

Domain-Specific Pretrained Models For Natural Language Generation

Thesis submitted in partial fulfillment
of the requirements for the degree of

*Master of Science in **Computer Science and Engineering** by Research*

by

Sahil Manoj Bhatt

2018111002

sahil.bhatt@research.iiit.ac.in



International Institute of Information Technology
Hyderabad - 500 032, INDIA
June 2023

Copyright © Sahil Manoj Bhatt, 2023
All Rights Reserved

International Institute of Information Technology
Hyderabad, India

CERTIFICATE

It is certified that the work contained in this thesis, titled “Domain-Specific Pretrained Models For Natural Language Generation” by Sahil Manoj Bhatt, has been carried out under my supervision and is not submitted elsewhere for a degree.

Date

Advisor: Prof. Manish Shrivastava

To my family, friends and well-wishers

Acknowledgments

I would like to take this opportunity to express my gratitude to everyone who has supported me throughout my journey (both academic and non-academic) at IIIT Hyderabad.

Firstly, I want to thank my parents, Mr. Manoj Kumar Bhatt and Mrs. Poonam Bhatt, for their unwavering love and support, and for instilling in me the importance of education, hard work and punctuality. Without their encouragement and constant motivation, I would not have made it this far.

I would also like to thank my sister, Ms. Shweta Bhatt, for her constant support and for being my confidante throughout the ups and downs of college life. I am grateful to her for always giving me the best advice, and for her excellent sense of humour.

I am very grateful to my advisor, Prof. Manish Shrivastava, for his guidance, mentorship, and patience. Manish Sir is a wonderful and motivating guide. My every interaction with him was one where I learnt something new, and I am grateful to have had an advisor like him who let me explore various problems in the field so that I can choose and work on something that interests me.

I am very fortunate to have worked with Prof. Manish Gupta, whose valuable insights and constructive feedback have been instrumental in shaping my research journey. I have learnt a lot during my time collaborating with him, whether it be learning how the latest models work, the best way to document research and track progress, or the art of writing papers.

I extend my heartfelt appreciation to all the professors who have taught me and challenged me to excel. Their dedication and passion for their subjects have inspired me to pursue my own interests with greater zeal. Whether it be CS courses, Science or Humanities electives, I am sure that I will carry the knowledge I have gained here for the rest of my life.

I would also like to thank all the staff at IIIT Hyderabad for keeping everything running so smoothly, thus providing me with a campus life better than I could have even imagined.

And as they say, not all wisdom comes from books (or laptops, since that is more relevant in my case). My friends, batchmates, seniors and juniors have all been instrumental throughout this journey. I would like to thank my friends for always being there for me, and for teaching me things that books cannot. Their support and humour have made my college experience more enjoyable. Long walks around campus and along ISB road, cycling expeditions, going to the canteens and DLF food court, music and gaming sessions, E-Cell and club meetings are some of my fondest memories.

To all these wonderful people, I owe my sincerest gratitude.

Thank you IIIT Hyderabad. Thank you, everyone!

Abstract

Natural Language Generation (NLG) focuses on the automatic generation of natural language text, which should ideally be coherent, fluent, and stylistically appropriate for a given communicative goal and target audience. The tasks in NLG are varied, whether it be summarization, headline generation, dialogue generation etc., and are also heavily dependent on the domain being considered.

Recent research has focused on creating domain-specific datasets and developing domain-specific models to make NLP systems more suited to real-world applications. Training models on data specific to a domain has been observed to yield significantly better results across different domains, whether it be legal, financial or biomedical.

However, we observe that there has not been much work done on problems in the tourism domain. The tourism industry is important for the benefits it brings and due to its role as a commercial activity that creates demand and growth for many more industries. Currently, there does not exist any standard benchmark for the evaluation of travel and tourism-specific data science tasks and models.

To address this gap, we propose a benchmark, TOURISMNLG, of five natural language generation (NLG) tasks for the tourism domain and release corresponding datasets with standard train, validation and test splits. Moreover, as NLG systems are diversifying across languages, the datasets we create and the models we contribute are also multilingual in nature, which is beneficial for the tourism industry globally.

Further, previously proposed data science solutions for tourism problems do not leverage the recent benefits of transfer learning. Thus, in this thesis, we also contribute the first rigorously pretrained mT5 and mBART model checkpoints for the tourism domain. The models have been pretrained on four tourism-specific datasets covering different aspects of tourism.

Using these models, we present initial baseline results on the benchmark tasks, that indicate an improvement in performance as compared to the respective models without domain-specific pretraining.

Additionally, we consider the problem of summarization for Indian languages, as described in the IL-SUM (Indian Language SUMmarization) shared task, which focuses on summarising content from the news domain in three important Indian languages: Indian English, Hindi, and Gujarati. We evaluate the performance of existing pretrained models for the task and present our results and findings. We also talk about steps that must be taken to create high-quality summarization datasets for Indian languages.

We hope that the contributions of this thesis will promote active research for natural language generation for travel and tourism, as well as other domain-specific and language-specific tasks and models.

Contents

Chapter	Page
1 Introduction	1
1.1 Motivation	1
1.2 Natural Language Generation	3
1.3 Contributions	3
1.4 Thesis Organization	4
2 An Overview of Problems, Models and Tasks	5
2.1 Pretrained Language Models	5
2.2 Domain-based perspective	6
2.3 Language-based perspective	6
2.4 Task-based perspective	7
2.4.1 Text Summarization	7
2.4.1.1 Extractive Summarization	8
2.4.1.2 Abstractive Summarization	8
2.4.1.3 Variation in Summarization tasks	9
2.4.2 Headline Generation	9
2.4.3 Question-Answering	10
3 Domain-specific NLG: Exploring the tourism domain	11
3.1 Introduction	11
3.1.1 How is travel text different?	12
3.1.2 Our contributions	13
3.2 Related Work	13
3.2.1 Tourism-focused Data Science	13
3.2.2 Domain-specific Pretrained Models	14
3.3 TOURISMNLG Benchmark	14
3.3.1 TOURISMNLG Datasets	14
3.3.2 TOURISMNLG Tasks	19
3.4 Baseline Models for TOURISMNLG	23
3.4.1 Model Selection	23
3.4.2 Pre-Training and Finetuning	24
3.4.3 Metrics	24
3.4.4 Implementation Details for Reproducibility	25
3.5 Experiments and Results	25
3.6 Conclusions	31

4	Indian Language Summarization	32
4.1	Introduction	32
4.2	Related Work	33
4.3	Corpus Description	33
4.4	Experiments and Results	33
4.5	Data Quality Assessment	35
4.5.1	Data Variation Experiments	36
4.6	Conclusions	36
5	Conclusions and Future Work	40
5.1	Conclusions	40
5.2	Future Work	41
	Bibliography	44

List of Figures

Figure	Page
3.1 An example of a forum discussion on TripAdvisor. Image-source: https://www.tripadvisor.in/ShowTopic-g293974-i368-k14185302	16
3.2 An example of a Wikipedia page: first paragraph of the article (right) and the corresponding infobox (left). Image-source: https://en.wikipedia.org/wiki/Larnaca	17
3.3 Word cloud for top few words in the text associated with TripAdvisorQnA (top-left), TravelWeb (top-right), TravelBlog (bottom-left) and TravelWiki (bottom-right) datasets.	18
3.4 Variation in Pretraining Loss on Training as well as Validation Data for our mT5 and mBART models under the MLM and MLM+Tasks settings.	28

List of Tables

Table	Page
3.1 Language Distribution across the four datasets.	19
3.2 Characteristics of the datasets in TOURISMNLG.	20
3.3 Characteristics of the tasks in TOURISMNLG. I = Avg Input Sequence Lengths (in words). O = Avg Output Sequence Lengths (in words).	22
3.4 Models used in TOURISMNLG	23
3.5 Results on TOURISMNLG Tasks: Long QA, Forum-title Generation, Paragraph Generation, Short QA and Blog-title Generation. For finetuning, STF=Single Task Finetune and MTF=Multi-task Finetune. For pretraining, (-) means no pretraining, (A) means MLM, (B) means MLM+Tasks. The best results in each block are highlighted.	26
3.6 Examples of predictions using our best model	29
3.7 Human Evaluation Results	30
3.8 Error Analysis: # errors across categories for each task (out of 50 judged samples). . .	31
4.1 ILSUM Dataset Statistics	34
4.2 Results for Validation Data. * indicates that the model was finetuned on the combination of Hindi and Gujarati Data	37
4.3 Results on Test Data	38
4.4 Experimental setup and parameter settings	38
4.5 Statistics of dataset examples that we consider valid after applying TeSum [94] filters .	38
4.6 Mean ROUGE scores over the validation sets along with standard deviation over 10 runs. O.D indicates Original Data, F.D indicates Filtered Data	39

Chapter 1

Introduction

In this chapter, we introduce the reader to the topic of this thesis: natural language generation tasks and domain-specific pretrained models. We present an overview of the problem space and talk about the motivation behind the problems worked on in this thesis. We provide an overview of natural language generation and talk about its applications. In particular, we look at the importance of research in natural language generation for the tourism domain. We also touch upon the problem of summarization for Indian languages. We then highlight the key contributions of this thesis and present its overall organization.

1.1 Motivation

With the rapid growth and adoption of technology worldwide, a vast amount of information is produced daily across different domains. Moreover, the data generated every day does not originate solely from companies, brands and businesses. A significant portion of the data created nowadays is user-generated content (UGC), where users contribute content, whether it be text, images, videos, etc. Such data can be commonly found across social media, discussion forums, Wikipedia pages, review/suggestion portals, etc. It is imperative to analyze, interpret and make use of such data since it is a very valuable source of information and can help in spurring the development of more efficient, automated and knowledgeable systems.

Natural Language Generation (NLG) systems are especially witnessing a wide adoption across a wide and diverse range of business sectors, due to their ability to meet the changing needs of users in an economical manner. These systems are used in wide variety of business applications, whether it be summarization, chatbots or question-answering, to name a few. Current NLG systems are playing a big role in significantly reducing manual work, and are increasingly finding themselves being used as assistive tools, if not an altogether replacement of manual work.

Pretrained language models (PLMs) are an obvious choice for creating new approaches (architectures) for NLG systems due to their success across a number of NLP tasks. To learn the complexities of language, their pretraining strategies make use of vast amounts of data.

However, the data these models have been trained on is typically generic and not specific to any domain. An additional problem is that a large number of the existing models and datasets are English-centric. While having a model that "knows it all" would be the ultimate goal, many of the current NLG models are not there yet and suffer from problems when it comes to niche, domain-specific tasks. Fortunately, developments in transfer learning, where knowledge learned through a few tasks can be used to solve different tasks, have made it possible to improve existing models to help them solve wide-ranging problems.

A lot of the latest research has focused on developing models specific to a particular domain to improve their usability and accuracy. These typically involve identifying and proposing tasks in such domains, creating the necessary, high-quality datasets, and then training the model using this data with the task objectives. Such models have found success in various domains, for instance, models for the legal domain [13], financial domain [5], biomedical domain [46], etc.

However, to the best of our knowledge, there has not been any benchmark or model specific to the tourism domain. Tourism was one of the hardest-hit industries during the COVID-19 pandemic [24] owing to both international and domestic travel restrictions. People's desire to travel again has led to a sudden boom and influx of travellers in tourist spots as businesses begin to return to pre-pandemic normalcy. This brings its own set of challenges, and thus, necessitates the development of automated systems that can assist those involved in the travel domain - businesses and travellers alike.

One of the ways to assist those in the tourism industry through AI (more specifically, NLP) is to first propose a benchmark - with an initial set of tasks and models capable of performing them - that can serve as a starting point for developing systems specific to tourism. More importantly, since travel and tourism is a global industry, it is important to have datasets for non-English languages as well. Given the nature of tasks such as question-answering related to trips, travel advice, providing assistance in writing travel articles, etc., generative tasks were our natural choice. Our choice can also be attributed to the rapid advancement in NLG performance by PLMs. Thus, this thesis discusses the creation of a tourism domain benchmark, titled TOURISMNLG, which comprises several multilingual NLG tasks and datasets.

PLMs have been trained on massive corpora of general domain text and display strong performance across most NLP tasks. However, research work across different domains has shown that PLMs underperform in specialised domains [6, 46]. PLMs can perform much better across domain-specific tasks if they are properly adapted to the respective domains. Pretraining on domain-specific datasets can help the model understand the distinct characteristics of the language used for a particular domain. Domain-specific texts often have a specialised vocabulary and semantics based on knowledge specific to that domain. For instance, the language used in legal texts has often been classified as a sublanguage [13, 92, 99].

We believe that the adaptation of PLMs to relatively unexplored domains such as travel and tourism will bring many benefits. Therefore, as part of TOURISMNLG, we also look at the development of pretrained models specific to the tourism domain.

The second part of this thesis looks at a different problem - Indian language summarization. We provide a comprehensive overview of the challenges faced in generative tasks such as summarization when it comes to Indian languages, which have relatively less available data than European languages such as English, French, Spanish, etc. It is important and relevant to the main focus of this thesis since it mainly deals with how PLMs perform in summarization tasks for the news domain (an NLG task), specifically in Indian languages. Such analysis is important from the perspective of the need to develop domain-specific models for different languages, especially low-resource ones. The results of such a study can also benefit research in the creation of datasets and the development of PLMs for low-resource languages, as well as NLG benchmarks for those languages.

1.2 Natural Language Generation

Natural Language Generation (NLG) is a subfield of NLP that focuses on producing natural language that can be understood by humans. A good NLG system should generate text that is coherent, stylistically fitting and relevant for a particular communication purpose and intended audience. NLG is useful in many fields, such as business intelligence, media, healthcare, and education, to name a few. Summarizing extensive datasets, creating customized reports, answering questions, coming up with descriptions for images, automating content creation, etc. are just some of the many applications of NLG.

Much of the latest research in NLG is centred around improving the diversity and quality of the generated text, in addition to enabling NLG systems to handle complex reasoning and inference. There is also a lot of work nowadays towards creating more interactive and personalized NLG applications.

1.3 Contributions

Overall, in this thesis, we make the following key contributions:

- We propose a benchmark of five novel, diverse and multilingual tourism NLG tasks called TOURISMNLG. As part of this benchmark, we also contribute four datasets along with standard splits to the research community.
- We pretrain multiple tourism-domain specific models. We also make the pretrained models publicly available.
- We experiment with multiple pretraining and finetuning setups, and present initial baseline results on the TOURISMNLG benchmark.
- We evaluate the performance of PLMs for summarization tasks in the news domain for Indian languages.

1.4 Thesis Organization

The thesis is divided into 5 chapters and is organized as follows:

- **Chapter 1** introduces the reader to the definition of Natural Language Generation and its applications. We discuss the motivation behind the problems highlighted and the contributions made in this thesis.
- **Chapter 2** discusses relevant work in the field from different perspectives. We first highlight the performance and capabilities of pretrained language models (PLMs). We then look at related research in the field from three perspectives - (1) domain-based perspective, where we look at work done with respect to the development of datasets, models and techniques for specific domains, (2) language-based perspective, where we look at models and datasets that are multilingual in nature, or specific to non-English languages or group of languages, and (3) a task-based perspective, where we look at different NLG tasks, definitions and methods.
- **Chapter 3** describes the main contribution of this thesis - the TOURISMNLG benchmark, where we contribute four multilingual datasets specific to the tourism domain to the research community - TravelWeb, TravelWiki, TripAdvisorQnA and TravelBlog. We discuss in detail the choice of tasks - blog title generation, forum title generation, short question answering, long question answering, and paragraph generation. We discuss data collection and processing techniques, different approaches used to solve them, and metrics (both automated and manual) used to evaluate our results.
- **Chapter 4** elaborates the performance of PLMs for summarization tasks for Indian languages in the news domain. The languages that we consider for this study are Hindi, Gujarati and Indian English.
- **Chapter 5** summarizes the contributions of this thesis and discusses potential future work that can arise from this thesis.

Chapter 2

An Overview of Problems, Models and Tasks

In this chapter, we first look at existing pretrained language models. Next, we focus on looking at related work from three perspectives:

- Domain-based perspective: We look at research done on pretrained language models for specific domains, and also focus on previous works done for the tourism domain.
- Language-based perspective: We present an overview of multilingual and language-specific models and datasets.
- Task-based perspective: We describe some of the popular and challenging problems in NLG that are relevant to our thesis. We describe the objective of each task and a description of the techniques used to solve it.

2.1 Pretrained Language Models

Pretrained language models (PLMs) have been shown to outperform most existing approaches for a number of natural language generation tasks. PLMs used for downstream tasks are pretrained using massive amounts of unlabeled text data. A PLM encodes extensive linguistic knowledge into a vast amount of parameters[49], which stimulates universal representations and improves generation quality. With the development of Transformer models [97], NLP has seen a significant boost across most task-specific metrics.

The BERT model [21] consists of encoder layers stacked together (12 in BERT_{BASE} and 24 in BERT_{LARGE}). At the time of its release, it had yielded state-of-the-art results across multiple NLP benchmarks such as SQuAD question answering task [76], GLUE [98], etc. As a result of BERT’s success, similar models followed - Domain-specific (see Section 2.2) such as FinBERT [5], PatentBERT [47], LegalBERT [13], ClinicalBERT [36], BioBERT [46], etc., and Language-specific/Multilingual (see Section 2.3) such as mBERT [21], IndicBERT [41], AraBERT [4], etc.

BART [48] is a denoising autoencoder for pretraining seq2seq models, which is similar to both BERT and GPT [73] since it uses a bidirectional encoder like BERT and an autoregressive decoder like GPT.

The model was trained by corrupting the text using a noising function, and reconstructing the original text. There exist multilingual variants[53] of the BART model, such as mBART and mBART-Large-50 (610M parameters) model[91], trained on 50 languages.

T5 [75] model proposes defining every NLP task in a text-to-text format. The model consists of an encoder-decoder Transformer architecture finetuned on the C4 corpus. The mT5 model [101] uses an architecture very similar to T5, and is trained on 101 languages, as described in the mC4 dataset.

The GPT (Generative Pretrained Transformer) suite of models [73, 74, 8] have also recently resulted in highly improved few-shot approaches to NLP tasks, where the model generates outputs based on few input-output examples provided to it.

2.2 Domain-based perspective

A number of domains have proposed their own specific tasks, along with pretrained models to solve them. LegalBERT [13] proposes two settings to pretrain BERT: (1) further pretraining and (2) pre-training from scratch, to solve several Legal NLP tasks such as text classification (multi-label text classification of EU laws [12] and binary/multi-label classification of cases [10]) and sequence tagging (named entity recognition of US contracts [11]). BioBERT [46] looks at various biomedical text mining tasks, such as biomedical NER, biomedical QA and biomedical relation extraction. SciBERT [6] was developed to improve performance on tasks such as text classification, dependency parsing and NER in the scientific domain, particularly focusing on data from the (1) biomedical domain, such as NCBI-disease [22], EBM-NLP [64], etc., and (2) computer science domain, such as ACL-ARC [40], PaperField [88], etc. Alsentzer et al. present two pretrained models based on BERT to solve tasks in the clinical domain such as NLI (MedNLI dataset [78]) and NER (i2b2 [65]). Covid-Twitter BERT [58] was pretrained on a large corpus of tweets related to COVID-19, and was used for classification tasks such as vaccine sentiment, maternal vaccine stance, sentiment analysis, etc. FinBERT [5] was used for financial sentiment analysis tasks, while PatentBERT [47] was used for patent classification.

With respect to the tourism domain, there has been very little work done. Work in this domain has mainly focused on tasks such as structured extraction of trip-related information [71], mining reviews [67], automatic itinerary generation [20, 25, 14], aspect extraction [29], analyzing travel blogs [42, 37] and hotel recommendation [3].

2.3 Language-based perspective

In recent times, a number of models have been proposed to solve multilingual tasks. These models have been pretrained on a vast amount of multilingual data. mBERT [21] was pretrained on monolingual corpora from 104 languages. The mT5 [101] model, based on the T5 model, was pretrained on the mC4 dataset, a multilingual version of the C4 dataset, including examples from 101 languages. Liu et al. proposed the mBART model [53], which is a sequence-to-sequence denoising auto-encoder that uses a

BART pretraining objective on large-scale monolingual corpora in many languages. mDeBERTa [32] is a multilingual version of DeBERTa [33] trained with CC100 multilingual data. Pretraining multilingual language models at scale has also led to significant performance gains for a wide range of cross-lingual transfer tasks, as reported in XLM-R [18]. Similar to LLMs like GPT which use causal language modelling, BLOOM [100] is an open-access multilingual language model containing 176 billion parameters that can generate text in 46 natural languages and 13 programming languages.

There has also been a lot of research in developing models and creating datasets for a specific language or class of languages. Martin et al. [55] propose CamemBERT, a monolingual Transformer-based language model for French. AraBERT [4] is a pretrained BERT model for the Arabic language. The IndicNLP Suite [41] released benchmark datasets and models (IndicBERT) for 11 Indian languages, and the IndicNLG Suite [44] released datasets for several generative tasks for Indian languages. IndicBART [19] is a pretrained sequence-to-sequence model trained on 11 Indic languages and English. It follows the masked span reconstruction objective similar to mBART. In contrast to available generation models, IndicBART utilizes the orthographic similarity between the Indian languages to achieve better cross-lingual transfer learning capabilities. This model size (244M) is much smaller than mBART and mT5 models with compact vocabulary. Khanuja et al. proposed MuRIL [43], a multilingual LM for Indian languages. TeSum [94] consists of a very large Telugu summarization corpus. Similar work has also been done for other languages: Chinese summarization corpus LCSTS [35], DaNewsroom for Danish [95], IndoNLG for Indonesian languages [9], Korean NLU tasks [68], Turkish [81] etc.

Additionally, there have been several multilingual datasets recently focused on specific tasks. Scialom et al. proposed MLSUM [86], a multilingual summarization dataset with over 1.5 million news article-summary pairs across five languages - French, German, Spanish, Russian and Turkish. XGLUE [50] is a benchmark dataset to train large cross-lingual models, and NLG tasks such as Question Generation and News Title Generation are part of this benchmark, with data available in languages such as English, French, Spanish, German, etc. XL-Sum [30] and MassiveSumm [96] release multilingual datasets for 44 and 92 languages respectively.

2.4 Task-based perspective

2.4.1 Text Summarization

Summarization is a task in natural language generation (NLG) that involves creating a concise and comprehensive summary of a longer text. The goal of summarization is to condense the most important information from the text into a smaller and more manageable format, while preserving the meaning and key aspects of the original content. Text summarization is typically classified into three types:

- *Extractive summarization*: This involves selecting and combining the most important sentences from the text

- *Abstractive summarization*: This involves generating new sentences that summarize the content of the text
- *Hybrid approach*: A combination of the above two approaches - extractive and abstractive

Pretrained language models have resulted in remarkable improvements across various language tasks. These models, such as those developed by [74], [103] are pre-trained on massive amounts of text data and can then be fine-tuned for specific tasks. Such models have been found to be very effective for summarization tasks. Some of the most commonly used datasets for English language summarization include CNN/Daily Mail [34], New York Times Corpus [82], Gigaword [61], Newsroom [28] and XSum [62].

2.4.1.1 Extractive Summarization

One of the earliest ideas with respect to choosing sentences based on their importance was TextRank [56], an algorithm that leverages a variant of PageRank [66] to identify important sentences or keywords in a given document using a graph-based approach where the nodes represent sentences and the edges represent the semantic similarity between the nodes. Cheng et al. [17] propose a single-document summarization framework composed of a hierarchical document encoder and an attention-based extractor. Nallapati et al. [60] adopted an encoder based on RNNs to perform extractive summarization. Neural models typically view extractive summarization as a sentence extraction problem. The framework primarily involves an encoder, which encodes sentence representations, and a classifier, that predicts whether a sentence is summary-worthy or not.

2.4.1.2 Abstractive Summarization

Rush et al. [80] combine a neural language model with an attention-based encoder to generate an abstractive summary. See et al. [87] propose a pointer-generator framework, which allows copying words from the source text via pointing, while also generating words from a fixed vocabulary. Paulus et al. [69] proposes a reinforced learning based approach to abstractive summarization involving a hybrid learning objective (a combination of maximum-likelihood estimation and reinforcement learning objectives). Chen et al. [16] proposes a model that first selects salient sentences and then rewrites them in an abstractive manner. CTRLsum [31] presents a framework for controllable summarization that enables users to control multiple aspects of generated summaries by interacting with the summarization system through textual input in the form of a set of keywords or descriptive prompts. PEGASUS [103] uses the extracted gap sentences (GSG) self-supervised objective strategy to train the encoder-decoder model, which involves masking the entire sentence, as compared to masking a smaller text span as seen in BART and T5. The pretraining is performed with C4[75] and HugeNews corpus. BRIO [54] is a novel training paradigm to achieve neural abstractive summarization, wherein a contrastive learning component is introduced to reinforce the abstractive model’s ability to estimate the probability of

system-generated summaries more precisely instead of using MLE training alone. Two stages are involved in this approach: the first stage generates the candidates using a pretrained sequence-to-sequence model, and next stage selects the best one. ProphetNet [72] introduces a novel self-supervised objective, wherein the goal is to predict the next- n tokens, instead of just optimizing for one-step ahead predictions.

2.4.1.3 Variation in Summarization tasks

Summarization is a very broad task and encompasses a number of sub-problems. We can look at summarization problems based on the following aspects:

- *Based on document length*: Single-document, Multi-document.
- *Based on summary language*: Monolingual, Multilingual, Cross-lingual.
- *Based on domain*: Domain-specific, Domain-independent
- *Based on summary type*: Full Summary, Sentence-level summary, Headlines, Highlights.

Summarization remains a challenging task, as it requires a deep understanding of the text, as well as the ability to generate fluent and coherent summaries that accurately reflect the meaning of the text. Additionally, there are often trade-offs between the quality of the summary and its length, as longer summaries may provide more information but may also be less concise.

2.4.2 Headline Generation

Headline generation is a task in natural language generation (NLG) that involves creating a title for a given input text, such as an article or a news report. It is a crucial task in NLG that has important applications in journalism, publishing, and online content creation. The goal of headline generation is to capture the most important information from the text and present it in a concise and attention-grabbing way. Headline generation is closely related to the problem of text summarization, since both involve producing text that captures the essence of an input text. However, headline generation differs in the expected output length, and in many cases, the interestingness aspect of the generated text. The kind of headlines required by an entertainment magazine publisher (attractive, clickbait-like), for instance, would be very different from that of an academic researcher looking for a title for their paper (professional, scientific, factual). As a result, there has been a lot of research in generating headlines/titles based on certain qualities.

Jin et al. [38] propose a Stylistic Headline Generation task to generate headlines attractive to readers in three styles: humor, clickbait and romance. [90] proposed a method that utilizes a coarse-to-fine approach to generate headlines where they first identify the significant sentences within a document using document summarization techniques and then use a multi-sentence summarization model with hierarchical attention to incorporate the important sentences into the headline generation process. Liu

et al. [52] use a Transformer decoder to produce various headlines for news articles containing key phrases that are of interest to the users, where they first generate multiple key phrases that are relevant to the news for the users and then produce several headlines that relate to those key phrases. [44] create a multilingual dataset specific to Headline generation for 11 Indian languages, and use Transformer models such as mT5 [101] and IndicBART [19] to generate headlines.

2.4.3 Question-Answering

Traditional Question Answering (QA) systems typically involved some information retrieval techniques in order to find answers to questions. With the advancement in neural language models, generative question answering systems have seen a significant boost in performance. Generative QA involves generating abstractive answers to questions, rather than selecting from a pre-defined set of answers or extracting answers directly from the input text, thus providing a more natural and flexible way of answering questions. For QA tasks, the main challenge lies in ensuring that the answers, whether extracted or generated, are both informative and accurate.

Rajpurkar et al. [76] created the popular QA benchmark dataset, SQuAD, where a model needs to extract a text span (the answer), given a question and a paragraph as the context. The MS MARCO dataset [63] consists of context passages derived from web pages retrieved by Bing, with Bing user queries as the questions. The answers in the dataset were generated by human annotators. The NewsQA dataset [93] has over 100,000 human-generated question-answer pairs related to the news domain (data from CNN), having text spans as answers. Similarly, PubMedQA [39] is a domain-specific dataset for biomedical question answering, while CoQA [77] is a dataset for conversational question answering systems. Morales et al. [57] present the InfoboxQA dataset with data from Wikipedia article infoboxes, and present a convolutional neural network (CNN) model that yields best results.

Chapter 3

Domain-specific NLG: Exploring the tourism domain

3.1 Introduction

According to the World Travel and Tourism Council, travel and tourism accounted for (1) 10.3 percent of global GDP in 2019, (2) 333 million jobs, or one in every ten jobs worldwide, and (3) US\$1.7 trillion in visitor exports (6.8 percent of total exports, 27.4 percent of global services exports)¹ in 2019. Tourism increases the economy’s revenue, creates thousands of jobs, improves a country’s infrastructure, and fosters a sense of cultural exchange between foreigners and citizens. This commercially important industry has resulted in a large amount of online data.

Data in the tourism domain can be typically seen in the form of:

- public web pages (blogs, forums, wiki pages, general information, reviews)
- travel booking information owned by travel portals which includes customer travel history, schedules, optimized itineraries, pricing, customer-agent conversations, etc.

As a result, research on tourism data mining has mainly concentrated on automated itinerary generation [14, 15, 20, 25], personalised sentiment analysis of visitor reviews [67], and structured extraction of trip-related information [71]. However, the majority of this work has utilised conventional techniques for performing natural language processing (NLP).

Recently, transfer learning techniques using pretrained models have shown immense success across almost all NLP tasks. Transformer [97] based models like Bidirectional Encoder Representations from Transformers (BERT) [21], Generative Pre-trained Transformer (GPT-2) [74], Extra-Long Network (XLNet) [102], Text-to-Text Transfer Transformer (T5) [75] have been major contributors to this success. These models have been pretrained on generic corpora like Books Corpus or Wikipedia pages. To maximize benefits, researchers across various domains have come up with domain-specific pretrained models like BioBERT (biomedical literature corpus) [46], SciBERT (biomedical and computer science literature corpus) [6], ClinicalBERT (clinical notes corpus) [36], FinBERT (financial services

¹<https://wtcc.org/research/economic-impact>

corpus) [5], PatentBERT (patent corpus) [47], LegalBERT (law webpages) [13], etc. There are no models, though, that have been specifically pretrained for the tourism domain. Additionally, there is no established benchmark for tasks related to tourism.

3.1.1 How is travel text different?

As observed in other domains, an investigation into the kind of text used in the travel domain reveals that it is very different from usual text across domains, and having models specifically trained for this domain will surely yield better results for tourism/travel-specific tasks. Skibitska [89] investigated the degree of specialization of the language of tourism in different kinds of tourism-related texts. They group tourism vocabulary into groups like types of tours and tourism (e.g. agro-tourism, incentive tour, rural tourism, week-end tour, day trip etc.), industry professionals (e.g. guide, event organizer, travel agent, tourist information centre assistant, etc.), catering (e.g. full board, white-glove service, buffet, a la carte, coffee shop, tip, bev nap, etc.), accommodation (e.g. standard room, daily average rate, reservation, cancellation, room facilities, spa, check-in, prepaid room etc.), transportation (e.g. charge, refund, non-refundable, actual passenger car hours, excess baggage, scheduled flight, frequent flyer, etc.), excursion (e.g., itinerary, overnight, local venue, sightseeing, city guide, departure point, meeting point, hop on hop off etc.), abbreviations (e.g. IATA, AAA, WTO, NTA, etc.). Compared to usual blogs, travel and tourism blog titles often:

- include a destination or type of travel experience to emphasize the appeal of the location
- emphasize the “adventure” aspect of traveling
- include words like “journey”, “voyage,” or “road trip” to emphasize the journey aspect of traveling
- include the words “explore” or “discover” to emphasize the discovery of new places
- use vivid adjectives or descriptive phrases to emphasize the beauty and uniqueness of the destination

Compared to generic answers, answers on travel forums are:

- more focused on specific destinations
- typically more concise and to-the-point
- written in a more conversational tone
- frequently include personal stories and anecdotes
- include advice or tips and less opinion-based
- often written in the first person

- written in a positive or helpful manner

Paragraphs in travel webpages tend to describe culture and event sequences (in blogs), temporal facts and planning (on forums), etc. For factual short question answering, the answer types are rather restricted in tourism domain to architectural types, geographic names, population, timings, cost, directions etc.

3.1.2 Our contributions

We propose a benchmark, TOURISMNLG, consisting of five novel natural language generation (NLG) tasks in the travel and tourism domain. The number of instances across these five tasks adds up to 4.2M instances. We make the datasets corresponding to these tasks, along with their train, validation and test splits publicly available². We also make the code and all our pretrained models publicly available.

Given this benchmark of five tourism NLG tasks and four different tourism-specific multi-lingual pretraining datasets, we also perform domain-adaptive pretraining of mT5 [101] and mBART [53] models for the tourism domain. Since all our tasks are generative, we chose mT5 and mBART as our primary model architectures. We show the efficacy of our models by finetuning them on the proposed TOURISMNLG benchmark tasks both individually as well as in a multi-task setup. This sets a good baseline for further researchers to compare their results on the TOURISMNLG benchmark.

Overall, we make the following contributions: (1) We propose a benchmark of five novel and diverse tourism NLG tasks called TOURISMNLG. As part of this benchmark, we also contribute four datasets along with standard splits to the research community. (2) We pretrain multiple tourism-domain specific models. We also make the pretrained models publicly available. (3) We experiment with multiple pretraining and finetuning setups, and present initial baseline results on the TOURISMNLG benchmark.

3.2 Related Work

3.2.1 Tourism-focused Data Science

Published work on data science in the tourism domain has been very sparse. It has been mainly focused on structured extraction of trip related information, mining reviews, and automatic itinerary generation. Popescu et al. [71] use tagged photos uploaded by tourists on Flickr to deduce trip related information such as visit times for a tourist spot, while Pantano et al. [67] build a model using online tourist reviews to predict tourists’ future preferences.

Automatic travel itinerary generation is another well-explored problem. De Choudhury et al. [20] construct intra-city travel itineraries using geo-temporal breadcrumbs. Friggstad et al. [25] propose an algorithm to provide high-quality tourist itineraries by maximizing the value of the worst day. Chang et al. [14] design an itinerary planning system that factors in monetary and time constraints.

²drive.google.com/file/d/1tux19cLoXclgz9Jwj9VebXmoRvF9MF6B/

Specifically in NLP, Gurjar et al. [29] study aspect extraction for the tourism domain for eleven factors (such as crowd, food, age, time, etc.), Kapoor et al. [42] identify travel-blog-worthy sentences from Wikipedia articles, and Iinuma et al. [37] propose a methodology that uses a graph-based approach to summarise multiple blog entries by finding important sentences and images. Finally, Antognini et al. [3] proposed a large dataset from the hotel domain, for the hotel recommendation task. Unfortunately, there is hardly any work on natural language generation for the tourism domain. We attempt to fill this gap in this chapter.

3.2.2 Domain-specific Pretrained Models

Numerous prior studies have introduced models that are trained for specific domains and their corresponding specialized tasks. Transformer models like BERT have been adapted to create pre-trained models such as BioBERT[46] for the Bio-Medical Domain, SciBERT[6] for scientific data domain, and other models like FinBERT[5] for NLP tasks in the financial domain, Covid-Twitter-BERT[58] trained on Covid related Twitter content and PatentBERT[47] for patent classification. Alsentzer et al. [2], one for generic clinical text and the other for discharge summaries. Additionally, there are models for the legal domain such as LegalBERT[13] and specialized models for conversational dialogues such as DialoGPT[105]. However, there are no domain-specific pretrained models available for the tourism domain. To address this gap, we propose the TOURISMNLG benchmark and associated initial models.

3.3 TOURISMNLG Benchmark

In this section, we present details of the four datasets and five NLG tasks which form the TOURISMNLG benchmark.

3.3.1 TOURISMNLG Datasets

The TOURISMNLG benchmark consists of four datasets: TravelWeb, TravelBlog, TripAdvisorQnA, and TravelWiki. These datasets were carefully chosen to cover diverse online content in the public domain.

TravelWeb: Given a large web crawl, we extract relevant webpages in the travel and tourism domain by utilizing a proprietary domain classifier based on [7]. From the resulting webpages, we only keep those that also appear in the mC4 dataset³, so that we can reuse cleaned text. We efficiently compute this intersection using Marisa trie[23]. The resulting dataset contains the URL, body text, and publication timestamp of 454553 documents published from 2013 to 2020, belonging to 80157 unique websites. We exclude instances where the body text is empty. Some of the most common websites in this dataset

³<https://huggingface.co/datasets/mc4>

include wikipedia, tripadvisor, britannica, rome2rio, lonelyplanet, maplandia, expedia, and theculture-trip.


TravelBlog: We gathered travel blogs from travelblog.org. The dataset contains the blog title, publication date, and body text. The dataset contains 491276 blogs from 2009 to 2020. The dataset is divided into ten geo-categories, with the following data split: Africa (33226), Antarctica (376), Asia (91626), Central America and Caribbean (21505), Europe (119202), Middle East (12093), North America (85270), Oceania (69783), Oceans and Seas (1802), and South America (56393). This dataset is used for the blog title generation task. We remove instances where the blog titles and/or body text are blank.

TripAdvisorQnA: We collect questions asked on TripAdvisor’s forums ⁴ to create a tourism-focused Question-Answering dataset. The dataset contains 217352 questions from various tourism-related categories, including Air Travel, Road Trips, Solo Travel, Cruises, Family Travel, and so on. We collect the question title, description, and all responses to the query from other forum members for each question. We also collect public user information for the user who asked the question, as well as information from users who responded. Furthermore, we record the number of ”helpful votes” for each response, indicating the usefulness of the responses as voted by others (see Fig 3.1 for an example of a tripadvisor forum layout consisting of a question and answers to it, along with the number of times the responses were voted as ”helpful”). We use this to assess the quality of responses to a specific question, and the most voted response serves as the gold standard for our tasks.

We discovered a set of standard messages from TripAdvisor staff that appeared frequently throughout the dataset and were not relevant to the discussion in the forum. To remove such comments, we identified at least three phrases in each of the language subsets. In English, for example, common messages included the phrases ”this post was determined to be inappropriate”, ”this post has been removed”, and ”message from tripadvisor staff”. Similarly, in Spanish, we removed examples where answers included phrases such as ”el personal de tripadvisor ha eliminado”, ”esta publicación ha sido eliminada por su autor”, ”no es posible responder a este tema ya que ha sido cerrado por inactividad”. Because the dataset is large, we only use one response per question, but all responses are included in the dataset. We remove instances where page titles are empty or where there is no answer.

TravelWiki: We gathered a list of top 1000 tourism spots worldwide from websites like lonelyplanet. Next, we discovered their Wikipedia pages (basic string match with typical normalizations) and gathered a histogram of Infobox template names for those Wikipedia pages. Further, we manually looked at the top 100 templates and identified a list of 29 Infobox templates like ”nrhp”, ”uk place”, ”mountain”, ”river”, etc. which seemed relevant to travel and tourism. We then collect the list of English Wikipedia

⁴<https://www.tripadvisor.<countrycode>/ListForums-g1-World.html>. We used these country codes: in, it, es, fr, de, pt, jp and ru.



Taxjay
Houston, Texas

Level 2 Contributor

- 1 post
- 8 reviews
- 10 helpful votes

Luggage storage at istanbul Airport

25 Nov 2022, 5:50 am

Hi:

What are the facilities at the Istanbul airport to store my luggage.

I have a 4 day halt in Istanbul and continue onward journey to India.

Thanks

Jay


Reply

Report inappropriate content

Save

3 replies to this topic

1-3 of 3 replies Sorted by Oldest first 1



GullibleGourmand
Istanbul, Turkiye

Level 6 Contributor

- 20,656 posts
- 14 reviews
- 20 helpful votes

1. Re: Luggage storage at istanbul Airport


25 Nov 2022, 11:32 am

<https://www.tripadvisor.co.uk/ShowTopic-g293974-i368-k1418211...> for a recent discussion.

Reply

Report inappropriate content

Save



enigma2007
Istanbul, Turkiye

Level 6 Contributor

- 53,261 posts
- 64 reviews
- 244 helpful votes

2. Re: Luggage storage at istanbul Airport

25 Nov 2022, 2:16 pm

Jay, as soon as you clear customs, the luggage storages are at each end of the arrivals hall.

Enigma...

Reply

Report inappropriate content

Save

Figure 3.1: An example of a forum discussion on TripAdvisor. Image-source: <https://www.tripadvisor.in/ShowTopic-g293974-i368-k14185302>

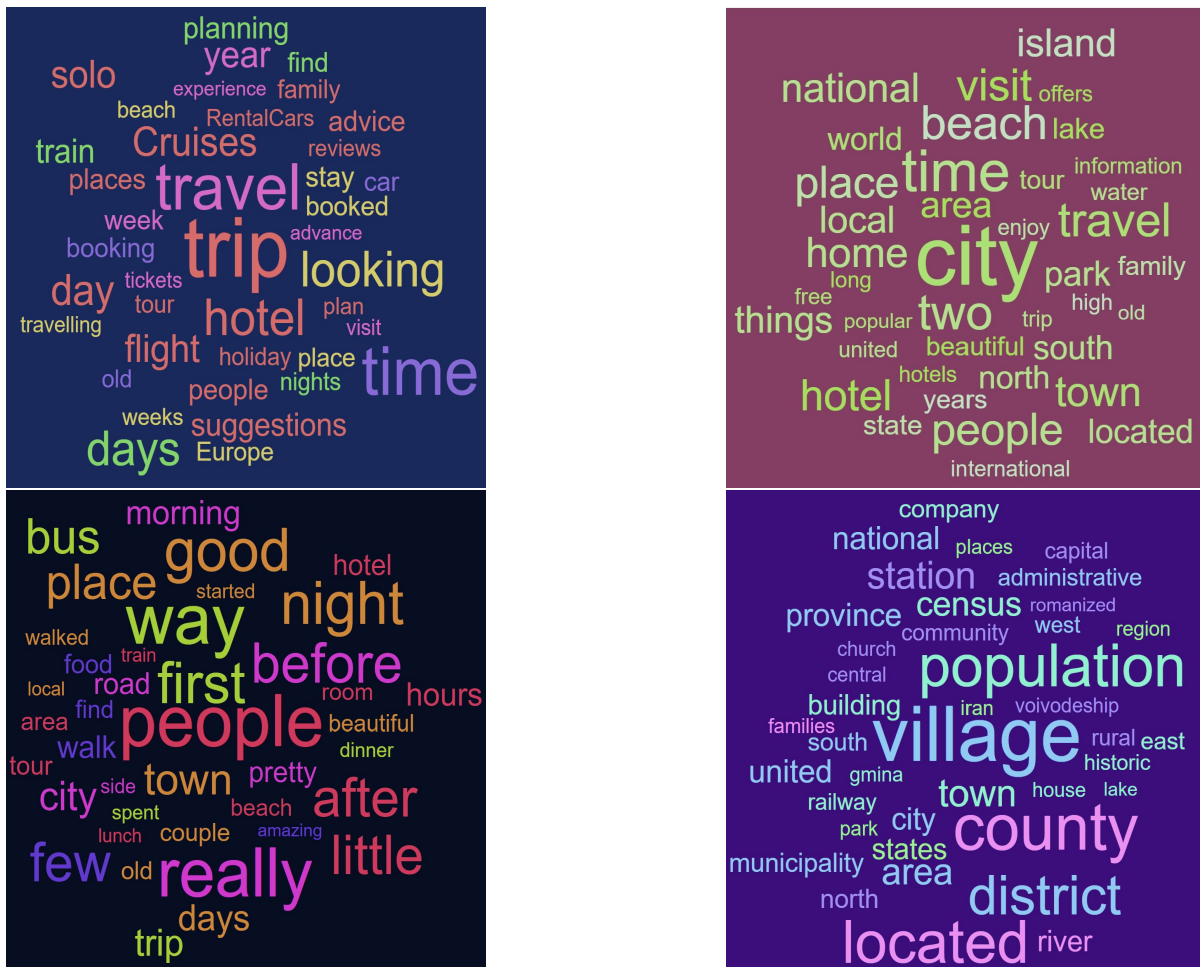


Figure 3.3: Word cloud for top few words in the text associated with TripAdvisorQnA (top-left), TravelWeb (top-right), TravelBlog (bottom-left) and TravelWiki (bottom-right) datasets.

value corresponding to a key. In the case of the paragraph generation task, all valid key-value pairs in the instance are used. We remove examples where the text is empty or if there are no key-value pairs in the infobox.

Figure 3.3 shows word clouds for the top few words in the text associated with TripAdvisorQnA, TravelWeb, TravelBlog and TravelWiki datasets. We manually removed stop words and create word clouds using English documents. We observe that formal words like ‘village’, ‘country’, ‘district’, ‘population’, etc. are frequent in TravelWiki. On the other hand, informal language with words like ‘people’, ‘really’, ‘way’, ‘night’, etc. is common in TravelBlog. Further, TripAdvisor has a lot of frequent words like ‘trip’, ‘travel’, ‘suggestions’, ‘time’, ‘hotel’, etc. which are related to advice around travel planning.

Document language is not explicitly known for these datasets except for TravelWiki. Hence, we predict the same using the langdetect library. Table 3.1 shows language distribution across languages

for these datasets. TravelBlog is heavily skewed towards English but TravelWiki and TripAdvisorQnA have higher representation from other languages.

Each dataset is split (stratified by language count) into four parts as follows: pretrain, finetune, validation and test. The pretrain part is used for pretraining, finetune part is used for task-specific finetuning, the validation part is used for early stopping as well as hyper-parameter tuning and the test part is used for reporting metrics. We allocate 7500 instances each for validation and testing, the remaining instances are divided equally into pretrain and finetune. We do not use TravelWeb dataset for any specific downstream task, thus for this dataset, we allocate 7500 instances for validation and the remaining for pretraining. Table 3.2 shows basic statistics of the datasets in TOURISMNLG.

3.3.2 TOURISMNLG Tasks

Our goal is to provide an accessible benchmark for the standard evaluation of models in the tourism domain. We select the tasks in the benchmark based on the following principles:

- Tourism specific: Tasks should be specific to tourism or defined on tourism-specific datasets.
- Task difficulty: Tasks should be sufficiently challenging.

Table 3.1: Language Distribution across the four datasets.

TravelWeb		TripAdvisorQnA		TravelBlog		TravelWiki	
Language	Docs (%)	Language	Docs (%)	Language	Docs (%)	Language	Docs (%)
en	70.93	en	58.84	en	88.84	en	27.28
de	4.91	it	21.84	de	2.26	pl	6.85
es	4.32	es	10.48	fr	2.14	fr	6.31
fr	4.21	fr	4.87	nl	1.19	fa	5.85
it	2.32	de	1.74	es	0.79	it	5.29
fa	1.85	pt	1.07	it	0.51	es	5.09
nl	1.81	ru	0.88	da	0.46	uk	4.71
pt	1.45	ja	0.16	fi	0.36	nl	4.66
pl	0.93	ca	0.02	no	0.29	sv	4.02
ru	0.85	nl	0.01	sk	0.26	de	3.81
Others	6.42	Others	0.09	Others	2.90	Others	26.13

Table 3.2: Characteristics of the datasets in TOURISMNLG.

Dataset	Domain	Pretrain	Finetune	Dev	Test	Tasks
TravelWeb	General	447053	-	7500	-	-
TripAdvisorQnA	Community Question Answering	101210	101142	7500	7500	Forum-Title Generation, Long Question Answering
TravelBlog	Social Media	238143	238133	7500	7500	Blog-Title Generation
TravelWiki	Encyclopedia	1531208	1531196	7500	7500	Paragraph Generation, Short Question Answering

- Task diversity: The generated output is of different sizes. Short QA generates very short answers. Blog-title generation and forum-title generation tasks generate sentence-sized outputs. Paragraph generation as well as Long QA tasks expect much longer outputs.
- Training efficiency: Tasks should be trainable on a single GPU for less than a day. This is to make the benchmark accessible, in particular to practitioners working with low resource languages under resource constraints.

TOURISMNLG consists of five generative NLP tasks. We give an overview of all tasks, including the average input and output sequence length for each task, in Table 3.3, and describe the tasks briefly as follows:

Paragraph Generation: The aim of this data-to-text generation task is to create the introductory paragraph of a Wikipedia article by using the (key, value) pairs extracted from the corresponding Wikipedia Infobox. By leveraging the information available in the infobox, the generated paragraph can provide a useful and informative introduction for a given topic. For example:

- Input: “nombre=Municipio de Benton; nombre_oficial=Municipio de Benton; unidad= Municipio; tipo_superior_1=Estado; tipo_superior_2=Condado; superior_2=Faulkner; mapa_loc=Arkansas; población=961; población_año=2010”.
- Output: “El municipio de Benton en inglés: Benton Township es un municipio ubicado en el condado de Faulkner en el estado estadounidense de Arkansas. En el año 2010 tenía una población de 961 habitantes y una densidad poblacional de 1269 personas por km²”.

Short QA: The objective of this task is to extract the accurate value for a given key in a Wikipedia Infobox by using the first paragraph of the corresponding Wikipedia article. This task is useful for the automated extraction of information from Wikipedia articles. For example:

- Input: “Tahannaout or Tahnaout is a town and commune capital of Al Haouz Province of the Marrakesh-Safi region of Morocco. It is located by road south of Marrakesh near the foot of the Atlas Mountains. It contains a Jewish cemetery”. Key: “Province”.
- Output: “Al Haouz Province”.

Blog-Title Generation: This task involves generating a title or headline for a travel blog article, based on the content of its body text. This is especially useful for blog writers and content creators, who often aim to capture the essence of an article in a few words to attract readers to engage with the content. For example:

- Input: “They say there are only two things to do in Malaysia - shopping and eating. Shopping doesn’t interest me, so I had never been tempted to go to Malaysia in spite of the fact that it is geographically close to India. When Air Asia started a service from Hyderabad (where I live) to Kuala Lumpur, I thought it was a good chance to go visit. It was a four hour flight to Kuala Lumpur...”
- Output: “A Gastronomical Journey in Malaysia”.

Forum-Title Generation: This task involves generating a title for a TripAdvisor forum page based on a given description and answer. The forum page typically includes a question posted by a user, along with one or more answers provided by other users or travel experts. The title serves as a brief summary of the content on the forum page, providing readers with an idea of what to expect from the discussion. A well-written title that is relevant to the forum discussion can help improve the relevance of search results, and also attract users to the forum page. For example:

- Input: “Hello, I’m looking for a place to trip to on the next weekend. A place with good food, not below 10 degrees or rainy in this season, and that will be fun to walk in its streets.I’m not like the ”ancient” places like Rome or Athens.Any recommendations?”
- Output: “Solo trip next weekend - where to?”

Long QA: The goal of this task is to provide an appropriate answer to a given question posted on a TripAdvisor Question-Answer forum page. The forum page contains a title and description of the question posed by a user seeking information or advice about a travel-related topic. Answering questions on the forum page is a crucial task since it helps travellers make informed decisions about their upcoming trips. An ideal answer is clear, concise, and directly addresses the question asked. It may include recommendations, tips, advice, and personal experiences to provide a comprehensive and valuable response. For example:

Table 3.3: Characteristics of the tasks in TOURISMNLG. $|I|$ = Avg Input Sequence Lengths (in words).
 $|O|$ = Avg Output Sequence Lengths (in words).

Task	Dataset	Input	Output	Metrics	$ I $	$ O $
Forum-Title Generation	TripAdvisorQnA	description, answer	title	ROUGE-1, ROUGE-L, METEOR, MRR	156.3	5.32
Long QA	TripAdvisorQnA	question ti- tle and de- scription	answer	ROUGE-1, ROUGE-L, METEOR, MRR	103.2	58.46
Paragraph Generation	TravelWiki	(key, value)	paragraph	ROUGE-1, ROUGE-L, METEOR, MRR	2.29	49.34
Short QA	TravelWiki	paragraph, key	value	F1, Accuracy, MRR	50.34	1.29
Blog-Title Generation	TravelBlog	body text	title	ROUGE-1, ROUGE-L, METEOR, MRR	793.9	4.41

- Input: “We are relocating from the Bay Area to Austin, Texas. We will be driving ourselves in a 2017 Honda Accord V6 towing a 5X8 cargo trailer. We will be leaving the Bay Area around January 19. We want to know if there is any possibility of snow in that time of the year on our way. If so is there any route that we can take to avoid dealing with the snow? We do not mind getting there a few hours later if we can avoid driving and towing the trailer in the snow. Thanks in advance for your responses”.
- Output: “Texas Canyon Rest Area West is scenic. We’ve driven on I-10 from Dallas to LA in December a few times”.

Table 3.4: Models used in TOURISMNLG

Characteristic	mT5-base	mBART-large-50
#Encoder layers	12	12
#Decoder layers	12	12
#Heads per layer	12	16
d_{model}	768	1024
Vocabulary size	250112	250054
#Parameters	582.40M	610.87M

3.4 Baseline Models for TOURISMNLG

In this work, our goal is to build generic pretrained models for the tourism domain which can be finetuned for individual tasks.

3.4.1 Model Selection

All of our tasks contain multi-lingual data and are generative in nature. mT5 [101] and mBART [53] are both multilingual encoder-decoder Transformer models and have been shown to be very effective across multiple NLP tasks like question answering, natural language inference, named entity recognition, etc. Thus, mT5 and mBART were natural choices for our purpose. mT5 [101] was pretrained on the mC4 dataset⁶ comprising of web data in 101 different languages and leverages a unified text-to-text format. mBART [53] was pretrained on the CommonCrawl corpus using the BART objective where the input texts are noised by masking phrases and permuting sentences, and a single Transformer model is learned to recover the texts. Details about the models we have used are provided in Table 3.4. Note that the two models have almost the same size.

We use the mC4-pretrained mT5-base and CommonCrawl-pretrained mBART-large models, and perform domain adaptive pretraining to adapt them to the tourism domain. These are then further finetuned using task-specific labeled data. We discuss pretraining and finetuning in detail later in this section.

mT5 requires every task to be modelled as sequence-to-sequence generation task preceded by a task prompt specifying the type of task. Thus, we use this format both while pretraining as well as finetuning. For the language modeling task while domain-specific pretraining, we use the task prefix “language-modeling”. For the downstream TOURISMNLG tasks, we use the following task prefixes: “infobox-2-para”, “para-2-infobox”, “blog-title-generation”, “forum-title-generation”, and “answer-generation”.

⁶https://www.tensorflow.org/datasets/catalog/c4#c4multilingual_nights_stay

Further, mBART also requires a language code to be passed as input. Thus, for mBART, we pass language code, task prefix and task-specific text as input⁷.

3.4.2 Pre-Training and Finetuning

For domain adaptive pretraining, we leverage our four datasets described in detail in the previous section. We pretrain mT5 as well as mBART using two different approaches: MLM and MLM+TASKS. MLM models have been pretrained only on masked language modeling (MLM) loss; MLM+TASKS models are pretrained using a combination of the MLM and task-specific losses. Pretraining tasks include masked language modeling on all the four datasets, paragraph generation and Short QA on Travel-Wiki, blog-title generation on TravelBlog and forum-title generation and Long QA on TripAdvisorQnA.

For MLM, the goal was to reconstruct the original text across all positions on the decoder side. The decoder input is the original text with one position offset. MLM uses text combined across pretrain parts of all datasets; large input sequences were chunked and masked to create training instances. All the other pretraining tasks are sequence generation tasks. Thus, the input was fed to the encoder and loss was defined with respect to tokens sampled at the decoder.

For pretraining, we use the standard categorical cross entropy (CCE) loss. For MLM, CCE is computed for masked words. For other tasks, CCE is computed for task-specific output words.

We finetune the pretrained models in two ways: (1) Single-task finetune (2) Multi-task finetune. Finetuning on individual tasks leads to one finetuned model per task. Managing so many models might be cumbersome. Thus, we also finetune one single model across all tasks. Another benefit of multi-task finetuning is that it can benefit from cross-task correlations.

3.4.3 Metrics

We evaluate our models using standard NLG metrics like ROUGE-1, ROUGE-L and METEOR for four tasks except for Short QA where we report F1 and accuracy (exact match). ROUGE measures the overlap between our model-generated text and reference texts. ROUGE-1 measures the overlap of unigrams between the generated text and the reference texts. ROUGE-L measures the longest common subsequence (LCS) between the generated text and the reference texts. The METEOR metric compares the model-generated output to the reference texts on a word-by-word basis, taking into account synonyms and stemming, and also considers the order of the words in the sentences.

However, these metrics are syntactic match-based and hence cannot appropriately evaluate predictions against the ground truth from a semantic perspective. For example, a blog title like “Trip to Bombay” is semantically very similar to “Five days of fun in Mumbai, Maharashtra, India” but has no word overlap.

One approach is to create a set with the prediction and K hard negative candidates and check if the predicted output is most similar to the ground truth. Hence, we use the popular mean reciprocal

⁷If the language of current instance was not among the 50 supported by the mBART model, we passed language=English.

rank metric (as also done in [83] for dialog quality evaluation) which is computed as follows: for every instance in the test set, we first gather 10 negative candidates. Given the predicted output, we rank the 11 candidates (1 ground truth and 10 negatives) and return the reciprocal of the rank of the ground truth. The ranking is done in the descending order of similarity between the prediction output and candidate text using the paraphrase-multilingual-MiniLM-L12-v2 model from Huggingface⁸. The equation for computing MRR can be seen below:

$$MRR = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{rank_i} \quad (3.1)$$

Here, Q represents the set of all queries, and $rank_i$ represents the rank of the ground truth for the i th query.

The negative candidates are sampled as follows: given a test instance and its ground-truth output, we compute the 20 most similar outputs from the train and dev sets of the same language. For the Short QA task, negatives are sampled from instances such that the “key” in the input also matches. Amongst the most similar 20 candidates, the top 10 are rejected since they could be very similar to ground truth and hence may not be negative. The remaining ten candidates are used as negative candidates. Note that these are fairly hard negatives and help differentiate clearly between strongly competing approaches.

3.4.4 Implementation Details for Reproducibility

For our experiments, we use 4 A100 GPUs, compatible with CUDA 11.0 and PyTorch 1.7.1. Our model is trained using a batch size of 16 and optimized with the AdamW optimizer. We perform both pre-training and fine-tuning for 3 epochs each. The maximum length for both input and output sequences is limited to 256. To generate predictions, we use a greedy decoding strategy.

Pretraining: We initialize our mT5 models using the google/mt5-base checkpoint and our mBART models using the facebook/mbart-large-50 checkpoint. We use a learning rate of 1e-5, and we use a dropout of 0.1. Our pretraining experiments take approximately 12, 26, 14 and 37 hours for mT5 MLM, mT5 MLM+Tasks, mBART MLM and mBART MLM+Tasks models respectively.

Finetuning: We use a learning rate of 5e-6 and 3e-6 for single-task finetune and multi-task finetune respectively.

3.5 Experiments and Results

In this section, we first present the main TOURISMNLG benchmark results using various proposed models. Next, we briefly present notes on pretraining stability, qualitative analysis of model outputs, human evaluation and detailed error analysis.

⁸<https://huggingface.co/sentence-transformers/paraphrase-multilingual-MiniLM-L12-v2>

Table 3.5: Results on TOURISMNLG Tasks: Long QA, Forum-title Generation, Paragraph Generation, Short QA and Blog-title Generation. For finetuning, STF=Single Task Finetune and MTF=Multi-task Finetune. For pretraining, (-) means no pretraining, (A) means MLM, (B) means MLM+Tasks. The best results in each block are highlighted.

Model		Long QA				Forum-title Generation			
		R-1	R-L	METEOR	MRR	R-1	R-L	METEOR	MRR
MTF	mT5 (-)	9.31	7.51	4.79	62.93	21.58	21.12	13.74	27.77
	mT5 (A)	10.55	8.18	5.35	64.20	22.09	21.50	14.13	27.53
	mT5 (B)	13.73	9.71	8.55	68.17	25.78	25.10	16.26	31.81
	mBART (-)	10.46	7.82	6.09	70.97	27.22	26.48	17.47	33.58
	mBART (A)	11.00	7.86	7.13	72.93	29.00	28.08	19.10	35.59
	mBART (B)	12.47	8.79	8.28	75.14	30.77	29.72	20.53	37.99
STF	mT5 (-)	10.22	7.36	7.57	60.61	15.70	15.33	10.10	22.21
	mT5 (A)	11.80	8.46	9.11	69.39	23.92	23.12	15.96	30.80
	mT5 (B)	11.03	8.37	7.62	72.49	28.59	27.57	18.96	35.07
	mBART (-)	13.17	9.20	9.22	75.94	31.55	30.21	21.61	37.73
	mBART (A)	12.39	8.81	9.17	75.63	31.65	30.38	21.56	38.16
	mBART (B)	13.82	9.88	9.65	76.12	33.00	31.56	22.30	39.42

Model		Paragraph Generation				Short QA		
		R-1	R-L	METEOR	MRR	F1	Accuracy	MRR
MTF	mT5 (-)	23.90	21.15	17.53	48.01	48.80	64.93	75.70
	mT5 (A)	32.27	28.85	22.22	54.61	58.98	73.68	82.83
	mT5 (B)	34.26	30.63	24.27	56.79	61.76	75.77	84.28
	mBART (-)	33.16	29.80	24.76	54.45	62.86	76.56	85.25
	mBART (A)	35.11	31.71	26.84	56.64	63.89	77.45	85.98
	mBART (B)	33.87	30.46	25.75	56.16	64.60	77.96	86.38
STF	mT5 (-)	19.73	17.48	12.87	33.93	48.80	65.00	75.40
	mT5 (A)	25.70	22.72	17.46	43.63	62.69	76.36	84.49
	mT5 (B)	26.08	22.88	19.14	44.84	64.17	77.60	85.51
	mBART (-)	35.23	31.31	28.04	53.16	69.66	81.37	88.36
	mBART (A)	31.73	28.26	24.18	50.07	71.07	82.39	89.10
	mBART (B)	35.62	31.43	27.30	55.59	71.17	82.40	89.16

Model		Blog-title Generation			
		R-1	R-L	METEOR	MRR
MTF	mT5 (-)	16.18	15.99	9.32	20.45
	mT5 (A)	15.86	15.75	8.93	20.86
	mT5 (B)	17.49	17.40	9.81	22.01
	mBART (-)	19.30	19.14	11.24	23.81
	mBART (A)	20.07	19.95	11.77	24.23
	mBART (B)	21.01	20.84	12.28	25.03
STF	mT5 (-)	12.48	12.31	7.69	16.11
	mT5 (A)	14.90	14.74	8.72	18.99
	mT5 (B)	17.98	17.85	10.41	21.96
	mBART (-)	20.44	20.21	12.99	23.98
	mBART (A)	20.99	20.70	13.21	24.00
	mBART (B)	21.86	21.59	13.74	24.95

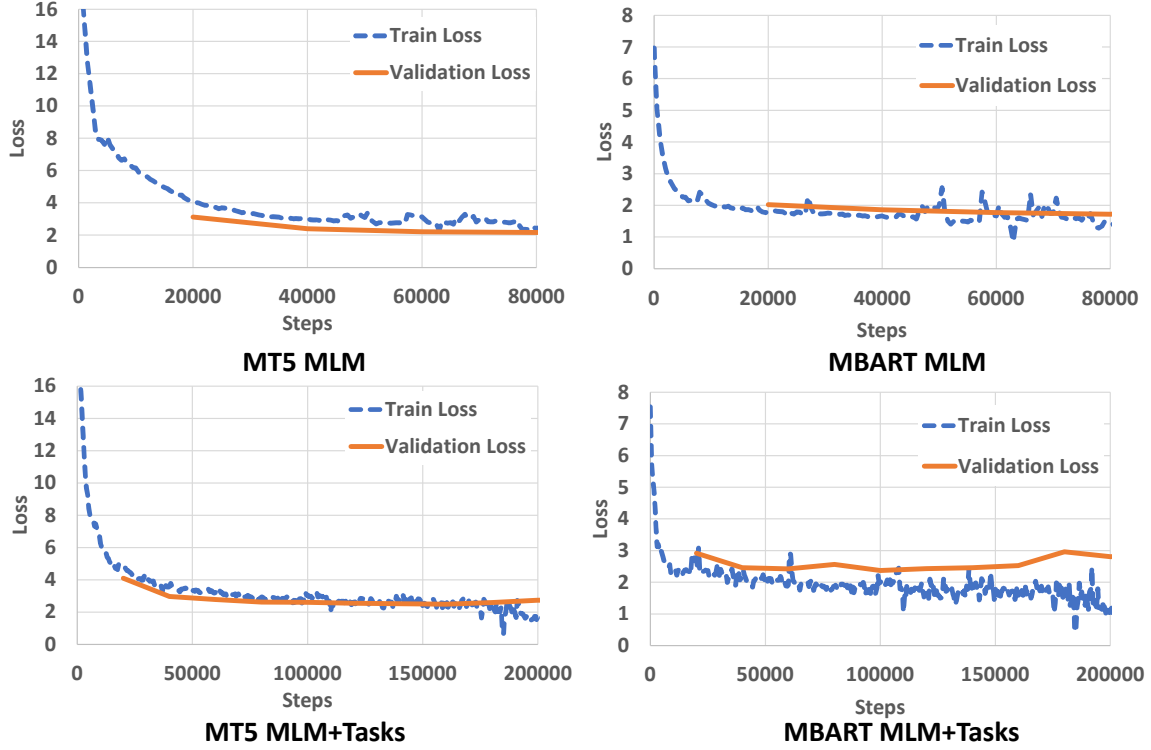


Figure 3.4: Variation in Pretraining Loss on Training as well as Validation Data for our mT5 and mBART models under the MLM and MLM+Tasks settings.

TOURISMNLG Benchmark Results: Table 3.5 shows results obtained using our models under various pretraining and finetuning setups for the five TOURISMNLG tasks on the test set. From the two tables we make the following observations:

- STF models lead to better results compared to MTF models. But MTF models are very close across all metrics. Thus, rather than retaining individual STF models, deploying just one MTF model is recommended.
- Domain-pretraining helps. Domain-pretrained models are better than standard models.
- Pretraining using MLM+Tasks is better than just MLM-based pretraining.
- Lastly, mBART models are significantly better than mT5 models except for the MTF Long QA setting.

Pretraining Stability: Fig 3.4 shows the variation in loss with epochs for the mT5 MLM, mT5 MLM+Tasks, mBART MLM and mBART MLM+Tasks models respectively.

Qualitative Analysis: Table 3.6 shows an example of generated output using our best model for each of the five tasks. Due to lack of space, we show shorter examples. We observe that the generated results are very relevant and well-formed.

Table 3.6: Examples of predictions using our best model

Task	Input	Output
Paragraph Generation	name = Quadyuk island; location = Bathurst Inlet; archipelago = Canadian Arctic Archipelago; country_admin_divisions = Nunavut; country_admin_divisions_title_1 = Region; country_admin_divisions_1 = Kitikmeot; population = Uninhabited; title = Quadyuk Island	quadyuk island is an uninhabited island in the canadian arctic archipelago in nunavut, canada. it is located in bathurst inlet and is part of the kitikmeot region.
Short QA	Mount Vernon is a home rule-class city and the seat of Rockcastle County, Kentucky in the United States. The intersection of US Routes 25 and 150 is located here. The population was 2477 at the time of the 2010 US census. Mount Vernon is part of the Richmond-Berea micropolitan area. Guess “county”	rockcastle
Blog-title Generation	Text from https://www.travelblog.org/Asia/China/Shanghai/Jing-an/blog-518801.html	day 1 - shanghai
Forum-title Generation	Hi, so I would love to hear from others who are well travelled & can give me an idea on the best place to visit for our 20th wedding anniversary & 40th Bday. We are thinking around September 2020 or April 2021 / not exactly sure yet but depends on some ideas. We are from Australia & are contemplating 2 - 3 week getaway...(continued, text from https://www.tripadvisor.in/ShowTopic-g1-i12522-k12187586)	20th anniversary trip ideas
Long QA	I sure would love some help deciding where we should take a family trip this summer. :) Just one daughter who is 12, almost 13. We'll have about one week total including travel. Criteria:- Not unbearably hot and awful in the summer. We went to DC last August...(continued, text from https://www.tripadvisor.in/ShowTopic-g1-i9658-k10343253)	i would look at the canadian rockies.

Table 3.7: Human Evaluation Results

Task	Fluency	Relevance
Paragraph Generation	4.18	3.36
Blog-title Generation	4.88	4.00
Forum-title Generation	4.82	4.08
Long QA	4.28	3.84

Human Evaluation and Error Analysis: Automated metrics do not always capture qualities such as fluency, readability, relevance, etc. in the generated text. Human evaluation is often considered more reliable than automatic metrics, as it takes into account factors such as context and background knowledge that are difficult to capture in automated evaluations. For our study, a manual evaluation of generated outputs is necessary to ensure whether the generated text appears 'good' to the reader. Therefore, to check fluency and relevance of the generated outputs for various tasks, one of the authors manually labeled 50 English samples per task on a 5-point scale. Note that we did not do such an evaluation for the Short QA task since the output is just the value (and not expected to be a well-formed sentence). Table 3.7 shows that our model generates human consumable output with high quality. Fluency measures the degree to which a text 'flows well', is coherent [1] and is not a sequence of unconnected parts. The below examples show what sentences are fluent/not fluent:

- Hyderabad is famous for its cuisine, especially the world-famous Biryani. (*Fluent*)
- Hyderabad cuisine famous Biryani, especially all over world. (*Not fluent*)

The other metric that we manually judge is relevance. Relevance measures correctness and the overall factual quality of the generated answer. For example, if our task is to generate a paragraph about Copenhagen, Denmark given its infobox, then the following examples would be considered relevant/irrelevant:

- Copenhagen is the capital and most populous city of Denmark with a population of around 2 million people in the metropolitan area. It is located on the islands of Zealand and Amager, and has an urban area of 525.50 square kilometres. (*Relevant*)
- Amsterdam is the capital and most populous city of the Netherlands with a population of 2.5 million people in the metropolitan area. It is located in the Dutch province of North Holland. (*Fluent but not relevant to Copenhagen, Denmark*)

In addition to scoring our outputs based on their fluency and relevance, we also performed an analysis of the kinds of errors in the generated outputs. Such an examination is necessary to identify the causes of these errors and help improve the performance of future models. These errors could have a number of causes: noise in the data, spelling mistakes (since a lot of the text is human-written, especially blogs),

Table 3.8: Error Analysis: # errors across categories for each task (out of 50 judged samples).

Error Category	Paragraph Generation	Blog-title Generation	Forum-title Generation	Long QA
Less Creative Response	5	3	0	5
Hallucination	15	5	6	3
Grammatical error	11	1	1	13
Incomplete	0	12	5	8

model training strategies, etc. Table 3.8 shows the distribution of errors across major categories. Some of our error categories are as follows:

- Less creative responses: Include cases where bland responses were generated, e.g., simply using the city name as the blog title, repeating the question as the answer for the long QA task or asking the user to post on another forum, or simply concatenating the key-value pairs as output for paragraph generation task.
- Incomplete category: Includes cases like blog/forum titles that do not take into account the entire context, or output that does not answer the user’s question completely in the long QA task.
- Hallucination: Includes cases where forum/blog titles have nothing to do with the input text, or unseen irrelevant information is added to the generated output.

3.6 Conclusions

In this chapter, we propose the first benchmark for NLG tasks in the tourism domain. The TOURISMNLG benchmark consists of five novel natural language generation tasks. We also pretrained mT5 and mBART models using various tourism domain-specific pretraining tasks and datasets. Our models lead to encouraging results on these novel tasks. We hope that our work will help further research on natural language generation in the travel and tourism domain. We expect that such models will help in writing automated tour guides, travel reviews and blogs, travel advisories, trip planning, multi-destination itinerary creation, and travel question answering. We plan to extend this work in the future by including multi-modal tasks and datasets like [26].

Chapter 4

Indian Language Summarization

4.1 Introduction

Automated text summarization is a technique for condensing lengthy documents while retaining their relevance. Text summarization for Indian languages has recently piqued the interest of the NLP community. However, due to a scarcity of high-quality datasets, progress in text summarization has been slow. Nonetheless, the availability of large-scale multilingual datasets like XL-Sum [30] and MassiveSumm [96] has resulted in significant progress in natural language generation and summarization tasks. While not perfect in terms of quality [94], these datasets are useful in terms of quantity. Furthermore, the field has undergone significant transformation as a result of recent advances in neural-based pretrained models.

The ILSUM challenge aims to create reusable data collections for summarising Indian languages. The collection is created by extracting news stories and their accompanying descriptions from publicly accessible news sources. The ILSUM dataset [84, 85] includes a summarization corpus for two important Indian languages, Hindi and Gujarati, as well as Indian English.

In this chapter, we provide an overview of the performance of existing sequence-to-sequence models that we used for our experiments. Our experiments yielded the best results across all three subtasks in the shared task (Hindi, Gujarati and Indian English). For Hindi and Gujarati, we used multilingual models such as mT5 [101], mBART [53] and IndicBART [19]. For English summarization experiments, we fine-tuned PEGASUS [103], BART [48], T5 [75] and ProphetNet [72]. We observe that for English, PEGASUS outperformed other models, while for Hindi, mT5 gave us the best results. For Gujarati, finetuning mBART yielded the best results. In addition to this, we ran various experiments on the dataset to combat model overfitting, such as k-fold cross-validation. We find that Hindi k-fold trials outperform experiments using the complete version of the provided data. We also use a number of filters to evaluate the quality of the released datasets. The efficacy of the pretrained generation models was later also analyzed using various combinations of our filtered data and the provided original data.

4.2 Related Work

Text summarization has received a great deal of attention, particularly in the English language. Early summarization research focused on extractive approaches, in which summary sentences were selected directly from the input text. Abstractive approaches to summarization, on the other hand, such as neural attention models [80], Seq2Seq RNNs [59], and Pointer-Generator networks [87], focus on generating summaries that capture the meaning of the input text without necessarily choosing sentences directly from the text. With the advent of large neural language models for generation tasks, abstractive approaches have grown in popularity and produce high-quality summaries. While there have been various improvements in model architectures and summarization techniques, a large part of the progress in English text summarization can be attributed to the availability of large-scale datasets, such as CNN/DailyMail [59, 34], Gigaword [80, 27], XSum [62], etc.

In contrast, little work has been done in summarization or related NLG tasks such as headline generation in Indian languages. However, there has been active research in this area recently, with the release of datasets such as XL-Sum [30], MassiveSumm [96], and others. These multilingual datasets are made up of article-summary pairs from publicly available news domains, including Indian languages like Hindi, Gujarati, Bengali, and so on. Several datasets for Indian language NLG tasks, such as sentence summarization and headline generation, have been released by the IndicNLG Suite [44]. More research is needed in this area to produce models that perform similarly to English summarization models.

4.3 Corpus Description

The dataset released for this task has been gathered from several leading Indian news sites such as India TV News¹, Divya Bhaskar², and News18 Gujarati³. One of the challenges of the dataset is that the Hindi and Gujarati examples include article-summary pairs that contain English words or phrases which have been code-mixed and script-mixed. We have also observed a few examples in the English and Gujarati datasets, where the summaries consist of only one word. Table 4.1 talks about the ILSUM training data statistics. We have used the Indic tokenizer [45] to generate the counts in Table 4.1.

4.4 Experiments and Results

We finetune various models, as mentioned in Table 4.4. A detailed description of these models can be found in Chapter 2. We have also used the recently proposed lightweight adapters[70] in some of our experiments, since they are effective at mitigating the overhead of PLMs for downstream tasks. In recent work[106], adapters were applied to perform Gujarati text summarization. Adapters can not only speed

¹<https://www.indiatvnews.com/>

²<https://www.divyabhaskar.co.in/>

³<https://gujarati.news18.com/>

Table 4.1: ILSUM Dataset Statistics

	English		Hindi		Gujarati	
#Pairs	12564		7957		8457	
	Text	Summary	Text	Summary	Text	Summary
#Avg. Words	595	36.24	553	40.17	414.43	32.26
Min. Words	1	1	17	6	25	1
Max. Words	5717	113	5034	113	2839	408
#Avg. Sentences	10.29	1.26	18.1	1.7	21.28	1.57
Min. Sentences	1	1	1	1	1	1
Max. Sentences	169	17	157	9	187	46

up training time but are also storage efficient since they require saving only adapter weights instead of entire finetuned model weights.

We ran experiments in two ways: one with the entire dataset and the other with a split dataset of ten folds, with 90% of the data used for training and 10% for validation. We used the released data for validation for testing purposes in both cases, and the results are shown in Table 4.2. It is worth noting that we had to conduct these k-fold cross-validation experiments to evaluate the performance of our model because we did not have access to any validation summaries.

To compute all of the scores, we use the standard ROUGE metric [51]. Our findings show that PEGASUS performs best on English when fine-tuned on the entire dataset during the validation phase. Meanwhile, during the validation phase, we got the best results when we finetuned IndicBART and mBART using both k-fold and complete data. It’s worth noting that fine-tuning a model with k-fold data can sometimes produce better results than fine-tuning with the entire dataset. This indicates that the dataset requires additional investigation, and appropriate filters should be implemented to determine which examples in the dataset are assisting the model in acquiring useful information.

We present the findings from the top-performing models in the test phase using the validation phase results. Although PEGASUS and mBART continue to provide the best results for English and Gujarati, respectively, when fine-tuned with k-fold data, mT5 outperforms IndicBART for Hindi. The hyper-parameter configurations used are detailed in table 4.4.

The multilingual models we used were pre-trained on large datasets, allowing them to handle the presence of code-mixing in the dataset effectively, which is also visible in their outputs. The models generate high-quality summaries and can incorporate relevant English text into Hindi and Gujarati examples. In particular, for Hindi and Gujarati, the average number of English words in training summaries is 0.25 and 1.91, respectively. Our models produced summaries with an average of 0.23 and 1.44 English words per summary for the test set in Hindi and Gujarati, respectively. Since many training

examples are entirely in Hindi and do not contain any English words or characters, it is important to note that the average number of English words in summaries in Hindi is lower.

4.5 Data Quality Assessment

To examine the quality of the data provided to us, we applied some of the filters described in TeSum [94]. Filters that were applied include checking whether there are:

- Empty instances
- Duplicate pairs and summaries within the dataset
- Cases where the first few sentences of the article itself are taken as the summary. We refer to this as a ‘prefix’ case. This filtering is done to ensure that the dataset does not contain trivial instances for system development, as mentioned in the MassiveSumm paper [96].
- Check whether the summary is ‘compressed enough’, i.e., we should not have summaries comparable in size to the text that has to be summarized. A good summary should result in a significant reduction in the size of the article while maintaining its relevance. Compression is a good measure of telling us if the summary provided is a shortened version of the input document/text or not.

The below example shows a summary present in the dataset that is not compressed enough as per our filters, and can also be considered as a prefix case:

- Article: “Magnitude 4.3 earthquake hits Hindukush region. An earthquake with a magnitude of 4.3 on the Richter scale hit Hindukush region today. According to the National Center for Seismology, the tremors were felt at 09:50 am. There were no reports of any loss of life or damage to property because of the earthquake”
- Summary: “An earthquake with a magnitude of 4.3 on the Richter scale hit Hindukush region today. According to the National Center for Seismology, the tremors were felt at 09:50 am.”

Filters counts for all the languages can be found in Table 4.5. It is worth noting that, according to our filters, only about 68% of the Hindi summaries are valid, as many are simply the first few sentences of the article. It could also be one of the reasons why models perform better on k-fold data. Some folds in the training data may contain a high percentage of high-quality, valid summaries while excluding a significant number of invalid summaries. It is worth noting that the number of final valid article-summary pairs in Gujarati and English is comparable to the original dataset size, which is why the top-performing models perform better when finetuned on the entire dataset rather than on k-fold subsets.

4.5.1 Data Variation Experiments

One of the main bottlenecks for neural models for text generation is the lack of large datasets. The summarization datasets for Indian languages that are currently available are quite small. We performed k-fold cross-validation on the best performing models to improve model generation capabilities on limited datasets (see Table 4.2). Table 4.6 reports the mean ROUGE scores and standard deviation scores over ten runs. Using the released training dataset, we performed 10-fold cross-validation with the following combinations:

- **Original data:** Finetuned for 5 epochs with the released training dataset
- **Original + Filtered data:** Finetuned for 3 epochs with original + 2 epochs with filtered data
- **Filtered data:** Finetuned for 5 epochs with only the filtered dataset
- **Filtered + Original data:** Finetuned for 3 epochs with filtered data + 2 epochs with original data

We used the filtered data obtained after applying the filters listed in Table 4.5 to carry out all of the experiments. To compare the models' performance on different variations of the training dataset, we have not made any changes in the validation data. As shown in Table 4.6, experiments with original data produce higher scores than experiments with filtered data. Furthermore, the models finetuned on the 'filtered + original' dataset performed better than the 'original+filtered' combination.

4.6 Conclusions

While better models trained solely for Indian languages may benefit research in the field of Indian Language Summarization, creating larger, high-quality datasets for such languages will undoubtedly lead to progress in this field. It may be worthwhile to look at sources other than news websites, and to keep the filters discussed earlier in mind while creating high-quality datasets.

We conclude that the pretrained transformer-based seq2seq models are capable of producing high-quality summaries for the ILSUM shared task.

Table 4.2: Results for Validation Data. * indicates that the model was finetuned on the combination of Hindi and Gujarati Data

Language	Model	Full Data/K-Fold	Validation scores		
			R-1	R-2	R-4
English	PEGASUS	Full Data	56.85	45.92	43.36
	T5 large	Full Data	56.05	45.03	42.36
	PEGASUS XSum	Full Data	54.66	43.48	40.64
	BRIO	Full Data	53.57	41.86	38.81
	BART large	K-Fold	54.83	43.58	40.71
	BART large XSum	K-Fold	53.35	41.74	38.75
	T5 base + Adapter	K-Fold	51.91	40.07	37.1
	ProphetNet	K-Fold	49.51	36.98	33.83
Hindi	IndicBART	K-Fold	60.73	51.26	47.57
	mT5 base	K-Fold	60.04	50.72	46.82
	IndicBART-SentSumm	K-Fold	58.09	47.99	43.72
	mT5 base*	Full Data	58.65	49.09	45.08
	mBART large 50 + Adapters	Full Data	56.26	45.56	41.21
	mBART large 50	Full Data	55.76	44.96	40.59
Gujarati	mBART large 50	Full Data	26.20	16.44	12.16
	mT5 base	Full Data	25.11	15.81	11.68
	mT5 base*	Full Data	24.16	14.68	10.79
	mBART large 50 + Adapter	Full Data	21.63	13.04	9.56
	IndicBART	K-Fold	23.38	13.34	9.35

Table 4.3: Results on Test Data

Language	Model	Full Data/K-Fold	Test scores		
			R-1	R-2	R-4
English	PEGASUS	Full Data	55.83	44.58	41.8
	T5 large	Full Data	54.73	43.08	40.12
Hindi	mT5 base	K-Fold	60.72	51.02	47.11
	IndicBART	K-Fold	58.38	48.31	44.25
Gujarati	mBART large 50	Full Data	26.11	16.51	12.41
	mBART large 50	Full Data (dropout=0.2)	26.07	16.60	12.58

Table 4.4: Experimental setup and parameter settings

Parameters	BART	T5	mBART	mT5	IndicBART	ProphetNet	PEGASUS	BRIO
Max source length	512	512	512	512	512	512	512	512
Max target length	75	75	75	100	75	75	75	75
Batch Size	2	1	4	2	2	1	2	2
Epochs	5	5	5	10	10	5	5	5
Learning Rate	5e-5	5e-5	5e-5	5e-5	5e-5	5e-5	5e-4	5e-5
Vocab Size	50265	32128	250054	250112	64015	30522	96103	50264
Beam Size	4	4	4	4	4	5	4	4

Table 4.5: Statistics of dataset examples that we consider valid after applying TeSum [94] filters

Filters	English	Hindi	Gujarati
Dataset Size	12565	7957	8457
Empty	1	0	0
Duplicate Pairs	0	23	0
Duplicate Summary	117	15	113
Compression <50%	182	11	37
Prefixes	486	2518	135
Final Valid	11779	5390	8172
Valid %	93.74%	67.74%	96.63%

Table 4.6: Mean ROUGE scores over the validation sets along with standard deviation over 10 runs.

O.D indicates Original Data, F.D indicates Filtered Data

Language	Model	Data composition	R-1	R-2	R-L
English	PEGASUS	O.D	52.51 ± 1.1	40.91 ± 1.36	47.81 ± 1.16
		O.D + F.D	51.65 ± 1.14	40.07 ± 1.25	46 ± 3.67
		F.D	51.88 ± 1.25	40.37 ± 1.39	47.32 ± 1.31
		F.D + O.D	53.28 ± 1.18	41.82 ± 1.3	48.67 ± 1.2
	T5 large	O.D	53.45 ± 0.95	42.16 ± 1.13	48.97 ± 1.05
		O.D + F.D	53.22 ± 1.23	42.04 ± 1.41	48.85 ± 1.31
		F.D	51.9 ± 1.37	40.49 ± 1.53	47.38 ± 1.46
		F.D + O.D	53.33 ± 0.83	42.1 ± 0.96	48.92 ± 0.86
	BART large	O.D	50.25 ± 1.52	38.15 ± 1.85	45.46 ± 1.63
		O.D + F.D	51.42 ± 0.88	39.85 ± 1.11	46.93 ± 1
		F.D	51.21 ± 1.3	39.83 ± 1.57	46.79 ± 1.38
		F.D + O.D	52.45 ± 1.05	40.98 ± 1.29	48 ± 1.17
Hindi	IndicBART	O.D	26.36 ± 1.02	12.66 ± 0.73	26.28 ± 0.98
		O.D + F.D	21.58 ± 0.66	9.84 ± 0.76	21.45 ± 0.6
		F.D	21.27 ± 0.88	9.75 ± 0.56	21.12 ± 0.86
		F.D + O.D	25.67 ± 1.04	12.16 ± 0.82	25.57 ± 1
	mT5 base	O.D	27.04 ± 1.22	13.21 ± 0.61	26.96 ± 1.22
		O.D + F.D	20.33 ± 0.91	9.26 ± 0.8	20.2 ± 0.92
		F.D	20.61 ± 1.55	9.47 ± 0.67	20.51 ± 1.53
		F.D + O.D	26.73 ± 1.11	12.83 ± 0.61	26.64 ± 1.1
Gujarati	mBART large 50	O.D	20.36 ± 0.67	11.65 ± 1.13	20.01 ± 0.72
		O.D + F.D	16.04 ± 1.12	9.23 ± 0.76	15.83 ± 1.15
		F.D	12.82 ± 2.28	6.6 ± 1.54	12.38 ± 2.36
		F.D + O.D	19.55 ± 0.74	11.42 ± 0.43	19.2 ± 0.72
	mT5 base	O.D	21.55 ± 0.77	11.81 ± 0.78	21.19 ± 0.83
		O.D + F.D	18.63 ± 0.93	9.23 ± 0.5	18.19 ± 0.92
		F.D	9.66 ± 0.97	4.84 ± 0.56	9.53 ± 0.92
		F.D + O.D	20.29 ± 0.62	10.7 ± 0.52	19.84 ± 0.56

Chapter 5

Conclusions and Future Work

5.1 Conclusions

Natural Language Generation and tasks such as Headline Generation, Summarization, etc. are well-explored problems in NLP, with many models capable of producing high-quality results. However, as we move towards making NLP systems deployable in the real world, there is a strong need for these systems to be able to address the very specific problems faced by different domains. Creating datasets for such domains and tasks, and having models specifically trained on these domain-specific tasks is the way forward.

We focus specifically on the travel and tourism domain, due to the lack of a standard benchmark and limited work on data science problems in this domain. We also take into account the need for NLP systems to be able to handle problems across different languages, and hence, multi-lingual datasets and models are key topics of interest in our research.

In **Chapter 1** and **Chapter 2** of this thesis, we presented an introduction of our research work, as well as the basic definitions and terminologies used. We highlighted the motivation for our work in this field and its importance in today’s world. We introduced various pretrained models and their applications, and also presented an overview of research that has been done in this field so far, from domain-based, language-based and task-based perspectives.

In **Chapter 3** of this thesis, we looked at the relatively unexplored tourism domain, and we proposed the first benchmark for NLG tasks in the tourism domain. We contributed the TOURISMNLG benchmark, which consists of five novel natural language generation tasks, ranging from headline generation to question answering. We contributed datasets for the same, consisting of data from various travel-specific sources. The nature of text across these sources is varied (informal, formal, factual, etc.), which reflects diversity in our choice of datasets. We also pretrained mT5 and mBART models using various tourism domain-specific pretraining tasks and datasets.

Our models lead to encouraging results on these novel tasks, and we conclude that pretraining models for a specific domain does lead to improvement in performance for tasks in that domain. We have made our code, data and pretrained models publicly available

In **Chapter 4**, we looked at the problem of Indian Language Summarization, in which our dataset belonged to the news domain. We presented a detailed evaluation of various PLMs for summarization in different languages, and show that they are capable of generating good-quality summaries. We also emphasized rules to keep in mind while creating summarization datasets, which can also be applied to NLG datasets in general.

5.2 Future Work

We hope that our work will help further research on natural language generation in the travel and tourism domain, and will motivate others to carry out research specific to other relatively unexplored domains, that have their own niche tasks and problems that need to be addressed.

Based on this thesis, there are several directions in which future work could be carried out:

- **Tourism-specific challenges:** We expect that such models will help in writing automated tour guides, travel reviews and blogs, trip planning, travel advisories, multi-destination itinerary creation, and travel question answering. As the travel industry continues to boom worldwide, we expect more research on data science problems in this domain and expect different parties (researchers, businesses, individuals, etc.) to use our dataset and models for their experiments and use cases.
- **Improving support for Indian Languages across domains:** While our datasets and models are multilingual in nature, and also have a significant number of Indian language examples, we believe that one of the directions in which research can proceed is to have datasets and models specifically for Indian languages in specific domains.

This research need not be restricted to the tourism domain. Most of the existing work on domain-specific pretrained models does not account for a number of Indian languages, and even if it does, the data available is either limited or of low quality. It is important that there is research in this direction. This would have positive implications for both academic literature as well as businesses due to its potential to boost domestic tourism in India.

This is also applicable to summarization tasks. As highlighted in Chapter 4, Indian language summarization datasets are far from perfect, and there is a need to come up with good-quality datasets and models that can serve as a benchmark for future research.

- **Classification tasks:** Researchers could make use of our dataset and also focus on tasks that are not necessarily generative in nature. For instance, classification tasks involving labels such

as 'suitable to visit in summer', 'family-friendly', etc. could be proposed to assign labels to travel destinations based on their descriptions and knowledge learned during pretraining. Through transfer learning, the knowledge learned by our tourism-centric pretrained models could be used to solve such tasks via finetuning.

- **Pretraining and Prompting strategies:** Different pretraining approaches could be explored, involving different loss functions and hyperparameter optimizations to improve performance across tasks.

Additionally, as large language models (LLMs, such as GPT-3 [8], OPT [104], etc.) become more popular and accessible to the public, research could also focus on prompt engineering techniques to generate coherent, accurate and fluent outputs. Few-shot learning techniques, especially those involving chain-of-thought prompting, have been shown to yield good results on new tasks, thus enabling these models to easily scale to newer domains and their respective tasks.

- **Multi-modal tasks:** Text generation capabilities for specific domains can be improved by incorporating additional information such as images, videos or speech. Future work could consider multi-modal tasks in multiple directions. From an image-to-text direction, we could identify problems and curate datasets corresponding to tasks such as generating creative text for a given location based on photos, videos, etc. From the text-to-image direction, we believe that as image generation models (such as stable diffusion models [79]) get more and more popular, research could also focus on tasks like blog-image generation, similar to our blog-title generation task in Chapter 3, where the model generates a photo-realistic image (instead of a blog title) based on a user-written blog description.

Related Publications

Relevant Publications

- **Sahil Manoj Bhatt**, Sahaj Agarwal, Omkar Gurjar, Manish Gupta, Manish Shrivastava. **TOURISMNLG: A Multi-lingual Generative Benchmark for the Tourism Domain**. Accepted at the 45th European Conference on Information Retrieval (ECIR-2023), April 2-6, 2023, Dublin, Ireland.
- Ashok Urlana, **Sahil Manoj Bhatt**, Nirmal Surange, Manish Shrivastava. **Indian Language Summarization using Pretrained Sequence-to-Sequence Models**. Accepted at the Forum for Information Retrieval Evaluation (FIRE-2022) Working Notes, December 9-13, 2022, Kolkata, India.

Other Publications

- **Sahil Bhatt**, Manish Shrivastava. **Tesla at SemEval-2022 Task 4: Patronizing and Condescending Language Detection using Transformer-based Models with Data Augmentation**. Accepted at Semeval-2022, Seattle, USA.

Bibliography

- [1] T. Abhishek, D. Rawat, M. Gupta, and V. Varma. Fact aware multi-task learning for text coherence modeling. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 340–353. Springer, 2022.
- [2] E. Alsentzer, J. R. Murphy, W. Boag, W.-H. Weng, D. Jin, T. Naumann, and M. McDermott. Publicly available clinical bert embeddings. *arXiv preprint arXiv:1904.03323*, 2019.
- [3] D. Antognini and B. Faltings. Hotelrec: a novel very large-scale hotel recommendation dataset. *arXiv preprint arXiv:2002.06854*, 2020.
- [4] W. Antoun, F. Baly, and H. M. Hajj. Arabert: Transformer-based model for arabic language understanding. *CoRR*, abs/2003.00104, 2020.
- [5] D. Araci. Finbert: Financial sentiment analysis with pre-trained language models. *arXiv preprint arXiv:1908.10063*, 2019.
- [6] I. Beltagy, K. Lo, and A. Cohan. Scibert: A pretrained language model for scientific text. *arXiv preprint arXiv:1903.10676*, 2019.
- [7] P. Bennett, K. Svore, and S. Dumais. Classification-enhanced ranking. In *World Wide Web Conference (WWW)*, pages 111–120, 2010.
- [8] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. Language models are few-shot learners. *CoRR*, abs/2005.14165, 2020.
- [9] S. Cahyawijaya, G. I. Winata, B. Wilie, K. Vincentio, X. Li, A. Kuncoro, S. Ruder, Z. Y. Lim, S. Bahar, M. Khodra, A. Purwarianti, and P. Fung. IndoNLG: Benchmark and resources for evaluating Indonesian natural language generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8875–8898, Online and Punta Cana, Dominican Republic, Nov. 2021. Association for Computational Linguistics.
- [10] I. Chalkidis, I. Androutsopoulos, and N. Aletras. Neural legal judgment prediction in English. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4317–4323, Florence, Italy, July 2019. Association for Computational Linguistics.

- [11] I. Chalkidis, I. Androutsopoulos, and A. Michos. Extracting contract elements. In *Proceedings of the 16th Edition of the International Conference on Artificial Intelligence and Law, ICAIL '17*, page 19–28, New York, NY, USA, 2017. Association for Computing Machinery.
- [12] I. Chalkidis, E. Fergadiotis, P. Malakasiotis, N. Aletras, and I. Androutsopoulos. Extreme multi-label legal text classification: A case study in EU legislation. In *Proceedings of the Natural Legal Language Processing Workshop 2019*, pages 78–87, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.
- [13] I. Chalkidis, M. Fergadiotis, P. Malakasiotis, N. Aletras, and I. Androutsopoulos. Legal-bert: The muppets straight out of law school. *arXiv preprint arXiv:2010.02559*, 2020.
- [14] H.-T. Chang, Y.-M. Chang, and M.-T. Tsai. Atips: automatic travel itinerary planning system for domestic areas. *Computational intelligence and neuroscience*, 2016, 2016.
- [15] G. Chen, S. Wu, J. Zhou, and A. K. Tung. Automatic itinerary planning for traveling services. *IEEE transactions on knowledge and data engineering*, 26(3):514–527, 2013.
- [16] Y. Chen and M. Bansal. Fast abstractive summarization with reinforce-selected sentence rewriting. *CoRR*, abs/1805.11080, 2018.
- [17] J. Cheng and M. Lapata. Neural summarization by extracting sentences and words. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 484–494, Berlin, Germany, Aug. 2016. Association for Computational Linguistics.
- [18] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov. Unsupervised cross-lingual representation learning at scale, 2019.
- [19] R. Dabre, H. Shrotriya, A. Kunchukuttan, R. Puduppully, M. M. Khapra, and P. Kumar. Indicbart: A pre-trained model for natural language generation of indic languages. *arXiv preprint arXiv:2109.02903*, 2021.
- [20] M. De Choudhury, M. Feldman, S. Amer-Yahia, N. Golbandi, R. Lempel, and C. Yu. Automatic construction of travel itineraries using social breadcrumbs. In *Proceedings of the 21st ACM Conference on Hypertext and Hypermedia*, pages 35–44, 2010.
- [21] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.
- [22] R. I. Doğan, R. Leaman, and Z. Lu. Ncbi disease corpus: A resource for disease name recognition and concept normalization. *Journal of Biomedical Informatics*, 47:1–10, 2014.
- [23] S. Ercoli, M. Bertini, and A. Del Bimbo. Compact hash codes and data structures for efficient mobile visual search. In *2015 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, pages 1–6. IEEE, 2015.
- [24] W. E. Forum. This is the impact of covid-19 on the travel sector. <https://www.weforum.org/agenda/2022/01/global-travel-tourism-pandemic-covid-19/>.

- [25] Z. Friggstad, S. Gollapudi, K. Kollias, T. Sarlos, C. Swamy, and A. Tomkins. Orienteering algorithms for generating travel itineraries. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, pages 180–188, 2018.
- [26] P. Gatti, A. Mishra, M. Gupta, and M. D. Gupta. Vistot: Vision-augmented table-to-text generation. In *EMNLP*, 2022.
- [27] D. Graff, J. Kong, K. Chen, and K. Maeda. English gigaword. *Linguistic Data Consortium, Philadelphia*, 4(1):34, 2003.
- [28] M. Grusky, M. Naaman, and Y. Artzi. Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 708–719, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.
- [29] O. Gurjar and M. Gupta. Should i visit this place? inclusion and exclusion phrase mining from reviews. In *European Conference on Information Retrieval*, pages 287–294. Springer, 2021.
- [30] T. Hasan, A. Bhattacharjee, M. S. Islam, K. Samin, Y.-F. Li, Y.-B. Kang, M. S. Rahman, and R. Shahriyar. Xl-sum: Large-scale multilingual abstractive summarization for 44 languages. *arXiv preprint arXiv:2106.13822*, 2021.
- [31] J. He, W. Kryscinski, B. McCann, N. F. Rajani, and C. Xiong. Ctrlsum: Towards generic controllable text summarization. *CoRR*, abs/2012.04281, 2020.
- [32] P. He, J. Gao, and W. Chen. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. *CoRR*, abs/2111.09543, 2021.
- [33] P. He, X. Liu, J. Gao, and W. Chen. Deberta: Decoding-enhanced BERT with disentangled attention. *CoRR*, abs/2006.03654, 2020.
- [34] K. M. Hermann, T. Kociský, E. Grefenstette, L. Espeholt, W. Kay, M. Suleyman, and P. Blunsom. Teaching machines to read and comprehend. *CoRR*, abs/1506.03340, 2015.
- [35] B. Hu, Q. Chen, and F. Zhu. LCSTS: A large scale Chinese short text summarization dataset. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1967–1972, Lisbon, Portugal, Sept. 2015. Association for Computational Linguistics.
- [36] K. Huang, J. Altsosaar, and R. Ranganath. Clinicalbert: Modeling clinical notes and predicting hospital readmission. *arXiv preprint arXiv:1904.05342*, 2019.
- [37] S. Iinuma, H. Nanba, and T. Takezawa. Automatic summarization of multiple travel blog entries focusing on travelers’ behavior. In *Information and Communication Technologies in Tourism 2018*, pages 129–142. Springer, 2018.
- [38] D. Jin, Z. Jin, J. T. Zhou, L. Oriei, and P. Szolovits. Hooks in the headline: Learning to generate headlines with controlled styles. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5082–5093, Online, July 2020. Association for Computational Linguistics.

- [39] Q. Jin, B. Dhingra, Z. Liu, W. Cohen, and X. Lu. PubMedQA: A dataset for biomedical research question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2567–2577, Hong Kong, China, Nov. 2019. Association for Computational Linguistics.
- [40] D. Jurgens, S. Kumar, R. Hoover, D. McFarland, and D. Jurafsky. Measuring the evolution of a scientific field through citation frames. *Transactions of the Association for Computational Linguistics*, 6:391–406, 2018.
- [41] D. Kakwani, A. Kunchukuttan, S. Golla, G. N.C., A. Bhattacharyya, M. M. Khapra, and P. Kumar. IndicNLP Suite: Monolingual Corpora, Evaluation Benchmarks and Pre-trained Multilingual Language Models for Indian Languages. In *Findings of EMNLP*, 2020.
- [42] A. Kapoor and M. Gupta. Identifying relevant sentences for travel blogs from wikipedia articles. In *International Conference on Database Systems for Advanced Applications*, pages 532–536. Springer, 2022.
- [43] S. Khanuja, D. Bansal, S. Mehtani, S. Khosla, A. Dey, B. Gopalan, D. K. Margam, P. Aggarwal, R. T. Nagipogu, S. Dave, S. Gupta, S. C. B. Gali, V. Subramanian, and P. Talukdar. MuriL: Multilingual representations for indian languages, 2021.
- [44] A. Kumar, H. Shrotriya, P. Sahu, R. Dabre, R. Puduppully, A. Kunchukuttan, A. Mishra, M. M. Khapra, and P. Kumar. IndicNLP suite: Multilingual datasets for diverse nlp tasks in indic languages, 2022.
- [45] A. Kunchukuttan. The IndicNLP Library. https://github.com/anoopkunchukuttan/indic_nlp_library/blob/master/docs/indicnlp.pdf, 2020.
- [46] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, and J. Kang. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240, 2020.
- [47] J.-S. Lee and J. Hsiang. Patentbert: Patent classification with fine-tuning a pre-trained bert model. *arXiv preprint arXiv:1906.02124*, 2019.
- [48] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*, 2019.
- [49] J. Li, T. Tang, W. X. Zhao, and J.-R. Wen. Pretrained language models for text generation: A survey. *arXiv preprint arXiv:2105.10311*, 2021.
- [50] Y. Liang, N. Duan, Y. Gong, N. Wu, F. Guo, W. Qi, M. Gong, L. Shou, D. Jiang, G. Cao, X. Fan, B. Zhang, R. Agrawal, E. Cui, S. Wei, T. Bharti, Y. Qiao, J. Chen, W. Wu, S. Liu, F. Yang, R. Majumder, and M. Zhou. XGLUE: A new benchmark dataset for cross-lingual pre-training, understanding and generation. *CoRR*, abs/2004.01401, 2020.
- [51] C.-Y. Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics.
- [52] D. Liu, Y. Gong, Y. Yan, J. Fu, B. Shao, D. Jiang, J. Lv, and N. Duan. Diverse, controllable, and keyphrase-aware: A corpus and method for news multi-headline generation. In *Proceedings of the 2020 Conference*

- on *Empirical Methods in Natural Language Processing (EMNLP)*, pages 6241–6250, Online, Nov. 2020. Association for Computational Linguistics.
- [53] Y. Liu, J. Gu, N. Goyal, X. Li, S. Edunov, M. Ghazvininejad, M. Lewis, and L. Zettlemoyer. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742, 2020.
 - [54] Y. Liu, P. Liu, D. Radev, and G. Neubig. Brio: Bringing order to abstractive summarization. *arXiv preprint arXiv:2203.16804*, 2022.
 - [55] L. Martin, B. Muller, P. J. Ortiz Suárez, Y. Dupont, L. Romary, É. de la Clergerie, D. Seddah, and B. Sagot. CamemBERT: a tasty French language model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7203–7219, Online, July 2020. Association for Computational Linguistics.
 - [56] R. Mihalcea and P. Tarau. TextRank: Bringing order into text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 404–411, Barcelona, Spain, July 2004. Association for Computational Linguistics.
 - [57] A. Morales, V. Premtoon, C. Avery, S. Felshin, and B. Katz. Learning to answer questions from Wikipedia infoboxes. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1930–1935, Austin, Texas, Nov. 2016. Association for Computational Linguistics.
 - [58] M. Müller, M. Salathé, and P. E. Kummervold. Covid-twitter-bert: A natural language processing model to analyse covid-19 content on twitter. *arXiv preprint arXiv:2005.07503*, 2020.
 - [59] R. Nallapati, B. Xiang, and B. Zhou. Sequence-to-sequence rnns for text summarization. *CoRR*, abs/1602.06023, 2016.
 - [60] R. Nallapati, F. Zhai, and B. Zhou. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. *CoRR*, abs/1611.04230, 2016.
 - [61] C. Napoles, M. Gormley, and B. Van Durme. Annotated Gigaword. In *Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction (AKBC-WEKEX)*, pages 95–100, Montréal, Canada, June 2012. Association for Computational Linguistics.
 - [62] S. Narayan, S. B. Cohen, and M. Lapata. Don’t give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. *ArXiv*, abs/1808.08745, 2018.
 - [63] T. Nguyen, M. Rosenberg, X. Song, J. Gao, S. Tiwary, R. Majumder, and L. Deng. MS MARCO: A human generated machine reading comprehension dataset. *CoRR*, abs/1611.09268, 2016.
 - [64] B. Nye, J. J. Li, R. Patel, Y. Yang, I. Marshall, A. Nenkova, and B. Wallace. A corpus with multi-level annotations of patients, interventions and outcomes to support language processing for medical literature. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 197–207, Melbourne, Australia, July 2018. Association for Computational Linguistics.

- [65] Özlem Uzuner, B. R. South, S. Shen, and S. L. DuVall. 2010 i2b2/VA challenge on concepts, assertions, and relations in clinical text. *Journal of the American Medical Informatics Association*, 18(5):552–556, June 2011.
- [66] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab, November 1999. Previous number = SIDL-WP-1999-0120.
- [67] E. Pantano, C.-V. Priporas, and N. Stylos. ‘you will like it!’ using open data to predict tourists’ response to a tourist attraction. *Tourism Management*, 60:430–438, 2017.
- [68] S. Park, J. Moon, S. Kim, W. I. Cho, J. Han, J. Park, C. Song, J. Kim, Y. Song, T. Oh, J. Lee, J. Oh, S. Lyu, Y. Jeong, I. Lee, S. Seo, D. Lee, H. Kim, M. Lee, S. Jang, S. Do, S. Kim, K. Lim, J. Lee, K. Park, J. Shin, S. Kim, L. Park, A. Oh, J.-W. Ha, and K. Cho. Klue: Korean language understanding evaluation, 2021.
- [69] R. Paulus, C. Xiong, and R. Socher. A deep reinforced model for abstractive summarization. *CoRR*, abs/1705.04304, 2017.
- [70] J. Pfeiffer, A. Rücklé, C. Poth, A. Kamath, I. Vulić, S. Ruder, K. Cho, and I. Gurevych. Adapterhub: A framework for adapting transformers. *arXiv preprint arXiv:2007.07779*, 2020.
- [71] A. Popescu and G. Grefenstette. Deducing trip related information from flickr. In *Proceedings of the 18th international conference on World Wide Web*, pages 1183–1184, 2009.
- [72] W. Qi, Y. Yan, Y. Gong, D. Liu, N. Duan, J. Chen, R. Zhang, and M. Zhou. Prophetnet: Predicting future n-gram for sequence-to-sequence pre-training. *arXiv preprint arXiv:2001.04063*, 2020.
- [73] A. Radford and K. Narasimhan. Improving language understanding by generative pre-training. 2018.
- [74] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8), 2019.
- [75] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv:1910.10683*, 2019.
- [76] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang. Squad: 100, 000+ questions for machine comprehension of text. *CoRR*, abs/1606.05250, 2016.
- [77] S. Reddy, D. Chen, and C. D. Manning. CoQA: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266, 2019.
- [78] A. Romanov and C. Shivade. Lessons from natural language inference in the clinical domain. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1586–1596, Brussels, Belgium, Oct.-Nov. 2018. Association for Computational Linguistics.
- [79] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-resolution image synthesis with latent diffusion models, 2021.
- [80] A. M. Rush, S. Chopra, and J. Weston. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389, Lisbon, Portugal, Sept. 2015. Association for Computational Linguistics.

- [81] A. Safaya, E. Kurtuluş, A. Goktogan, and D. Yuret. Mukayese: Turkish NLP strikes back. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 846–863, Dublin, Ireland, May 2022. Association for Computational Linguistics.
- [82] Sandhaus, Evan. The new york times annotated corpus, 2008.
- [83] B. Santra, S. Roychowdhury, A. Mandal, V. Gurram, A. Naik, M. Gupta, and P. Goyal. Representation learning for conversational data using discourse mutual information maximization. In *Proceedings of the 2022 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 2022.
- [84] S. Satapara, B. Modha, S. Modha, and P. Mehta. Findings of the first shared task on indian language summarization (ilsum): Approaches, challenges and the path ahead. In *Working Notes of FIRE 2022 - Forum for Information Retrieval Evaluation, Kolkata, India, December 9-13, 2022*, CEUR Workshop Proceedings. CEUR-WS.org, 2022.
- [85] S. Satapara, B. Modha, S. Modha, and P. Mehta. Fire 2022 ilsum track: Indian language summarization. In *Proceedings of the 14th Forum for Information Retrieval Evaluation*. ACM, December 2022.
- [86] T. Scialom, P.-A. Dray, S. Lamprier, B. Piwowarski, and J. Staiano. MLSUM: The multilingual summarization corpus. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8051–8067, Online, Nov. 2020. Association for Computational Linguistics.
- [87] A. See, P. J. Liu, and C. D. Manning. Get to the point: Summarization with pointer-generator networks. *CoRR*, abs/1704.04368, 2017.
- [88] A. Sinha, Z. Shen, Y. Song, H. Ma, D. Eide, B.-J. P. Hsu, and K. Wang. An overview of microsoft academic service (mas) and applications. In *Proceedings of the 24th International Conference on World Wide Web, WWW '15 Companion*, page 243–246, New York, NY, USA, 2015. Association for Computing Machinery.
- [89] O. Skibitska. The language of tourism: Translating terms in tourist texts. *Translation Journal*, 18(4), 2015.
- [90] J. Tan, X. Wan, and J. Xiao. From neural sentence summarization to headline generation: A coarse-to-fine approach. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 4109–4115, 2017.
- [91] Y. Tang, C. Tran, X. Li, P. Chen, N. Goyal, V. Chaudhary, J. Gu, and A. Fan. Multilingual translation with extensible multilingual pretraining and finetuning. *CoRR*, abs/2008.00401, 2020.
- [92] P. M. Tiersma. *Legal Language*. University of Chicago Press, Chicago, IL, 2 edition, June 2000.
- [93] A. Trischler, T. Wang, X. Yuan, J. Harris, A. Sordoni, P. Bachman, and K. Suleman. Newsqa: A machine comprehension dataset. *CoRR*, abs/1611.09830, 2016.
- [94] A. Urlana, N. Surange, P. Baswani, P. Ravva, and M. Shrivastava. Tesum: Human-generated abstractive summarization corpus for telugu. In *Proceedings of the Language Resources and Evaluation Conference*, pages 5712–5722, Marseille, France, June 2022. European Language Resources Association.

- [95] D. Varab and N. Schluter. DaNewsroom: A large-scale Danish summarisation dataset. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 6731–6739, Marseille, France, May 2020. European Language Resources Association.
- [96] D. Varab and N. Schluter. MassiveSumm: a very large-scale, very multilingual, news summarisation dataset. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10150–10161, Online and Punta Cana, Dominican Republic, Nov. 2021. Association for Computational Linguistics.
- [97] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *NIPS*, pages 5998–6008, 2017.
- [98] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. *CoRR*, abs/1804.07461, 2018.
- [99] C. Williams. *Tradition and Change in Legal English*. Peter Lang CH, Apr. 2005.
- [100] B. Workshop, :, and T. L. Scao et al. Bloom: A 176b-parameter open-access multilingual language model, 2022.
- [101] L. Xue, N. Constant, A. Roberts, M. Kale, R. Al-Rfou, A. Siddhant, A. Barua, and C. Raffel. mt5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, 2021.
- [102] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov, and Q. V. Le. Xlnet: Generalized autoregressive pretraining for language understanding. *arXiv:1906.08237*, 2019.
- [103] J. Zhang, Y. Zhao, M. Saleh, and P. Liu. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, pages 11328–11339. PMLR, 2020.
- [104] S. Zhang, S. Roller, N. Goyal, M. Artetxe, M. Chen, S. Chen, C. Dewan, M. Diab, X. Li, X. V. Lin, T. Mihaylov, M. Ott, S. Shleifer, K. Shuster, D. Simig, P. S. Koura, A. Sridhar, T. Wang, and L. Zettlemoyer. Opt: Open pre-trained transformer language models, 2022.
- [105] Y. Zhang, S. Sun, M. Galley, Y.-C. Chen, C. Brockett, X. Gao, J. Gao, J. Liu, and W. B. Dolan. Dialogpt: Large-scale generative pre-training for conversational response generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 270–278, 2020.
- [106] Z. Zhao and P. Chen. To adapt or to fine-tune: A case study on abstractive summarization. *arXiv preprint arXiv:2208.14559*, 2022.