledged challenge of identifying fake news in the fake subset was overcome by recruiting experienced Urdu-speaking journalists who were given explicit instructions to purposefully produce false news reports. The dataset consists of five unique topics: the first four are business, health, show business, sports, and technology. Urdu Question Answering Dataset (UQuAD) (Kazi, 2021) [27] using human-generated samples from Wikipedia articles and Urdu RC worksheets from Cambridge O-level books along with machine-translated Stanford Question Answering Dataset (SQuAD), this study examines the semi-automated production of the UQuAD1.0. Urdu Question Answering Dataset (UQuAD1.0) is a large Urdu dataset including 49k question-answer pairs organized in a question, passage, and response structure for extractive machine reading comprehension tasks. To construct the 45000 pairs of Question Answer (QA) for UQuAD1.0, the original Stanford Question Answering Dataset (SQuAD1.0) was machine translated, and around 4000 pairs were crowdsourced. Online Reviews in Urdu (Safder, 2021) [28] In this essay, understanding consumer behavior is utilized to develop marketing tactics. They are developing a deep learning model for the emotions transmitted in this under-resourced language using an open-source dataset of 10,008 assessments from 566 online debates on sports, gastronomy, software, politics, and entertainment. This effort has two goals: (a) to construct a dataset annotated by humans for the study of sentiment analysis in Urdu, and (b) to utilize a dataset to assess current model performance.

CC100 (Conneau, 2020) [29] shows that pretraining multilingual language models at scale yields significant performance improvements for a range of cross-lingual transfer tasks. It trains a Transformer-based masked language model on one hundred languages using more than two terabytes of filtered Common Crawl data. It also includes a thorough empirical examination of the key factors required to achieve these benefits, such as the trade-offs between (1) positive transfer and capacity dilution and (2) the scale performance of high and low-resource languages.

Finally, they demonstrate for the first time the potential of multilingual modeling without losing per language performance; on the GLUE and XNLI benchmarks, XLM-R is quite competitive with powerful monolingual models. Adversarial and Multilingual Meaning in Context (Qianchu Liu1, 2021) [30] is required for effectively developing multilingual and cross-lingual text representation models for 14 language pairs to interpret word meaning in cross-lingual situations. They employ WiC, XL-WiC, and MCL-WiC datasets since they are obtained from succinct dictionaries. Overall, the information provided above

leads us to the conclusion that we need to work on this thesis, and the next part emphasizes the flaws.

Table 2.2: Dataset for Linguistic

References	Year	Datasets	Languages	Samples	Tasks	Resources Related	Sufficient	Available
[31]	2022	RUECD	Roman Urdu	30K	Sentiment Analysis	Not Related	Enough	Obtainable
[23]	2021	Roman Urdu	Roman Urdu	20000	Sentiment Analysis	Not Related	Enough	Obtainable
[30]	2021	AM2iCo	Multilingual(14)	1500	Word Meaning	Not Related	Enough	Obtainable
[28]	2021	Online Reviews	Urdu	60M	Sentiment Analysis	Not Related	Enough	Obtainable
[27]	2021	UQuAD	Urdu	45000	Machine Reading Comprehension (QA)	Not Related	Enough	Unknown
[24]	2020	RUSHOLD	Roman Urdu	10000	Hate Speech and Offensive Language	Not Related	Enough	Obtainable
[29]	2020	CC100	Multilingual(100)	25B	Language Modelling Cross-Lingual Transfer	Unknown	Enough	Not Obtainable
[26]	2020	Bend the Truth	Urdu	900	Fake News Detection	Not Related	Enough	Obtainable
[25]	2020	Urdu Sentiment	Urdu	17185	Sentiment Analysis Polarity Detection Corpus	Not Related	Enough	Obtainable

2.2 Research Gap

As previously said, datasets are a subset of the many that we explored, but there are many more that worked in other categories like image, graph, and audio. since a major amount of web material is made of sentences in either English or other languages This is particularly true for social media assessments, comments, and conflicts.

- Due to a shortage of datasets for low and mid-resource languages, previous work on Abstractive Text Summarization (ATS) concentrated on high-resource languages such as English.

- The situation in Roman Urdu (RU) is inefficient since not a lot of study has been done there that is relevant to sentiment analysis. Unfortunately, there is a scarcity of materials published inRoman Urdu (RU); the only material accessible is reviews, comments, or social media debates.

The fact leads to a complicated situation: there are few resources for Roman Urdu (RU) in the text summarising area. In the case of Roman Urdu (RU), however, they were mostly targeted in separate areas. To execute text summarising, the Roman Urdu collection requires a dataset, which we must construct.

CHAPTER 3

METHODOLOGY

This section elaborates on our suggested strategy, which is represented in Figure 3.1. The planned architecture is divided into five stages:

1. Data collection
2. Data Development
3. Datat Cleaning
4. Data Wrangling [1]
5. Training
6. Testing

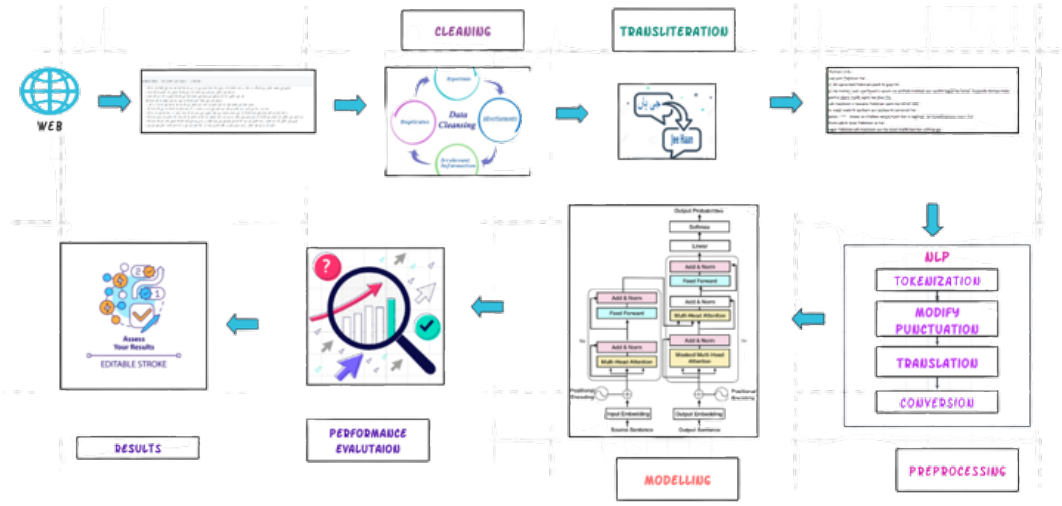


Figure 3.1: Research Proposed Methodology

[1]Dataset Wrangling can be defined as the process of cleaning, organizing, and transforming raw data into the desired format for analysts to use for prompt decision-making.

3.1 Data Collection

The vast majority of the dataset is made up of many datasets, including Wikilingua, Wikimulit, CrossSum, Xsum, XLSum, and MLSum. The majority of them make use of the CNN/DM dataset, although they can also make use of NQA, gigaword, DUC, and other datasets. Low-resource summarization languages are limited due to a lack of datasets. As a result, it leads to the creation of a text summary dataset for Roman Urdu, which also serves as the ground truth for this research.

Table 3.1: Statistics of Datasets

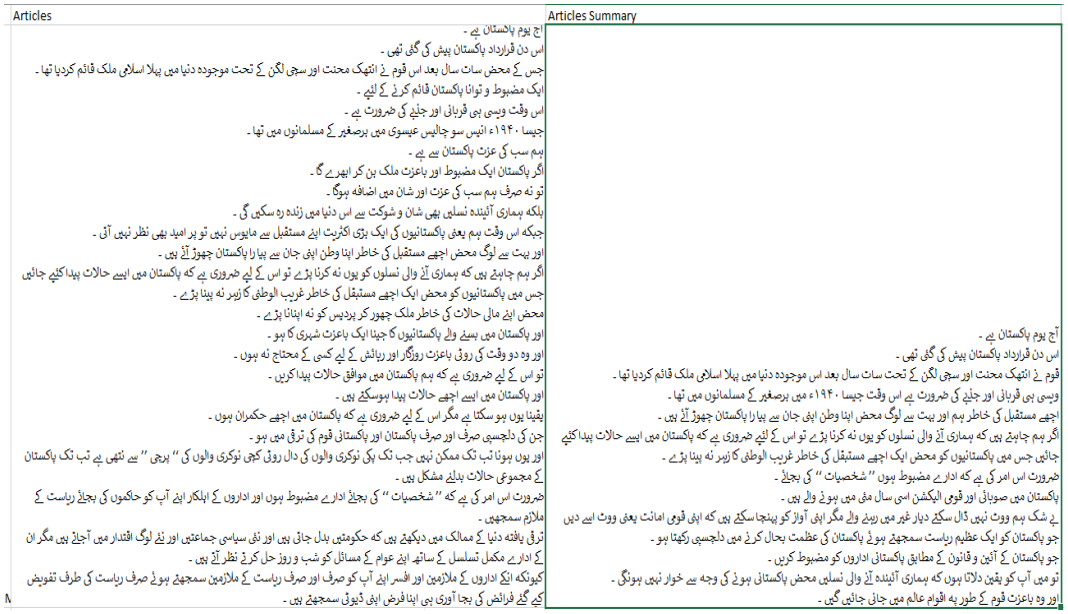
Statistics	**Articles**		**Summaries**	
Category	**Dataset 1**	**Dataset 2**	**Dataset 1**	**Dataset 2**
Total Word Counts	**31986**	**43891564**	**12221**	**2990554**
Average Word Counts	**639.72**	**1478.21**	**244.42**	**102.42**
Length	**50**	**84567**	**50**	**84567**
Data Type	**Text**	**Text**	**Text**	**Text**
Language	**Urdu**	**Urdu**	**Urdu**	**Urdu**

3.1.1 Analysis of the Dataset

Datasets relevant to this study are already detailed in the literature review section. However, datasets on Roman Urdu are not easily accessible, and the few that are available are mostly focused on extractive text summarization. We therefore make use of an Urdu-language dataset whose authors are currently working in or have previously worked in text summarization. As a result, information was gathered from various dataset sources in order to solve this problem. The data was chosen using a variety of online sources, primarily news portals and blogs, as well as a novel dataset. The required dataset is based on two classifications: the original text and a summary of the original text. To achieve abstractive text summarization, consider the Urdu summary dataset in this study, where there must have been two classifications. The Urdu Summary Corpus [11] and the XLSum [21] dataset are two datasets that support this kind of classification.

3.1.2 Quality of Datasets

So far, the Urdu Summary Corpus and the XLSum dataset have been used to generate datasets. These two were chosen because they rely on novel and significant datasets. This kind of information satisfies the research criterion and offers authenticity and truthful facts. These datasets have already undergone preprocessing from raw data into normalized form in accordance with their specifications. Additionally, Table 3.1 provides their statistics.



Articles	Articles Summary
آج یوم پاکستان ہے ۔	
اس دن قرارداد پاکستان پیش کی گئی تھی ۔	
جس کے محض سات سال بعد اس قوم نے انتھک محنت اور سچی لگن کے تحت موجودہ دنیا میں پہلا اسلامی ملک قائم کردیا تھا ۔	
ایک مضبوط و توانا پاکستان قائم کرنے کے لیئے ۔	
اس وقت ویسی ہی قربانی اور جذبے کی ضرورت ہے ۔	
جیسا ۱۹۴۰ء انیس سو چالیس عیسوی میں برصغیر کے مسلمانوں میں تھا ۔	
ہم سب کی عزت پاکستان سے ہے ۔	
اگر پاکستان ایک مضبوط اور باعزت ملک بن کر ابھرے گا ۔	
تو نہ صرف ہم سب کی عزت اور شان میں اضافہ ہوگا ۔	
بلکہ ہماری آئیندہ نسلیں بھی شان و شوکت سے اس دنیا میں زندہ رہ سکیں گی ۔	
جبکہ اس وقت ہم یعنی پاکستانیوں کی ایک بڑی اکثریت اپنے مستقبل سے مایوس نہیں تو پر امید بھی نظر نہیں آتی ۔	
اور بہت سے لوگ محض اچھے مستقبل کی خاطر اپنا وطن اپنی جان سے پیارا پاکستان چھوڑ آئے ہیں ۔	
اگر ہم چاہتے ہیں کہ ہماری آنے والی نسلوں کو یوں نہ کرنا پڑے تو اس کے لیے ضروری ہے کہ پاکستان میں ایسے حالات پیدا کیئے جائیں	
جس میں پاکستانیوں کو محض ایک اچھے مستقبل کی خاطر غریب الوطنی کا زہر نہ پینا پڑے ۔	
محض اپنے مالی حالات کی خاطر ملک چھوڑ کر پردیس کو نہ اپنانا پڑے ۔	
اور پاکستان میں بسنے والے پاکستانیوں کا جینا ایک باعزت شہری کا ہو ۔	
اور وہ دو وقت کی روٹی باعزت روزگار اور رہائش کے لیے کسی کے محتاج نہ ہوں ۔	
تو اس کے لیے ضروری ہے کہ ہم پاکستان میں موافق حالات پیدا کریں ۔	
اور پاکستان میں ایسے اچھے حالات پیدا ہوسکتے ہیں ۔	
یقینا یوں ہو سکتا ہے مگر اس کے لیے ضروری ہے کہ پاکستان میں اچھے حکمران ہوں ۔	
جن کی دلچسپی صرف اور صرف پاکستان اور پاکستانی قوم کی ترقی میں ہو ۔	
اور یوں ہونا تب تک ممکن نہیں جب تک بکی نوکری والوں کی دال روٹی کبھی نوکری والوں کی " پرچی " سے نتھی ہے تب تک پاکستان	
کے مجموعی حالات بدلنے مشکل ہیں ۔	
ضرورت اس امر کی ہے کہ " شخصیات " کی بجائے ادارے مضبوط ہوں اور اداروں کے اہلکار اپنے آپ کو حاکموں کی بجائے ریاست کے	
ملازم سمجھیں ۔	
ترقی یافتہ دنیا کے ممالک میں دیکھتے ہیں کہ حکومتیں بدل جاتی ہیں اور نئی سیاسی جماعتیں اور نئے لوگ اقتدار میں آجاتے ہیں مگر ان	
کے ادارے مکمل تسلسل کے ساتھ اپنے عوام کے مسائل کو شب و روز حل کرنے نظر آتے ہیں ۔	
کیونکہ انکے اداروں کے ملازمین اور افسر اپنے آپ کو صرف اور صرف ریاست کے ملازمین سمجھتے ہوئے صرف ریاست کی طرف تفویض	
کیے گئے فرائض کی بجا آوری ہی اپنا فرض اپنی ڈیوٹی سمجھتے ہیں ۔	
	جو یوم پاکستان ہے ۔
	اس دن قرارداد پاکستان پیش کی گئی تھی ۔
	قوم نے انتھک محنت اور سچی لگن کے تحت سات سال بعد اس موجودہ دنیا میں پہلا اسلامی ملک قائم کردیا تھا ۔
	ویسی ہی قربانی اور جذبے کی ضرورت ہے اس وقت جیسا ۱۹۴۰ء میں برصغیر کے مسلمانوں میں تھا ۔
	اچھے مستقبل کی خاطر ہم اور بہت سے لوگ محض اپنا وطن اپنی جان سے پیارا پاکستان چھوڑ آئے ہیں ۔
	اگر ہم چاہتے ہیں کہ ہماری آنے والی نسلوں کو یوں نہ کرنا پڑے تو اس کے لئیے ضروری ہے کہ پاکستان میں ایسے حالات پیدا کئیے
	جائیں جس میں پاکستانیوں کو محض ایک اچھے مستبقل کی خاطر غریب الوطنی کا زہر نہ پینا پڑے ۔
	ضرورت اس امر کی ہے کہ ادارے مضبوط ہوں " شخصیات " کی بجائے ۔
	پاکستان میں صوبائی اور قومی الیکشن اسی سال مئی میں ہونے والے ہیں ۔
	بے شک ہم ووٹ نہیں ڈال سکتے دیار غیر میں رہنے والے مگر اپنی آواز کو پہنچا سکتے ہیں کہ اپنی قومی امانت یعنی ووٹ اسے دیں
	جو پاکستان کو ایک عظیم ریاست سمجھتے ہوئے پاکستان کی عظمت بحال کرنے میں دلچسپی رکھتا ہو ۔
	جو پاکستان کے آئین و قانون کے مطابق پاکستانی اداروں کو مضبوط کریں ۔
	تو میں آپ کو یقین دلاتا ہوں کہ ہماری آئیندہ آنے والی نسلیں محض پاکستانی ہونے کی وجہ سے خوار نہیں ہونگی ۔
	اور وہ باعزت قوم کے طور پہ اقوام عالم میں جانی جائیں گیں ۔

Figure 3.2: Dataset 1.

Urdu Summary Corpus (USC)

USC [2] was gathered from various sources focusing on news and blogs with specific criteria being real text written by native speakers from various backgrounds and compared to online printed media. The essential information is covered in their summary, but the writer's perspective on the original text is not included. They chose a group of volunteers to write summaries. These volunteers are native Urdu speakers who are professors at universities and students majoring in Urdu literature.

The summary writing criteria were that they did not impose any size restriction on human-written summaries, which makes it difficult to compare with DUC summary datasets in their research. Summaries were requested from writers without consideration for their size—small, medium, or large—but there were

[2] `https://github.com/humsha/USCorpus`

a few restrictions, including the requirement that no single summary exceed half the length of an article. They also followed three basic steps:

1. Identify the key phrases in a text after reading it.

2. If necessary, use these key phrases at the sentence level.

3. If necessary, insert sequential markers in between to create a proper flow.

The six editing operations in human abstracting that influence these steps are sentence reduction, sentence combination, syntactic transformation, lexical paraphrasing, generalization and specification, and reordering. The quality of these human-written summaries was evaluated on a scale of 1 to 5 by five peer contributors for each article summary. 1 represents a very poor summary, 2 a poor summary, 3 an adequate summary, 4 a good summary, and 5 an excellent summary.
When assigning scores, they consider peer contributors in terms of the following aspects:

1. Is the summary grammatically correct?

2. Is the summary non-repetitive?

3. Is the summary free of anaphora and other references?

4. Is the summary well-structured and coherent?

The average scores provided by peer contributors ranged from 3.8 to 4.8. Figure 3.2 provides an example article-summary pair.

Text	Summary
سرکاری عمارتوں پر قبضے کا واقعہ اس وقت پیش آیا ہے جب ایک روز قبل ہی روس کی حمایت کرنے والوں اور یوکرین کے نئے رہنما کے حمایتوں کے درمیان تصادم ہوا حکام کے مطابق کرائمیا کے دارالحکومت سمفروپول میں مسلح افراد نے سرکاری عمارتوں پر قبضہ کر کے روسی جھنڈا لہرا دیا ہے۔ مقامی حکومت کا کہنا ہے کہ وہ مسلح افراد کے ساتھ مذاکرات کر رہی ہے۔ سرکاری عمارتوں پر قبضے کا واقعہ اس وقت پیش آیا ہے جب ایک روز قبل ہی روس کی حمایت کرنے والوں اور یوکرین کے نئے رہنما کے حمایتوں کے درمیان تصادم ہوا۔ خیال رہے کہ یوکرین میں روسی زبان بولنے والے متعدد افراد نے یانوکووچ کی برطرفی اور ملک میں یورپ کی جانب جھکاؤ رکھنے والی انتظامیہ لانے کی مخالفت کی تھی۔ دوسری جانب روس بھی یوکرین میں ہونے والی تبدیلیوں کی وجہ سے ناراض ہے، تاہم وزیر خارجہ سرگے لاوروف کا کہنا ہے کہ ان کا ملک یوکرین کے معاملات میں مداخلت نہیں کرے گا۔ سرگے لاوروف نے منگل کو ماسکو میں میڈیا سے بات کرتے ہوئے کہا دوسرے ممالک یوکرین کی صورتِ حال سے فائدہ اٹھانے کی کوشش نہ کریں تاہم ان کا یہ بھی کہنا تھا کہ روس یوکرین میں عدم مداخلت کی پالیسی پر کاربند رہے گا۔ یوکرین میں کئی ماہ تک جاری رہنے والے عوامی احتجاج کے بعد سنیچر کو پارلیمان کے اراکین نے صدر یانوکووچ کے مواخذے کے لیے ووٹ ڈالا تھا۔ یوکرین میں حکومت مخالف مظاہرے گذشتہ سال نومبر میں اس وقت شروع ہوئے تھے جب صدر یانوکووچ نے یورپی یونین کے ساتھ ایک تجارتی معاہدے کو مسترد کرتے ہوئے روس کے ساتھ تجارتی معاہدے کو ترجیح دی تھی۔ ان پرتشدد مظاہروں کے نتیجے میں درجنوں افراد ہلاک ہو گئے تھے۔	یوکرین کے حکام کا کہنا ہے کہ روسی اکثریت والے کرائمیا میں مسلح افراد نے سرکاری عمارتوں پر قبضہ کر لیا ہے جس کے بعد سکیورٹی فورسز کو الرٹ کر دیا گیا ہے۔

Figure 3.3: Dataset 2.

XLSum

The British Broadcasting Corporation (BBC) [3] is used in this dataset to publish news in 43 languages ranging from low-resource to highresource. They dealt with the majority of the 44 languages.

They retrieved articles from the BBC News website. They have a somewhat similar structure, which gives them an advantage in scraping articles from all sites. They also ignored any textual or multimedia content before continuing to work. The BBC typically provides a summary of an entire article in the form of a bold paragraph containing one or two sentences at the beginning of each article. The authors of the articles write these summaries professionally in order to convey the main story in one small paragraph. This is in contrast to the headline, which serves to entice viewers to read the article that is shown as an example article-summary pair from BBC Urdu in Figure 3.3.

On their website, BBC News does not offer an archive or an RSS feed. As a result, they created a crawler that recursively crawls pages, beginning with the homepage, and visits various article links present on each page visited.

The process of automatically collecting article summaries varies depending on the dataset. The CNN/DM dataset (Hermann et al., 2015) [11]and XSum dataset (Narayan et al., 2018) [12] were used to merge bullet point highlights provided with the articles as reference summaries, the first line of the article as the summary, and the rest of the article as the input. The consistent editorial style of the crawled BBC articles made their method of collecting summaries easier. By carefully examining the HTML structures of the crawled pages, they created a number of heuristics to make the extraction more effective:

1. The desired summary must be present within the first two paragraphs of an article.

2. Some text in the summary paragraph must be in bold type.

3. The summary paragraph may include hyperlinks that are not bold. The percentage of bold and hyperlinked text in relation to the total length of the paragraph in question must be at least 95%.

4. Except for the summary and the headline, all texts must be included in the input text (including image captions).

5. The input text should be at least twice the size of the summary.

[3] `https://www.bbc.co.uk/ws/languages`

Any sample that failed to meet these heuristics was discarded. Their strategy for automatic annotation of summaries is similar to XSum in some ways, but they discovered meta-information in many articles in the first line (e.g., author information, date of last modification). As an outcome, the bold paragraphs were used as summaries instead.

They generated articles written by professionals, which is critical for ensuring that the XL-Sum [4] dataset is valuable and can be used by a larger community for abstractive summarization. They did this by conducting quantitative human assessments on a subset of the dataset. They hired three professional annotators to assess the proficiency of the languages in relation to their global speaker population. By selecting "Yes" or "No" in response to the following questions, each evaluator was asked to rate the quality of a chosen sample of the dataset (roughly 250 article-summary pairs):
Property A: Does the summary convey the topic of the article?
Property B: If property A is true, does the summary contain any information that contradicts the article?
Property C: Does the summary contain any information that cannot be inferred from the article if the answer to property A is "yes"? They manually filtered the data to include only Roman Urdu comments and tweets.

3.1.3 Selection of Roman Urdu Summary

The information gathered was manually filtered to include only Urdu articles and the summary. Transliteration was implemented to pursue the Romanized Urdu data.

Overall, we gather these datasets to achieve our objective. As a consequence of the fact that they provided information that was pertinent and met our needs. The information gathered is organised into several documents with varying file formats. The next section goes into great detail about the steps we took to create our dataset.

[4] `https://github.com/csebuetnlp/xl-sum`

3.2 Data Development

To begin with, the information about how we chose and gathered data comes from a variety of sources with varying file formats and dataset histories, as well as how their authors generated it with their unique characteristics. In this section, we will modify the collected dataset to meet our needs. This step involves creating a text summarization dataset for Roman Urdu, which also serves as our ground truth.

The following datasets are required: The dataset we utilized is in Urdu since, as far as we are aware, the dataset in Roman Urdu was not available in the market.

Table 3.2: Statistics of Manipulated Datasets

Statistics	**Articles**		**Summaries**	
Category	**Dataset 1**	**Dataset 2**	**Dataset 1**	**Dataset 2**
Length	**50**	**250**	**50**	**250**
Data Type	**Text**	**Text**	**Text**	**Text**
Language	**Urdu**	**Urdu**	**Urdu**	**Urdu**

3.2.1 Crawling and Scraping the Data

We crawled the internet and downloaded content from websites that contained news, blogs, postings, books, and other content from the business, technology, politics, entertainment, and sports sectors. For abstractive text summarization, we were able to create a Urdu dataset and a Roman Urdu dataset.

The file format is changed to ".csv" before transliteration. As the two datasets utilize various file formats, the XLSUM dataset uses JSON lines (.jsonl) [5], while the USC dataset uses multiple documents with text extensions (.txt).

Significantly improving the analysis, we gathered the data in Excel format (.xlsx) for some adjustments and thereafter transformed all datasets into CSV files. Before preprocessing, each of them has its own unique, distinct file.

Samples: We personally collect a few samples from each dataset for examination. The information in the table 3.2

[5]Jsonl: there is no limit on the size of a JSONL file.

3.2.2 Data Cleaning

At this phase, we traditionally deleted this kind of information from the collection as the information that was acquired included unwelcome components like hyperlinks, advertising, and irrelevant information.

3.2.3 Transliteration

To convert Urdu data into Roman Urdu, we are using transliteration. We are transliterating Urdu in Roman Urdu form from one language into another; however, our linguistic analysis is based on monolingual abstractive text summarizing. The Ijunoon [6] platform is being used to modify our data.

Table 3.3: Transliteration Scripts

Scripts	**Reasons**
Googletrans googletransliterationapi googletransliteration()	**Do not have script for Urdu to Roman Urdu.**
Aksharamukha	**Words are the combination of English and Urdu alphabets.**
Urduhack	**Quality is not good.**

3.2.4 Transliteration Techniques

We explore both techniques to create efficient workflows, but there are no acceptable scripts or APIs to execute automatic efficient transliteration. So this is why we employed a transliteration tool, which is capable of transliterating data. We are ultimately use an automated form. For scripted tables 3.3 show the reason we are not selected automated method for transliteration.

Besides that, there are also certain tools for transliteration, and their specifics are presented in table 3.4.

[6] https://www.ijunoon.com/transliteration/urdu-to-roman/

Table 3.4: Transliteration Tools

Tools	**Ijunoon**	**Translatiz**	**Meaningin.**
Limitation	**5000 Words** **45 Sentences**	**1000 Words**	**More than** **5000 Words.**
State	**Operated**	**Operated**	**Operated**
Performance	**Efficient**	**Good**	**Awful**
Problems	**1. English Words** **Not Converted.** **2. Space Issue.**	**1. Converted RU** **to Urdu Only.** **2. Not Urdu to RU.**	**Remove Words** **from Sentences.**

We utilized Ijunoon due to the fact that they performed better than other platforms as a tool. The data was compiled into one CSV file after transliteration, and we also analyzed the statistics for the transliterated state in the table 3.5.

Table 3.5: Statistics of Transliterated Datasets

Statistics	**Articles**		**Summaries**	
Category	**Dataset 1**	**Dataset 2**	**Dataset 1**	**Dataset 2**
Length	**50**	**250**	**50**	**250**
Data type	**Text**	**Text**	**Text**	**Text**
Language	**RU**	**RU**	**RU**	**RU**

Conclusion: We conducted some work using gapped datasets from several platforms. The truth is that we change data to meet our needs, which are a few carefully chosen samples, eliminate unnecessary information, and apply transliteration using certain procedures. Overall, this piece was completed manually.

Table 3.6: Statistics of Our Datasets

Statistics		
Category	**Articles**	**Summaries**
Length	**300**	**300**
Data type	**Text**	**Text**
Language	**RU**	**RU**

The dataset we created for Roman Urdu during this phase served as our ground truth, and the general statics are displayed in table 3.6 and figure 3.4. The next chapter delves into the details of data preprocessing steps.

Articles	Summaries
Iraq aur Afghanistan mein jari karwaiyon aur wahan par tainaat afwaj ke ilawa androoni tor پرصحت aur Falah ke kamon par honay walay akhrajaat ki bana par Amrici hukoomat ke qarzon mein record izafah sun-hwa hai . un akhrajaat ko poora karne ke liye Amrici wizarat khazana do arab dollar rozana ke hisaab se qarzay haasil kar rahi hai . nama nigaron ko kehna hai ke bandz ka dobarah ajra is baat ka aitraaf hai ke America ko –apne akhrajaat ko poooray karne ke liye dosray zaraye se qarzay darkaar hain . chaar saal qabal Amrici hukoomat ne lambi muddat ke bandz jari karna band kardiye thay. is waqt Amrici maeeshat behtar thi aur is ko fazil budget dastyab tha .	America ne –apne barhatay hue qarzon ko kam karne ke liye aglay baras se tees sala bandz ka dobarah ajra karne ka faisla kya hai .
yeh faisla itwaar ko asso si ation ke salana aam ijlaas mein kya gaya. Dehli ne haal hi mein do hazaar das ke doulat مشترہ khelon ke muqablay munaqqid karne ka vote jeet liya tha. usay canada ke shehar hemilton se muqablay ka saamna tha . doulat mushtarqa ke khail olympics ke baad duniya mein khelon ka sab se bara bain al aqwami muqaabla hai . olympics munaqqid karne ki Dehli ki koshisho ke baray mein tafseelaat jari nahi ki gayeen hain . taham mubasireen ka kehna hai ke Bharti avlmpk asso si ation ka khayaal hai ke doulat mushtarqa khelon ke liye jin sahuliyaat ko taamer kya jaye ga un se is ko avlmpk khail munaqqid karanay ki boli jeetnay mein madad miley sakti hai .	Bharti avlmpk asso si ation ne kaha ke woh do hazaar solah ke avlmpk khail Dehli mein munaqqid karanay ke liye international اومپک committee ko paish kash kere gi .

Figure 3.4: Transliterated Datasets

3.3 Data Wrangling

After completing our ground truth, which was developing our dataset for Roman Urdu text summarising, we are now optimizing the dataset to execute state-of-the-art summarization models.

The idea behind data preprocessing is to examine or model data using a variety of approaches to enhance its quality and usefulness. However, a variety of languages lead to unique form transformations, such as language structure from right to left or left to right, grammar or non-grammar languages, model types employed, and more. As an outcome of these situations, we modify our preprocessing stages to meet our needs.

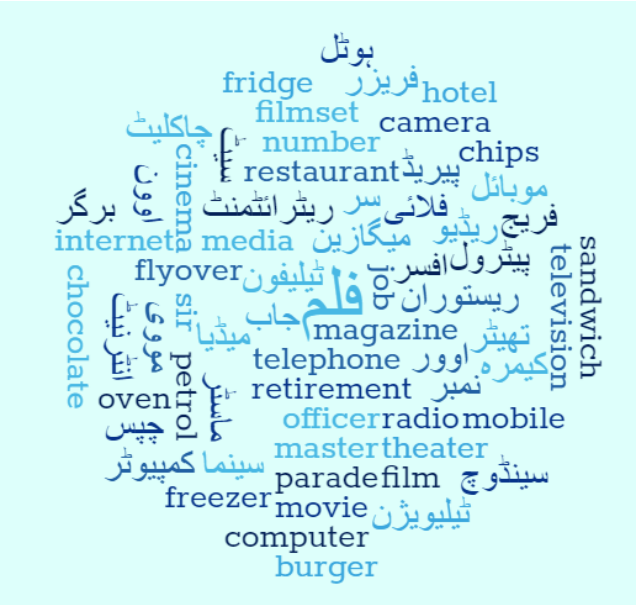


Figure 3.5: Loanwords

In our scenario, we're working with Roman Urdu, a non-grammar language. We altered the following stages in this data preprocessing:

3.3.1 Tokenization

Tokenization is the first step in converting big segments into sentences using NLTK, a common Python module. There are few kinds of tokenization: sentence tokenization, word tokenization and so on. In this technique, we worked on sentence tokenization.

3.3.2 Modify Punctuation

The elimination of punctuation is most commonly employed during the pre-processing phase. There is no deletion of punctuation or unique symbols on this path. The Roman Urdu Dataset contains both English and Urdu punctuation, indicating that distinct characters used in Urdu are not included in the accessible punctuation. Therefore, we just insert white spaces between the tokens and only eliminate double white spaces from the dataset. We don't want to modify the meaning of sentences, and it will be easy to do so while producing summaries.

3.3.3 Machine Translation (MT)

There are specific English terms in Urdu, such as caption, computer, petrol, Lady Diana, and so on. In this stage, the dataset is translated into English arguments using the GoogleTrans module and detecting tokens with Urdu-regex to translate tokens. To clarify, those words are written in Urdu, yet they are English terms referred to as loanwords [7].Also, some of them are shown in figure 3.5

3.3.4 Dataset Structures

The information is transformed into a paragraph in this phase. We apply chosen State-of-the-Art (SOTA) models to the pre-processed dataset after the pre-processing stage.

In numerous research articles that focus on a range of languages as well as in Roman Urdu publications, preprocessing includes a few extra procedures such as stop words, Part-of-Speech (POS) tagging, stemming, and lemmatization. The language used in this study was formed by the fusion of two languages, each with its own set of principles and grammatical structures. There has been no study that has produced any grammar or methods for dealing with these two languages; hence, the processes outlined above do not apply to the present scenario.

[7]Loanwords: These words are taken directly from English and do not have a translation in Urdu.

3.4 Training

In the last chapter, we turned our dataset into a format that is appropriate for implementation into models. At this point, we will be adapting models to our situation.

3.4.1 Baseline

In this step, we review how we gathered baseline results and which approaches were used. We are using the Google Bard platform to generate baseline summaries. After gathering the summaries, we evaluate them by using cosine similarity.

Table 3.7: Model Background

Feature	**Seq2seq**	**Transformers**
Architecture	**General-purpose**	**Specialized for Seq2Seq**
Attention mechanism	**No**	**Yes**
Performance	**Good**	**Excellent**

3.4.2 Overview

Models are essentially representations of anything that are used to learn new routes, forecast future occurrences, and create assumptions. We concentrate on computational models that operate on machine learning systems in their many forms. However, these models were useful in the fields of Machine Learning (ML) and Artificial Intelligence (AI) since they taught data to do a specific job, and deep learning is a subset of a more sophisticated type of Machine Learning (ML).Abstractive Text Summarization (ATS) uses a variety of Machine Learning (ML) models and methodologies, but the most effective approaches have included deep learning models like sequence-to-sequence architectures, pertained language models, encoder-decoder architectures, attention mechanisms, and transformer-based architectures. Our challenge may be solved in a variety of ways, including by using Machine Learning (ML) and Deep Learning (DL). There are distinct models available for both of them.

Transfer learning is a Machine Learning (ML) approach that is used to train on a single job. It is up to the user to decide how it will be utilised, such as by fine-tuning the model's weights or by employing the model as a feature extractor.

Additionally, State-of-the-Art (SOTA) models are models for numerous tasks. That is, it has previous experience working on the matter at hand; therefore, we reuse it at particular moments to train the model to our specifications. Other State-of-the-Art (SOTA) models include the BERT, T5, BART, PEGASUS, and so on.

Table 3.8: Model Parameters

Features	**Parameters**	
Model	**T5**	**BERT**
Small	**220 Million**	**110 Million**
Base	**1.1 Billion**	**340 million**
Large	**11 Billion**	**340 million**

In our study, we utilised transformers due to the fact they can learn long-term connections between words, which helps them comprehend the meaning of a phrase. Transformers are a sort of neural network that is ideal for transfer learning. Transformers can learn long-term relationships between words, which helps them grasp the meaning of a statement.

The human brain inspired neural networks, a sort of machine learning model. They are composed of linked nodes that individually conduct a basic computation. The nodes are organised into layers, and each layer learns how to change data in a meaningful way for the job at hand. It is used for image classification, object identification, natural language processing, speech recognition, machine translation, and medical diagnosis, among other things.

Machine Translation (MT) is a branch of natural language processing that deals with the automatic translation of text from one language to another, and it is presently the most advanced technique for many language pairings. This is also utilised and discussed in the last data processing section.

In summary, deep learning models are utilised for abstractive text summarization, notably those based on the sequence-to-sequence architecture and transformer-based architectures. Because they can handle variable-length inputs and outputs, exploit attention processes, use pre-trained language models, and perform well on large-scale summarising datasets. By creating coherent, contextually relevant, and human-like summaries from larger source texts, these models have made important advances in the field of automatic abstractive summarization.

Pre-trainings[8], fine-tuning, and beam search are his strategies for train the Abstractive Text Summarization (ATS), which uses transformers, seq2seq, and pointer generators as models. It made use of architectures like encoder-decoder, attention, and self-attention.

We utilise transformers in this dataset since they are particularly successful for summarization jobs and due to their function of self-attention, which lets the transformer learn these associations directly from the data, is reasonably straightforward to train, and is also very scalable. It's because self-attention is particularly successful at tasks that involve comprehending the links between distinct sections of the input sequence; it's computationally efficient; and it can handle vast volumes of data. There are several kinds, including the original transformer, the Transformer-XL, the BERT, the GPT-3, and the Roberta. Its strategies include attention, self-attention, encoder-decoder, and others.The most appropriate mechanism depends on the requirement and the information that is accessible. Further information shown in table 3.7.

State-of-the-Art (SOTA) attention is often obtained in abstractive text summarization by fine-tuning a pre-trained transformer model with an attention mechanism, such as BART, T5, or Pegasus, using a dataset of texts and summaries. The attention mechanism enables the model to understand the connections between various portions of the text, which is required for abstractive text summarization. Here are some of the attention-based, pre-trained transformer models that have been demonstrated to attain State-of-the-Art (SOTA) performance in abstractive text summarization.

These models were trained on vast datasets of text and learned to recognise generic language characteristics, create text, and learn the links between different portions of the text. This information may be transferred to a new model to increase its abstractive text-summarising performance.

Pre-trained transfer learning is a strong strategy for improving the performance of abstractive text summarization algorithms. When combined with State-of-the-Art (SOTA) attention models, it can produce cutting-edge performance on a wide range of challenges.

3.4.3 Selection of Model

Following the research, we are applying current State-of-the-Art (SOTA) summarising models to our data, and the transformers we are employing can al-

[8]Pre-trained Model: It is one that has been trained on a huge text and code dataset.

ready generate summaries. One is frequently used, whereas the other is only used once in a blue moon. There are the following transformer models available:

- Bidirectional Encoder Representations from Transformers (BERT)
- Text-to-Text Transfer Transformer (T5)

These models were classified as tiny, small, large, or base, and each had a distinct input and output size as well as a varied summary max length. Each model has its own classificationIn contrast to the T5 models, the Bert model contains two parameters that are classified as uncased and cased. Some of the models are show in table 3.8.

Bidirectional Encoder Representations from Transformers (BERT)

Bidirectional Encoder Representations from Transformers (BERT) is an Natural Language Processing (NLP), Large Language Model (LLM), supervised learning, black box model, and transformer-based model that employs an attention mechanism to understand the connections between distinct portions of the input text.

Its working procedure begins with fine-tuning the model on a dataset, which entails modifying the model's weights to make it more suitable for the purpose of text summarization. Once refined, the model may be used to create summaries for fresh text. BERT and LLM models, for example, have drawbacks in that they need a lot of computer resources to train and fine-tune. In addition, supervised learning requires a large dataset to train. A black box signifies that the model's predictions are difficult to comprehend. And its operation differs depending on the scenario.

Text-to-Text Transfer Transformer (T5)

Text-to-Text Transfer Transformer (T5) is likewise a language processing (NLP) and transformer-based model that can be used to achieve state-of-the-art results on a variety of natural language processing tasks, including text summarization, question answering, and translation. He is also a versatile model that can be fine-tuned to accomplish various jobs. LLM and black boxes, which take a lot of computational resources to train and fine-tune, are the obstacles to adopting T5. Furthermore, it is difficult to understand how the model produces its predictions.

It is dependent on the models we are working on; each model has its own set of methodologies for automation or architecture.

Analysis Before Training

This dataset was updated in its raw form, where it originated from collection and development prior to training, then processed to transform its raw form into platform form to build applications. More details may be found in the experimental procedure in section 4.

3.5 Testing

The experiment described how the dataset was separated into two halves. One component was used to train the models, while the other was used to test them. The testing set was also subjected to preprocessing, normalisation, and embedding procedures.

3.5.1 Evaluation Metrics

In this stage, a testing dataset is sent to each trained model for evaluation, and the outcomes in each case are recorded. The outcomes of all deep learning models were reviewed at this stage by doing a statistical analysis of the test data. The models were evaluated using the following metrics:

- Accuracy
- Intrinsic Metrics
- Extrinsic Metrics

The purpose of abstractive text summarization in natural language processing (NLP) is to provide a compact and cohesive summary of a given piece of text. Accuracy is not commonly employed as a criterion for evaluating abstractive summarising models, owing to the subjective nature of the concept of a "correct" or "accurate" summary. Instead, criteria such as ROUGE or BLEU are widely employed to assess the quality of generated summaries by comparing them to system-provided reference summaries.

In relation to this, when comparing generated and reference summaries, there is a potential for zero if accuracy is measured. There are several abstractive text summarization measures, including BLEU, ROUGE, METEOR, BERT Score, and CIDEr. The optimum metric for assessing an abstractive text summarization system is determined by the application. If the objective is to create as factual summaries as feasible, a measure like BLEU or ROUGE may be more suited. If the purpose is to develop innovative or interesting summaries, a measure like METEOR or CIDEr may be more suited.

When it came to reviewing the material, we decided to utilise these measures. Section 4 contains more information on the evolution of accuracy, intrinsic, and extrinsic.

3.5.2 Visualization Information

Graphical representations are used to interpret the performance of each model. Python includes the Matplotlib and Seaborn libraries for visualisation. To illustrate the findings of the various models investigated in this study, we used Python's Matplotlib module.

Overall, we present an overview of models, types, methodologies, architectures, and processes in this part, as well as tell readers about the approaches employed in this research and the justifications for adopting State-of-the-Art (SOTA) summarization models. Also, discuss the various assessment measures for this aim. There are several tasks, and assessment metrics depend on them, like in this study, where the task is Abstractive Text Summarization (ATS) and the dataset is in text format; both have distinct measurements. In terms of the broader circumstances, we are considering acceptable measurements and models to complete this demanding assignment.

In the next part, there will be thorough information regarding model measurements and outcomes, as well as an analysis of model measurements.

CHAPTER 4

ANALYSIS & RESULTS

4.1 Analysis

In the last chapter, we discussed how to fulfill our goals, create our dataset, and modify the data to match our needs. Following the arrangement of our dataset in a meaningful manner, we perform a data processing phase in which we utilize Natural Language Processing (NLP) processes that are appropriate for the dataset. Extend the State-of-the-Art (SOTA) models and assessment criteria as well. Explain the experiments concerning models and their procedures in this part, as well as exhibit the outcomes and compare them.

4.1.1 Baseline

As the baseline is used to identify whether the quality of the research is growing or not, due to this point of view, there is no research on abstractive text summarization for Roman Urdu. As for this, we use baselines as Google Bard [1] summaries to generate manually to compare the models generated summaries. The bard generated summaries based on key points of text, language, grammar to read and understand, and how the structure of sentences smoothly flows, which means how much accuracy, information, and fluency occur in their summaries.

After gathering the summaries, we evaluate them by using cosine similarity. However, first we have to convert the text format into numeric by using the techniques of feature extraction, which is Term Frequency-Inverse Document Frequency (TF-IDF). This feature extraction approach is being used to determine model performance. We still rely on outdated methods since language

[1]Google Bard: It is a conversational generative artificial intelligence chatbot developed by Google, based initially on the LaMDA family of Large Language Model (LLM) and later on the PaLM LLM.

cannot be transmitted through them; thus, we must ensure that they are neither excellent nor terrible linguistically.

For baseline, we use cosine similarity however, it is a vector space converted into mathematical space form, which is represented by the vectors of text. Cosine similarity calculates the dot product of vectors that show similarity between two vectors. The higher the product indicates, the more similarity there is; otherwise, it will have a lower resemblance. This technique represents each word as a vector of real numbers. The word embeddings are learned from a corpus of text and capture the semantic meaning of the words.

Term Frequency-Inverse Document Frequency (TF-IDF)

It is a statistical measure that is an important term in a document corpus. There is a word used as a term. There is a word used as a term called "frequency" (TF), which means the number of times a term appears in a document. First, it calculates the frequency of each word present in the document. After that, the Inverse Document Frequency (IDF) is the inverse of the number of documents in the corpus that contain the term, which means the TF-IDF score is calculated by multiplying the TF score by the IDF score. A higher TF-IDF score indicates that the term is more important to the document. It is mostly used to measure information retrieval and text mining. It can be used to find relevant documents, summarize text, and identify important terms. It is a powerful tool that can be used to extract meaning from text. It is a versatile measure that can be used for a variety of tasks. The formula for TF-IDF 4.1 is as follows:

$$TF - IDF(term, document) = TF(term, document) \cdot IDF(term) \quad (4.1)$$

$$TF(term, document) = \frac{TotalAppperanceofterminaDocument}{TotalTermsinaDocument} \quad (4.2)$$

$$IDF(term) = log \cdot \frac{TotalNumberofDocumentsinaDocumentSet}{DocumentFrequencyofaTerm} \quad (4.3)$$

Where:
1. Term: Word or Phrase.
2. TF (term, document): Frequency of a Term in a Document.
3. IDF (term): Inverse Document frequency of a Term.

4.2 Operating Environment

We ran our trials using Google's Colab environment, which includes Python 3 resources and a Google Compute Engine backend (GPU) with 15 GB of RAM and 120 GB of storage. Also, we are using the Windows laptop and PC devices, and the laptop characteristics are the Windows 10 Pro with the processor being an 8th Generation Intel(R) Core(TM) i7 with 8 GB of RAM, a 64-bit operating system, and an x64-based processor, and the Windows PC worked in Windows 11 with an 11th Generation Intel(R) Core(TM) i7 processor.

These are the operating environments we are using to perform our research on computational parameters. In terms of browser support, we are using Google Chrome, and for research, we go through Google Scholar, Medium, and data sciences websites to understand different terms according to our usage.

4.3 Baseline Experiment

After obtaining the baseline summaries, we transform them into a model-readable format in order to generate results. For the experiment, we are using the vectorizer function, which is TF-IDF eq, to convert text form into numeric form because machines work in it and our dataset is in text format. After vectorizing, convert sparse matrices to dense arrays because sparse matrices have most of the elements zero and dense matrices do not have non-zero elements.

For this conversion, we use the todense () method: This method is available in most libraries that support sparse matrices, such as NumPy, and it is used to convert a sparse matrix to a dense array. Also, for this, we need to do iterations of the matrix over and over to fill the array with zero-to-nonzero elements for the desired results. After this, we calculate the similarities between the two summaries.

4.4 Experiment

4.4.1 Packages and Libraries

We had discussed in the last model selection the approaches we were going to use for them. There are sacrifice design platforms. For it, we have to install the Python environment and their packages to use some of the libraries, and they have their own built-in functions, which helped us in this research. Some of the libraries for machine learning and deep learning algorithms were built in Python using Keras-backed Tensor Flow, Pandas, Gensim NumPy, PyTorch, NLTK, and Sklearn. We also make use of string, re, rouge, transformers, torch, and matplotlib, as well as Google Cloud Drive, for accessing and saving data.

4.4.2 Classes and Functions

We create a class to provide a way to load and preprocess datasets. We are using methods to create and define objects in it. And in this class, converting text form into numeric form by using an encoder and applying both input and target output with variables input ids, attention marks, and labels These three were used to train and generate text.

The input ids key in the dictionary contains a list of integers that signify the tokenized input sequence. The attention mask key contains a mask that indicates which tokens are padding tokens. These padding tokens are used to add padding to input and output sequences of the same length, and the attention mask is used to ignore the padded tokens while adding attention weights. The label key contains the tokenized output sequence, which is a list of integers. The input ids are the text token ids. The attention mask is a binary mask that specifies which tokens should be prioritized. The attention mask instructs the model on which tokens are critical for predicting the following word.

There are three variables, but before using them, check whether the text is an integer or not. After checking, convert the text into a string and set it to one of the input str variables. After that string is set into the contains list of strings, it will be stored in another input list variable as a sequence. It was applying to each row and storing it in the third input tuple variable, which contains a list of rows. Before tokenizing, we had to convert text into a string if it was in integer form.

Table 4.1: Model Hyper-Parameters

Features	**Model**	
Parameters	**T5**	**BERT**
Batch Size	**8**	**8**
Beam Size	**4**	**4**
Learning Rate	**1e-4**	**1e-5**
Number of Epochs	**50, 60**	**50, 60**

The tokenizer uses an encoder plus built-in functions. There are different encoders of input and target output that use max-length arguments with padding and truncation with retuning formats. Max length is assigned according to models, padding the tokens of those who are less than the length with the equivalent of the given length. Also, truncate the tokens that exceed the assigned lengths to have the same length. The same length is necessary to train and test machines. After this, it assigned the return form, which we further used. The tokenizer encoder argument returned output in PyTOrch tensors (return tensors = pt.') before retuning. At the end of the class, we use the flatten() method to tokenize the sequence into a single list that is accessible by the Pythorch Dataset class.

4.4.3 Fetching and Splitting

After class, accessing datasets from Google Drive Then separated into two sets of data: the training set and the testing set. The splitting ratio of the dataset is 80:20, using the sklearn library's built-in function train_test_split in Python on the Google Colab platform with a random state argument to shuffle the data before splitting it. The use of this to ensure that it is essential for reproducibility, debugging problems more effectively, and comparing the performance of different models more accurately results in Also, a random state helps prevent overfitting[2].

[2]Overfitting: It is aproblem that occurs when a model learns the training data too well and is not able to generalize to new data.

4.4.4 Tokenizers

As we know, there are lots of ways to use tokenizers; however, we are using transformers, and they have their own tokenizer that is specifically designed for it. It uses tokenization to tokenize the text, creating attention masks and converting text into token IDs. This is accomplished by splitting the text down into tokens and then assigning a unique id to each token. The features extractor additionally generates a mask that specifies which tokens should be addressed. It is an essential aspect of the machine learning process since it ensures that the model understands the input text and makes correct predictions. Using the pretrained method, which is already trained on a large dataset, pass through the split sets in the create class to generate embeddings.

4.4.5 Model

Using The Data Loader class is a PyTorch class to improve performance, make it more efficient, save time and resources, and provide a convenient way to load data in batches to train and evaluate models. It is used in both training and validation sets. It uses three arguments: the dataset, the batch size, and the shuffle argument. And also have a number of works that reduce the load of data. We set the device between GPU and CPU; if GPU is not available, then use CPU; otherwise, go with GPU for training time to make the processing faster.
We utilized it as a transformer-based mode and trained it on a dataset of text and code. This pre-training assists the model in learning the associations between words and sentences. It then imports the dataset of articles and summaries and divides it into two parts: training and validation.

4.4.6 Hyperparameters

There are three hyperparameters used: learning rate, optimizer, and epochs. These parameters are important to achieve the best outcomes. Leaning rate, epochs, batch size, maximum length, and number of beams are the hyperparameters employed in our studies. Optimizers do not have hyperparameters. During training, we utilize optimize to optimize the model's parameters. Adam, also known as adaptive moment estimation, is the optimization we employed since it is suitable for deep learning models. The Adam optimizer is used in the given code, with a different learning rate. Which is the learning rate that is frequently employed for training deep learning models. Table 4.1

provides an overview of the experimental information and hyperparameters utilized in each experiment.

4.4.7 Training and Validation

In training, we train and evaluate datasets over a specified number of epochs. And each epoch was trained and evaluated on a batch of data. The training loss and validation loss are calculated using the Adam optimizer and selected learning rates at some specific epochs. Because epochs had problems if the size was not mentioned according to the models, here we are using input ids, attention masks, and labels to generate the predicted output of the model. After that calculation, apply back propagate, optimizer, predicted labels, number of correct predicted data, and overall number of correct predictions. And validation loss: the gradients of the model's parameters are not used in this. This is done because the model is in evaluation mode and the gradients are not needed. After that, save and reload the models.

The model is trained using the training set. The model is trained for a predetermined number of epochs, and the loss and accuracy on the validation set are assessed after each epoch. The code then loads the stored model and utilizes it to build summaries for the validation set's articles. The summaries are then compared against the original summaries to get the precise match ratio. Fine tuning occurs throughout the code's training phase. The algorithm is trained on a vast corpus of text before being fine-tuned using a collection of Roman Urdu articles and summaries. This code is used for word embeddings to be generated.

The BERT tokenizer and T5 tokenizer are used to generate the word embedding in this code. The tokenizer divides the text into words before assigning a vector representation to each word. The code initially imports the required libraries, which include the T5 tokenizer and model, as well as the PyTorch deep learning library. Our major focus in this study is on employing pretrained and fine-tuned computers to create summaries since they are trained to utilize them for tasks, and some scenarios tell us which strategy to use.

4.4.8 Generating Summaries

We are using a testing dataset to generate summaries. The function generate summaries () is used to generate summaries in each batch, and then it decodes the dataset into numeric, text, or string form. In it, there are mechanisms for

early stopping and a number of beams.To construct summaries, the program also employs a technique known as beam search.

4.4.9 Transformers Utilized

Bert, Bart, t5, and PEGASUS are the SOTA summarization models on which the strategy we use is built. In recent years, a variety of summarization models that are more successful than BERT have been created. These models, which are usually based on encoder-decoder architectures, can provide more fluid and coherent summaries than BERT. BART was initially released in 2019, and it rapidly became one of the most popular summary models. T5 was released in 2020 and has proven to be particularly successful for summarizing. Prophet Net was released in 2021 and has proven to be more successful than BART and T5.

4.4.10 Fine-Tuning and Pretrained

The approach we propose is based on SOTA summarization models are Bert-Base-Uncased and T5-Small and used fine-tuned method of pretrained models for it. We do not use typical LSTM baseline models; instead, we use transformer-based models, and for baseline, we use Bard manually to generate and compare our models. Our dataset was in normalized form since it had been processed, tokenized, and embedded to translate text data into numerical form because the machine understood the language of numeric. There are also several types of tokenization: phrase embedding, contextual embedding, and word embedding. Word embeddings are used in both types. Bert, on the other hand, utilizes the BERT tokenizer, while the other uses the T5 tokenizer because it was intended for these models. Instead of both cased and uncased having the same parameters as indicated in Chapter 3 Section Model, the BERT basic uncased design has 110M parameters, and the T5 architecture consists of 220M parameters.

In this section, we have completed another objective of our challenging task, which is to design a model that is effectively capable of generating a summary for abstractive text summarization in Roman Urdu. We explain the experiments in detail with a baseline and which attributes and evaluations were used with their explanations. In the results section, we show the outcomes along with the comparison.

4.5 Results

In this section, we present the results of each model and its comparative effectiveness. The dataset we're utilizing comes in two flavors: one that merely conducts NLP and the other that aims to increase corpus size [5]. The other concentrated on a few high-level languages as well as a few low-level languages. They scraped the crawling pages' HTML structures. They are also focusing on human evolution by recruiting students who are native or multilingual in the languages allocated to them. They'd also provided a slew of automated measures for quantifying crucial aspects of abstractive summaries, such as unique words, abstractivity, compression, and redundancy [21].

4.5.1 Baseline Outcomes

Due to a lack of baseline findings from prior decays, we create our own. As a result, we decided to look for baseline summaries using Google Bard. In the table 4.2, we calculate accuracy, precision, recall, and F1 score to establish a baseline for comparing the outcomes of the supplied and developed models. We employed TF-IDF with cosine similarity for baseline fallouts, and the results were as follows: accuracy and precision were 62.9%, recall was 0.21%, and the F1 score was 49.8%. A comparison of the TFIDF model's performance in terms of accuracy, precision, and F1 score to the model's performance in the dataset's supplied summaries.

Nonetheless, observations reveal that the model's performance utilizing cosine similarity metrics with the TF-IDF vectorizer demonstrates that the baseline has been performing well in terms of the required data results. It demonstrates the resemblance between articles and supplied summaries in terms of baseline discoveries in order to get information about the model's efficiency and how well they performed. Baselines are used to compare abstractive text summarization methods to predefined models.

Table 4.2: Model Performance

Similarity	**Model Performance**			
TF-IDF	**Accuracy**	**Precision**	**F-1**	**Recall**
Baseline	**62.9%**	**62.9%**	**49.8%**	**0.21%**
T5	**56%**	**56%**	**41.5%**	**0.18%**
BERT	**85%**	**85%**	**78.8%**	**0.29%**

Outcomes Overviews

Accuracy, recall, F1 score, and precision are all measures used to evaluate an abstractive text summarization model's performance.

Accuracy

This is the proportion of correct summaries. A proper summary summarizes the major elements of the content. This indicator indicates how frequently the model delivers an accurate summary. A high accuracy score shows that the model was successful in capturing the major ideas of the content. The following demonstrates how we compute the results, and the baseline results are 62.9%, which is sufficient to reflect the core notion of the data rather than 43% of the cited accuracy.

Precision

Precision is the percentage of words in the summary that also appear in the article. This metric counts how often the terms in the summary appear in the article as well. A high accuracy score implies that the model is not producing overly long or verbose summaries. It is the same for both baseline and referral evaluations in terms of results and accuracy.

Recall

It is the proportion of the article's major points that are included in the summary. This metric counts how many of the article's key points are mentioned in the summary. A high recall score suggests that the model can provide a detailed summary. Referred recollection is 0.14%, whereas baseline recall is 0.21%. In this situation, both the baseline and the reference are small, indicating that it is difficult to be properly ordered and understandable.

F1-Score

F1-score is a metric for both accuracy and recall. The harmonic mean of accuracy and recall is used to compute it. This statistic combines precision and recall into a single metric. A high F1 score suggests that the model can create accurate and comprehensive summaries. The figures for baseline are 49.8% and referred are 28.5%, demonstrating that produced summaries are not as accurate and complete as referred summaries. It is crucial to highlight that these measurements are not ideal. They can be influenced by the length of the article, the complexity of the content, and the quality of the training data. They are, nonetheless, still helpful tools for assessing the performance of an abstractive text summarization model.

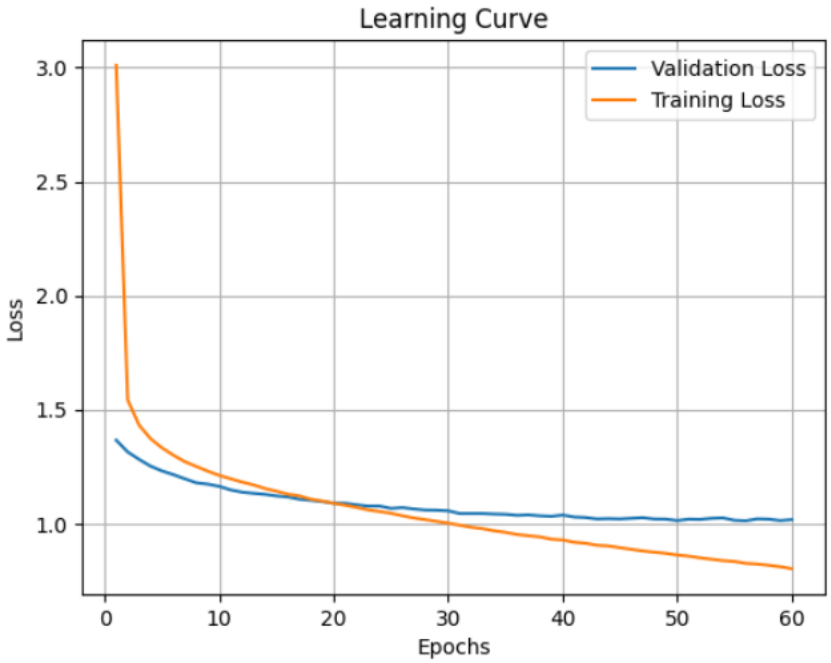


Figure 4.1: T5 Learning Curve

4.5.2 Model Outcomes

Text-to-Text Transfer Transformer (T5)

Model T5-small estimates training and validation losses using the training and validation datasets. The graph depicts both losses and computes the total loss by averaging the losses over the epochs. And each epoch travels through the whole cumulative loss across all of the validation and training phases, which are likewise depicted in Fig 4.1. with their respective learning curves. The total validation loss and overall training loss indicate that the validation and training losses are relatively close, which is a positive indicator. In this example, however, the validation loss is just slightly higher than the training loss, indicating that the model is not overfitting to the training data and is expected to transfer well to new data. It also shows that the model hasn't fully converged. We attempt to train the model for more epochs to see if we can lower the validation loss even more. It is an issue that must be addressed, so we raised the number of epochs for this condition. Additionally, there are various approaches to getting around the issue and enhancing the model, such as regularization methods like dropout or L2 regularization. Increase the set

size, switch to a different optimizer, such as AdamW, and change the schedule for the learning rate.

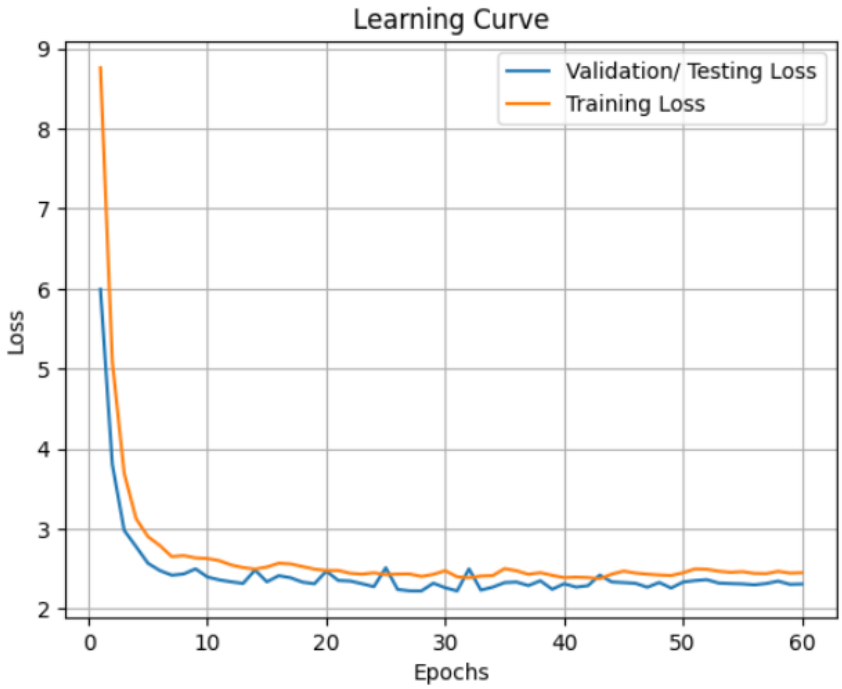


Figure 4.2: BERT Learning Curve

Bidirectional Encoder Representations from Transformers (BERT)

The procedure is the same for this, as seen in Fig 4.2.. Additionally, they are dividing the total loss by the number of epochs to make an evaluation of the entire training and testing loss. And the image with the learning curves illustrates their loss. The model is not overfitting the training set of data since the total validation loss is less than the entire training loss. The validation loss in this instance, though, is only somewhat less than the training loss. Additionally, the model is expected to generalize effectively to new data since it is not overfitting the training set of data.
It exhibits underfitting, which indicates that the model is too simple to adequately capture the subtleties of the training data. Try increasing the number of levels or the number of nodes per layer to resolve this. We also employ regularization strategies like dropout.

4.5.3 Intrinsic Evaluation Metrics

This comparison solely evaluates the text's meaning; it does not compare the similarities between different words, phrases, or sentences. It is one method of quantifying the degree to which the two texts share a common meaning. In terms of clarity, the semantic measurements include the level of abstraction, repetition, coherence, content coverage, human judgment, and others. The similarity between articles with generated summaries and articles with provided summaries is shown in this table 4.3 so that you may compare them and determine which is in a good state in terms of these dependencies. The basic approach provides the best content coverage and fluency ratings, but the redundancy and abstraction level values are the lowest. This shows that while summaries created using the baseline technique are thorough and simple to read, they might not be as illuminating or abstract as summaries created using other methods.
The Bert technique has the highest redundancy and abstraction level scores but the lowest content coverage score. This shows that while the summaries generated by the Bert technique may not be as thorough as those generated by other methods, they may be more useful and abstract. The presented approach has the second-lowest redundancy rating and the second-highest content coverage rating. It demonstrates that although they could be a little less redundant, they offer summaries that are comparable to those generated by the baseline technique. The second-highest fluency and abstraction level scores are achieved by the T5 approach. This shows that while the Bert technique and the T5 method both create summaries, the T5 method may be a little bit simpler to read. The supplied approach, the baseline method, and the T5 method all yield the best overall results. Overall, the Bert technique produces the worst outcomes.

Table 4.3: Model Intrinsic Metrics Performance

Summaries	**Content**	**Fluency**	**Redundancy**	**Abstraction**
Baseline	**6.34%**	**87.62%**	**33.09%**	**93.66%**
BERT	**23.46%**	**71.61%**	**43.25%**	**76.55%**
T5	**8.24%**	**87.31%**	**31.64%**	**91.76%**

4.5.4 Extrinsic Evaluation Metrics

Metrics for evaluating a system's performance are numerical measurements or scoring systems that are used to rate and quantify how well a system summarizes data in response to particular criteria. Metrics offer a means of numerically evaluating how effectively the system satisfies the established assessment criteria. To gauge various facets of the quality of summarization, many metrics are developed. ROUGE, BLEU, METEOR, CIDER, the Flesch-Kincaid readability score, and other commonly used assessment metrics for text summarization are also included here. You may compare and order the various system-generated summaries using these criteria, which offer objective scores. These metrics offer impartial rankings that enable you to contrast and order various system-generated summaries. In summary, evaluation parameters provide the overall goals and criteria for summarization quality, whereas evaluation metrics give particular numerical measures that let you quantify how effectively the system achieves those standards. Both parameters and metrics are critical components of a thorough text summary assessment procedure.

The quality of XL-Sum has been explored in earlier sections. Furthermore, it is critical to examine how state-of-the-art models perform when trained on this dataset. To the best of our knowledge, there is no publicly available dataset or benchmark for abstractive text summarization. In this part, we train summarization models with the XL-Sum dataset and present different baselines and benchmark results.

Fine-tuning On numerous abstractive text summarization datasets, transformer-based seq2seq models started with pretrained weights have been demonstrated to reach state-of-the-art performance. The Hugging Face Transformers provide a plethora of multilingual, pretrained criteria. We picked the T5 and Bert models from among them. We conducted summarizing trials in many situations, including (I) T5, (ii) Bert, and (iii) baseline, with both articles and a provided summary to see how comparable and well summarized they were. For automated evaluation, we employed the ROUGE-1, ROUGE-2, and ROUGE-L. The T5 model offers the best summary in the table 4.4, with ROUGE scores of 0.52, 0.12, and 0.27 for ROUGE-1, ROUGE-2, and ROUGE-L, respectively. With ROUGE scores of 0.47, 0.8, and 0.25 for ROUGE-1, ROUGE-2, and ROUGE-L, respectively, the Bert model delivers summaries that are slightly less close to the reference summaries. And, as indicated in the table, baseline yields summaries that are the least comparable to the reference summaries, with ROUGE scores of 0.47, 0.07, and 0.24 for ROUGE-1, ROUGE-2, and ROUGE-L, respectively.

Table 4.4: Model Extrinsic Metrics Performance

Model	**Rouge-1**	**Rouge-2**	**Rouge-L**
Baseline	**0.47%**	**0.07%**	**0.24%**
T5	**0.52%**	**0.12%**	**0.27%**
BERT	**0.47%**	**0.8%**	**0.25%**

The ROUGE scores for all three metrics in the table are highest for the T5 model, followed by the BERT model and the baseline. The T5 model produces the best summary, followed by the BERT model and the baseline. The baseline model has somewhat higher ROUGE scores than the original example, indicating that it produces more similar summaries to the reference summaries. The baseline model, however, has lower ROUGE ratings than the T5 and BERT models. Overall, the graphic shows that the T5 model provides the best summary, followed by the BERT model and baseline. Overall, analyses reveal that the T5 model, followed by the BERT model, generates the best summary. The baseline model and the model summaries referenced are less comparable to the original text.

The equations are solely for the reader and researchers to see what is going on at the backend and what their functionality is. The prove of generated summaries by the models are also shown in the appendix for clarification.

4.6 Comparative Analysis

The stated summaries We compare the anticipated output with the ground truth labels to determine the model's performance using several criteria. The higher the cosine similarity, the closer the predicted models are to the ground truth labels. This is a circumstance since we are working on an abstract text summary, and the similarity may be at its lowest. This does not imply that the model is ineffective. In this situation, the difficulty is to produce his own writing, which may or may not be identical but is succinct and addresses the critical point of fluency. The section reflects the fact that anticipated tasks need semantic similarity. However, it's important to note that text summarization is a challenging task, and these metrics don't provide the complete picture. The choice of evaluation metrics may vary based on the specific goals and requirements of the summarization task.

4.6.1 Similarity

We utilize cosine similarity to quantify the model's performance in terms of accuracy, precision, recall, and F1-score. However, cosine similarity does not account for the model's prediction accuracy, precision, or recall. As a result, it is critical to combine these measurements with cosine similarity to provide a more full view of the model's performance. In other circumstances, accuracy may be low because anticipated positive labels are inaccurate. In summary, BERT seems to outperform with higher accuracy, precision, and F-1 score.

4.6.2 Model training and testing losses

According to the T5 model, the overall losses are closer together. There is also no evidence of overfitting. The validation loss, however, is slightly more than the training loss. Furthermore, the Bert model validation loss is smaller than the training loss, and they did not overfit the training data. T5 models outperform the Bert model in terms of performance, indicating that they are a preferable alternative for this job as long as they are not overfitting. Furthermore, numerous hyperparameters and approaches may be used to increase the performance of models.

4.6.3 Intrinsic Evaluation considerations

According to the overall evaluation, the baseline summary gets the lowest content coverage score, suggesting that it does not adequately convey the essential elements of the original text. It also has the highest score for redundancy, which

implies that it is repetitive. The Bert summary gets a strong content coverage score but a low fluency and redundancy score. This means the summary is helpful, but it is difficult to read and comprehend. The reference summary has a high fluency score and a good content coverage score. It does, however, have a low abstraction level score, showing that it is written similarly to the original text. The T5 summary has the highest abstraction level score, demonstrating that it is written more broadly than the others. It also has a good fluency score and a strong content coverage score. This shows that among the three summaries, the T5 summary is the most informative, simple to understand, and abstract. As a result, the T5 summary is the best. It effectively conveys the key elements of the original text, is simple to read and understand, and is presented in a more broad tone.

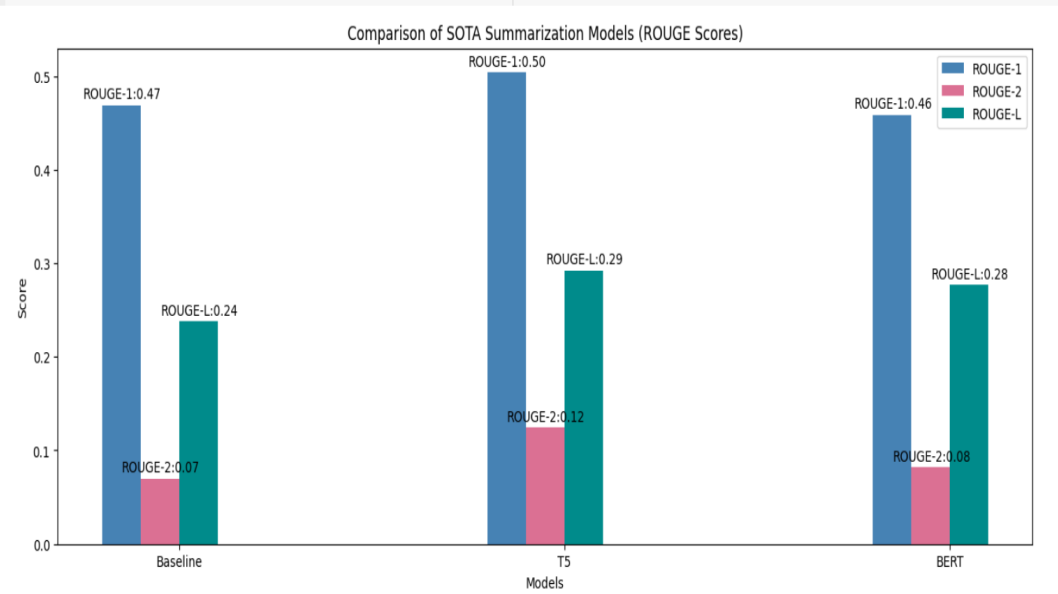


Figure 4.3: Comparison of Models

4.6.4 Extrinsic Evaluation

The findings indicate that the t5 model produces superior summaries compared to the Bert model. The baseline model and the referenced model summaries are less identical than the original text. Rouge is depicted in graph form.

Although they have greater extractive power, the baseline model and the referenced model summaries are less identical to the source text. This means that they merely copy sentences from the original text and make minor changes. The T5 and BERT models are more abstract. This implies that they rework

the original content in their own terms, resulting in more succinct and useful summaries. This implies that T5 can extract the most relevant information from the original text and rewrite it in a succinct and instructive manner. Overall, the T5 model is the most effective summary technique for this assignment. It generates summaries that are both succinct and instructive, and it may consider the context of the original text. The BERT model also generates decent summaries, although not as well as the T5 model. This is due to the BERT model's inability to consider the context of the original text as well as the T5 model. In addition to intrinsic and extrinsic evaluation metrics, there are also hybrid evaluation metrics that combine both intrinsic and extrinsic metrics. These metrics are typically used to get a more comprehensive assessment of the quality of the summary.

We determine baseline results and evaluate the effectiveness of current state-of-the-art summarization methods on our data. Additionally, as stated in the performance evaluation section, examine the models from several perspectives. And evaluate them in light of the State-of-the-Art (SOTA) models. The models we employed show that we approached our challenge using a pre-trained, fine-tuned strategy. These are the goals of our challenges.

CHAPTER 5

CONCLUSION & FUTURE WORK

5.1 Conclusion

In this study, we are given a dataset of Roman Urdu for abstractive text summarization from the BBC. We built our own baseline because benchmark results were unavailable. We employed two strategies.

First, a manual technique of dataset transliteration is used, followed by the generation of baseline summaries. T5 and BERT, transformer-based summarization models, are used in the second approach. We use Natural Language Processing (NLP) phases to create something brief, instructive, and intelligible for this model.

In this research, we are converting an Urdu dataset into Roman Urdu and looking at articles. For conversion, we use a tool. We built our own baseline to generate summaries due to a lack of resources. We assess the outcomes once we generate them. Following that, we used data wrangling to transform our produced dataset from its raw form into a workable one. There are many phases to it based on the language used. Three separate metrics are used for models that use State-of-the-Art (SOTA) summarization models for ATS and assessment.

The first is similarity, in which we compute baseline , T5, and Bert findings; the second is intrinsic metrics; and the third is extrinsic assessment, in which we identify created summaries in a variety of ways.

5.2 Future Work

Due to limited resources and unavailability, this platform will require a significant amount of improvement in the future. Our purpose is to expand the Roman-Urdu dataset. And looking into State-of-the-Art (SOTA) techniques as we progress through the data processing stages to increase the quality of our dataset. Other embeddings, traditional approaches, and pertained models will be employed in the future to provide concise, instructive, and understandable summaries for the models.

REFERENCES

[1] Z. Ali, J. Ali, Zhang lining, zeeshan, niaz muhammad. automatic text summarization for urdu roman language by using fuzzy logic, Journal of Autonomous Intelligence 3 (2) (2020) 23–30.

[2] M. Irfan, Text summarization for urdu: Part 1, website (March 29, 2021). URL `https://www.urdunlp.com/`

[3] U. Khalid, M. O. Beg, M. U. Arshad, Rubert: A bilingual roman urdu bert using cross lingual transfer learning, arXiv preprint arXiv:2102.11278 (2021).

[4] M. P. Akhter, Z. Jiangbin, I. R. Naqvi, M. Abdelmajeed, M. T. Sadiq, Automatic detection of offensive language for urdu and roman urdu, IEEE Access 8 (2020) 91213–91226.

[5] M. Humayoun, R. M. A. Nawab, M. Uzair, S. Aslam, O. Farzand, Urdu summary corpus, in: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), European Language Resources Association (ELRA), Portorož, Slovenia, 2016, pp. 796–800. URL `https://aclanthology.org/L16-1128`

[6] M. U. Arshad, M. F. Bashir, A. Majeed, W. Shahzad, M. O. Beg, Corpus for emotion detection on roman urdu, in: 2019 22nd International Multitopic Conference (INMIC), IEEE, 2019, pp. 1–6.

[7] B. Hu, Q. Chen, F. Zhu, Lcsts: A large scale chinese short text summarization dataset, arXiv preprint arXiv:1506.05865 (2015).

[8] P. J. Liu, M. Saleh, E. Pot, B. Goodrich, R. Sepassi, L. Kaiser, N. Shazeer, Generating wikipedia by summarizing long sequences, arXiv preprint arXiv:1801.10198 (2018).

[9] M. Grusky, M. Naaman, Y. Artzi, Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies, arXiv preprint arXiv:1804.11283 (2018).

[10] W. Kryscinski, N. F. Rajani, D. Agarwal, C. Xiong, D. R. Radev, Booksum: A collection of datasets for long-form narrative summarization, https://arxiv.org/abs/2105.08209 (2021).

[11] R. Nallapati, B. Zhou, C. Gulcehre, B. Xiang, et al., Abstractive text summarization using sequence-to-sequence rnns and beyond, arXiv preprint arXiv:1602.06023 (2016).

[12] S. Narayan, S. B. Cohen, M. Lapata, Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization, arXiv preprint arXiv:1808.08745 (2018).

[13] M. Koupaee, W. Y. Wang, Wikihow: A large scale text summarization dataset, arXiv preprint arXiv:1810.09305 (2018).

[14] A. Cohan, F. Dernoncourt, D. S. Kim, T. Bui, S. Kim, W. Chang, N. Goharian, A discourse-aware attention model for abstractive summarization of long documents, arXiv preprint arXiv:1804.05685 (2018).

[15] B. Kim, H. Kim, G. Kim, Abstractive summarization of reddit posts with multi-level memory networks, arXiv preprint arXiv:1811.00783 (2018).

[16] M. Farahani, M. Gharachorloo, M. Manthouri, Leveraging parsbert and pretrained mt5 for persian abstractive text summarization, in: 2021 26th International Computer Conference, Computer Society of Iran (CSICC), IEEE, 2021, pp. 1–6.

[17] T. Scialom, P.-A. Dray, S. Lamprier, B. Piwowarski, J. Staiano, Mlsum: The multilingual summarization corpus, arXiv preprint arXiv:2004.14900 (2020).

[18] K. Nguyen, H. Daumé III, Global voices: Crossing borders in automatic news summarization, arXiv preprint arXiv:1910.00421 (2019).

[19] F. Ladhak, E. Durmus, C. Cardie, K. McKeown, Wikilingua: A new benchmark dataset for cross-lingual abstractive summarization, arXiv preprint arXiv:2010.03093 (2020).

[20] H. Chouikhi, M. Alsuhaibani, Deep transformer language models for arabic text summarization: A comparison study, Applied Sciences 12 (23) (2022) 11944.

[21] T. Hasan, A. Bhattacharjee, M. S. Islam, K. Samin, Y.-F. Li, Y.-B. Kang, M. S. Rahman, R. Shahriyar, Xl-sum: Large-scale multilingual abstractive summarization for 44 languages, arXiv preprint arXiv:2106.13822 (2021).

[22] Z. Sharf, S. U. Rahman, Lexical normalization of roman urdu text, International Journal of Computer Science and Network Security 17 (12) (2017) 213–221.

[23] Z. Sharf, Tagged for sentiment (positive, negative, neutral), sentiment (2018).

[24] H. Rizwan, M. H. Shakeel, A. Karim, Hate-speech and offensive language detection in roman urdu, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2020, pp. 2512–2522.

[25] M. Y. Khan, M. S. Nizami, Urdu sentiment corpus (v1. 0): Linguistic exploration and visualization of labeled dataset for urdu sentiment analysis, in: 2020 International Conference on Information Science and Communication Technology (ICISCT), IEEE, 2020, pp. 1–15.

[26] M. Amjad, G. Sidorov, A. Zhila, H. Gómez-Adorno, I. Voronkov, A. Gelbukh, "bend the truth": Benchmark dataset for fake news detection in urdu language and its evaluation, Journal of Intelligent & Fuzzy Systems 39 (2) (2020) 2457–2469.

[27] S. Kazi, S. Khoja, Uquad1. 0: Development of an urdu question answering training data for machine reading comprehension, arXiv preprint arXiv:2111.01543 (2021).

[28] I. Safder, Z. Mahmood, R. Sarwar, S.-U. Hassan, F. Zaman, R. M. A. Nawab, F. Bukhari, R. A. Abbasi, S. Alelyani, N. R. Aljohani, et al., Sentiment analysis for urdu online reviews using deep learning models, Expert Systems (2021) e12751.

[29] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, V. Stoyanov, Unsupervised cross-lingual representation learning at scale, arXiv preprint arXiv:1911.02116 (2019).

[30] Q. Liu, E. M. Ponti, D. McCarthy, I. Vulić, A. Korhonen, Am2ico: Evaluating word meaning in context across low-resource languages with adversarial examples, arXiv preprint arXiv:2104.08639 (2021).

[31] B. A. Chandio, A. S. Imran, M. Bakhtyar, S. M. Daudpota, J. Baber, Attention-based ru-bilstm sentiment analysis model for roman urdu, Applied Sciences 12 (7) (2022) 3641.

APPENDIX A

Abstractive Text Summarization Summaries

Example of abstractive summary generated by our models and baseline with the source text and reference summary.

Table 5.1: Abstractive Summary of our Baseline, Models with Reference Summary

Article:
Amrici ehalkaron ka kehna hai ke is wagea mein koi zakhmi nahi sun hwa aur nah helly copter ko nugsaan pounchanay ki koi ittila hai . ghair musadeqa it tila aat ke mutabiq Aini shahdon ne kaha tha ke is helly copter ko rocket ka nishana banaya gaya tha jis ke baad yeh gir kar tabah hogaya tha . guzashta mah faloj_h ke qareeb hi aik shnok helly copter ko maar giraya gaya tha jis mein solah Amrici fouji halaak ho gaye .
Reference Summary:
Amrici fouji ehalkaron ne kaha hai ke aik fouji helly copter ko faloj_h ke gareeb hungami tor par utarna para hai . taham unhon ne kaha hai ke sun hwa baz ne helly copter ko gaboo se bahar nahi honay diya .
Baseline Summary:
Afghanistan mein Amrici helly copter ko aik rocket se nishana banaya gaya, lekin koi jani nugsaan nahi sunhwa . ghair mulki media reports ke mutabiq, yeh waqgea Afghanistan ke soobah warduk ke zila tangi mein paish aaya. mugami logon ka kehna hai ke helly copter ko rocket se nishana banaya gaya, jis ke baad woh zameen par gir gaya. taham, Amrici hakkaam ne kaha hai ke helly copter ko koi nugsaan nahi pouncha hai. guzashta mah bhi Afghanistan ke soobah warduk mein aik Amrici helly copter ko taliban ne maar giraya tha, jis mein 16 Amrici fouji halaak ho gaye thay .
T5 Model Generated Summary:
Amrici ehalkaron ka kehna hai ke is helly copter ko nugsaan pounchanay ki koi ittila hai. ghair musadeqa ila aat ke mutabiq Aini shahdon ne kaha tha ke is helly copter ko rocket ka nishana banaya gaya tha jis ke baad yeh gir kar tabah hogaya tha jis mein solah Amrici fouji halaak ho gaye.
BERT Model Generated Summary:
amrici ehalkaron ka kehna hai ke is wagea mein koi zakhmi nahi sun hwa aur nah helly copter ko nugsaan pounchanay ki koi ittila hai. ghair musadeqa it tila aat ke mutabiq aini shahdon ne kaha tha ke is helly copter ko rocket ka nishana banaya gaya tha jis ke baad yeh gir kar tabah hogaya tha. guzashta mah faloj _ h ke qareeb hi aik shnok helly copter ko maar giraya gaya tha jis mein solah amrici fouji halaak ho gaye.

Laraib plag

ORIGINALITY REPORT

17%	13%	9%	6%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Winston Churchill Middle School Student Paper	2%
2	arxiv.org Internet Source	1%
3	lib.buet.ac.bd:8080 Internet Source	1%
4	www.aclweb.org Internet Source	1%
5	Submitted to Higher Education Commission Pakistan Student Paper	1%
6	"Natural Language Processing and Information Systems", Springer Science and Business Media LLC, 2023 Publication	1%
7	Muhammad Bilal, Atif Khan, Salman Jan, Shahrulniza Musa. "Context-Aware Deep Learning Model for Detection of Roman Urdu Hate Speech on Social Media Platform", IEEE Access, 2022	<1%

	Publication	
8	www.researchgate.net Internet Source	<1 %
9	Muhammad Pervez Akhter, Zheng Jiangbin, Irfan Raza Naqvi, Mohammed Abdelmajeed, Muhammad Tariq Sadiq. "Automatic Detection of Offensive Language for Urdu and Roman Urdu", IEEE Access, 2020 Publication	<1 %
10	Submitted to Liverpool John Moores University Student Paper	<1 %
11	Muhammad Mohsin, Shazad Latif, Muhammad Haneef, Usman Tariq et al. "Improved Text Summarization of News Articles Using GA-HC and PSO-HC", Applied Sciences, 2021 Publication	<1 %
12	digitalcollection.utem.edu.my Internet Source	<1 %
13	"Intelligent Information and Database Systems", Springer Science and Business Media LLC, 2022 Publication	<1 %
14	prr.hec.gov.pk Internet Source	<1 %

15	www.arxiv-vanity.com Internet Source	<1 %
16	paperswithcode.com Internet Source	<1 %
17	www.mdpi.com Internet Source	<1 %
18	Muhammad Bilal, Atif Khan, Salman Jan, Shahrulniza Musa, Shaukat Ali. "Roman Urdu Hate Speech Detection Using Transformer-Based Model for Cyber Security Applications", Sensors, 2023 Publication	<1 %
19	d-scribes.philhist.unibas.ch Internet Source	<1 %
20	archives.univ-biskra.dz Internet Source	<1 %
21	Submitted to University of Rijeka Student Paper	<1 %
22	ayaka14732.github.io Internet Source	<1 %
23	wlv.openrepository.com Internet Source	<1 %
24	Submitted to University College London Student Paper	<1 %

25	aclanthology.org Internet Source	<1 %
26	www.simplilearn.com Internet Source	<1 %
27	Xiaoyan Li, W. Bruce Croft. "Improving novelty detection for general topics using sentence level information patterns", Proceedings of the 15th ACM international conference on Information and knowledge management - CIKM '06, 2006 Publication	<1 %
28	rucore.libraries.rutgers.edu Internet Source	<1 %
29	Submitted to Aristotle University of Thessaloniki Student Paper	<1 %
30	Submitted to Ngee Ann Polytechnic Student Paper	<1 %
31	uscupstate.libguides.com Internet Source	<1 %
32	Samreen Ahmed, shakeel khoja. "UQuAD1.0: Development of an Urdu Question Answering Training Data for Machine Reading Comprehension", Institute of Electrical and Electronics Engineers (IEEE), 2021 Publication	<1 %

33	icon2021.nits.ac.in Internet Source	<1 %
34	Jai Prakash Verma, Shir Bhargav, Madhuri Bhavsar, Pronaya Bhattacharya et al. "Graph-Based Extractive Text Summarization Sentence Scoring Scheme for Big Data Applications", Information, 2023 Publication	<1 %
35	www.urdunlp.com Internet Source	<1 %
36	"Natural Language Processing and Chinese Computing", Springer Science and Business Media LLC, 2020 Publication	<1 %
37	Ihsan Ullah Khan, Aurangzeb Khan, Wahab Khan, Mazliham Mohd Su'ud et al. "A Review of Urdu Sentiment Analysis with Multilingual Perspective: A Case of Urdu and Roman Urdu Language", Computers, 2021 Publication	<1 %
38	www.ijtsrd.com Internet Source	<1 %
39	"Machine Learning and Knowledge Discovery in Databases", Springer Science and Business Media LLC, 2020 Publication	<1 %

40	Submitted to CSU, Los Angeles Student Paper	<1 %
41	huggingface.co Internet Source	<1 %
42	Submitted to The University of Texas at Arlington Student Paper	<1 %
43	sebastianraschka.com Internet Source	<1 %
44	"ECAI 2020", IOS Press, 2020 Publication	<1 %
45	"Intelligent Systems and Applications", Springer Science and Business Media LLC, 2022 Publication	<1 %
46	soar.wichita.edu Internet Source	<1 %
47	web.archive.org Internet Source	<1 %
48	www.ijert.org Internet Source	<1 %
49	2021.emnlp.org Internet Source	<1 %
50	Asmaa Elsaid, Ammar Mohammed, Lamiaa Fattouh Ibrahim, Mohammed M. Sakre. "A	<1 %

	Comprehensive Review of Arabic Text Summarization", IEEE Access, 2022 Publication	
51	dspace.bracu.ac.bd Internet Source	<1 %
52	krex.k-state.edu Internet Source	<1 %
53	speedypaper.x10.mx Internet Source	<1 %
54	www.coursehero.com Internet Source	<1 %
55	www.infoq.com Internet Source	<1 %
56	Muhammad Pervez Akhter, Zheng Jiangbin, Irfan Raza Naqvi, Mohammed AbdelMajeed, Tehseen Zia. "Abusive language detection from social media comments using conventional machine learning and deep learning approaches", Multimedia Systems, 2021 Publication	<1 %
57	Arash Ghafouri, Iman Barati, Mohammad Hossein Elahimanesh, Hamid Hasanpour. "A Question Summarization Method based-on Deep Learning in Persian Language", 2023	<1 %

	28th International Computer Conference, Computer Society of Iran (CSICC), 2023 Publication	
58	etheses.whiterose.ac.uk Internet Source	<1 %
59	Submitted to Babes-Bolyai University Student Paper	<1 %
60	Kostadin Mishev, Ana Gjorgjevikj, Irena Vodenska, Lubomir T. Chitkushev, Dimitar Trajanov. "Evaluation of Sentiment Analysis in Finance: From Lexicons to Transformers", IEEE Access, 2020 Publication	<1 %
61	anjali001.github.io Internet Source	<1 %
62	repository.nida.ac.th Internet Source	<1 %
63	www.researchsquare.com Internet Source	<1 %
64	9pdf.net Internet Source	<1 %
65	Submitted to Indian Institute of Technology-Bhubaneswar Student Paper	<1 %
66	Submitted to Texas A&M University, College Station	<1 %

	Student Paper	
67	acikbilim.yok.gov.tr Internet Source	<1 %
68	compmech.unipv.it Internet Source	<1 %
69	Aashu Kumar, Peeta Basa Pati. "Offline HWR Accuracy Enhancement with Image Enhancement and Deep Learning Techniques", Procedia Computer Science, 2023 Publication	<1 %
70	deepai.org Internet Source	<1 %
71	ebin.pub Internet Source	<1 %
72	kb.psu.ac.th:8080 Internet Source	<1 %
73	"Natural Language Processing and Chinese Computing", Springer Science and Business Media LLC, 2018 Publication	<1 %
74	Phathutshedzo Makovhololo, Ferin Taylor, Tiko Iyamu. "Diffusion of Abstractive Summarisation to Improve Ease of Use and Usefulness", 2018 Open Innovations Conference (OI), 2018 Publication	<1 %

75	ajomc.asianpubs.org Internet Source	<1 %
76	artemis.cslab.ece.ntua.gr:8080 Internet Source	<1 %
77	downloads.hindawi.com Internet Source	<1 %
78	espace.library.uq.edu.au Internet Source	<1 %
79	kth.diva-portal.org Internet Source	<1 %
80	pure.uva.nl Internet Source	<1 %
81	"Advances in Computational Intelligence", Springer Science and Business Media LLC, 2019 Publication	<1 %
82	Avaneesh Kumar Yadav, Ranvijay, Rama Shankar Yadav, Ashish Kumar Maurya. "State-of-the-art approach to extractive text summarization: a comprehensive review", Multimedia Tools and Applications, 2023 Publication	<1 %
83	Dhruv Kolhatkar, Devika Verma. "Indic Language Question Answering: A Survey", 2023 Third International Conference on	<1 %