Enriching Structured and Unstructured Knowledge for Low-Resource Languages by Cross Lingual Fact Extraction and Text Generation

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

Master of Science
in
Computer Science and Engineering
by Research

by

Bhavyajeet Singh
2018111022
bhavyajeet.singh@research.iiit.ac.in

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

International Institute of Information Technology
Hyderabad, India

CERTIFICATE

It is certified that the work contained in this thesis, titled **"Enriching Structured and Unstructured Knowledge for Low-Resource Languages by Cross Lingual Fact Extraction and Text Generation"** by **Bhavyajeet Singh**, has been carried out under my supervision and is not submitted elsewhere for a degree.

Date	Advisor: Prof. Vasudeva Varma
Date	Co-advisor: Dr. Manish Gupta

To Daddy Ji

Acknowledgments

The last few years have been an incredible and unforgettable part of my life and I am grateful to everyone who has contributed to it. This thesis, just like everything else that I have done at IIIT has been a collaborative effort with various individuals helping me in their own ways and even a thousand pages would not be sufficient to express my gratitude to all of you.

I would like to express my immense gratitude to my advisor Prof. Vasudeva Varma who has always been a great inspiration to me. Vasu sir always believed in me, provided me the support and guidance necessary to conduct this research and has influenced me in ways that go way beyond the lab. I have always admired you for your professionalism and perfection. You have always motivated me to be better in a truly holistic way and I can not thank you enough for that. I have learnt a lot from you and I am certain that those lessons would stick with me forever. And of course, I am extremely grateful to you for providing me the opportunity of working with the amazing peers at iREL.

I would also like to thank my co-advisor Dr. Manish Gupta for his continuous guidance and mentoring throughout the course of my research. Manish sir's immense meticulousness, brilliant intellect and dedication towards the field has always inspired me and I aspire to someday be able to incorporate some of those qualities myself too. Sir, I have learnt a lot of things from you, both academic and non-academic and I can not thank you enough for your patience, guidance and contribution. I could not have imagined looking up to a better set of advisors.

I would also like to thank all the amazing people at iREL. You all have been an important part of my life at IIIT. Firstly I would like to thank Himanshu, who has been a great mentor and one of the many reasons why I decided to join the lab. Himanshu has been an amazing friend and a constant source of support and 'wisdom'. You owe me a lot of treats. I would also like to thank Tushar, who was my first mentor at IRE and has mentored me in my research journey throughout. I feel extremely fortunate that I had the chance of working with you and extremely grateful for all the good things that came with it. I would also like to thank the entire XAlign team, Shivprasad Sir and Anubhav, you guys were great friends and amazing team mates. Thank you Aditya for being an extremely helpful and hardworking team mate, someone that I could always count on. Thank you Sagar for all the help, you are an amazing person and talking to you always brought a smile on my face. Thank you Pavan for giving me company through the various all-nighters at the lab. Thank you manav for your ada and

for being a 'resourceful' junior. I would also like to thank my other peers at the lab, Ankita, Dhaval, Shivansh, Tathagata, Tanmay, Rahul, Gokul, Harshit and all others. All our times together and outings have been fun.

I would like like to thank a lot of people beyond the lab who have shaped my life at IIIT. I would like to thank my entire wing group (NSMJ) for the constant love and companionship that I received from them. I would like to thank Jai, I was very fortunate to have an amazing room mate like you out of pure luck. Periwal, Ahish and Yoog you were my set of first friends at IIIT and I am glad I found you guys without wasting any time, could not have asked for anything else. Kirsur and Vikrant, it was great travelling around with you guys and I will always remember everything about that trip. Rogro, I will always cherish our Apex memories. Thank you Mahajan for always being a source of fun and happiness. Thank you Tado, you hosted this bunch of stinking, tired and extremely hungry guys at your place and I will always be grateful for that. Thank you Jaidev for being a great friend, and an amazing person . Kalp, Arpan, and Kashyap you guys were a constant source of support throughout our common Dulla struggles. Thank you all for always being there. Thank you Jivitesh for being a great friend, for always sharing, listening and motivating, can not thank enough for all our long conversations. All of you were amazing friends, peers, team mates and wing mates throughout my duration here at IIIT

I would also like to thank some of my great seniors who made everything a little bit easier. Prazwal sir, you have been an amazing person and I have always valued your company, thank you for being the fun, protective and helpful senior that you are. Thank you Akhil and Akshay for always providing a place to crash whenever I was tired of college. Thank you Nonidh for all the fun memories, I still have the broken specs frame and I look forward to watching more India-Pak matches with you. Thank you Ayush, Manoj and Mehtab for all the fun that we had.

Finally, I would like to thank the people outside IIIT for all their support. Thank you Vrinda for always being there, for always listening, and for being a constant source of strength and happiness. Thank you Mom and Dad for your unconditional support and love, I could not have done any of it without you and I can never thank you enough for all that you have done. Thank you for raising me the way you did and constantly showering your love on me. Thank you Dadi for all the home food that kept me going at IIIT, and Thank you to everyone in my family who always wished the best for me and supported me in every way possible.

Last but not least, I would like to thank all the people I could not name here for want of space but who were just as important to this journey.

Here's to hoping for a bright and happy future.

Abstract

Natural language generation has gained tremendous popularity in recent times primarily due to the advent of large pretrained language models trained on vast amount of data. However, most of this progress has only been limited to few high resource languages like English. Almost all low resource(LR) languages still suffer from the lack of sufficient training data and hence the lack of usable generative models. Furthermore, multiple business scenarios also require an automated generation of descriptive human-readable long text from structured input data, where the source is typically a high-resource language and the target is a low or medium resource language.

In this work, we present systems and approaches which can be utilised to ultimately enrich the structured and unstructured content available for low resource languages in the encyclopedic domain. In order to do so, we introduce cross lingual techniques which efficiently utilise the abundant structured data available in high resource languages. We also introduce systems to further enrich this structured data using the information present in the form of natural language text in low resource languages.

Firstly we propose novel problem of cross lingual fact to text alignment in order to construct the XAlign dataset for the purpose of cross lingual fact to text generation and fact extraction. We explore several methods to automatically align English facts from Wikidata to sentences from native language Wikipedia. We experiment with approaches accounting for syntactic and semantic matches between the fact and the sentence and propose a two stage pipeline for automated alignment and evaluate it on a manually annotated high quality test set. We also experiment with distant supervision and transfer learning based techniques in order to achieve quality alignment. We use the best approach to create the XAlign dataset which consists of more than half a million aligned (sentence, facts) pairs across 12 Indian languages.

Following the construction of the dataset we propose the problem of Cross Lingual Fact Extraction (CLFE). Recent approaches concentrate on automatically enriching large knowledge graphs like Wikidata and DBPedia from text. However a lot of information present in the form of natural text in low resource languages is often missed out. Furthermore, considering the potential use case of utilising structured data for generating content in various LR languages, the CLFE task aims at extracting factual information in the form of English triples from LR Indian Language text. Despite its massive potential, progress made on this task is lagging

when compared to Monolingual Information Extraction. We propose strong baselines and an end-to-end generative approach for the CLFE task which achieves an overall F1 score of 77.46.

We then introduce and explore the problem of cross lingual fact to text generation (XF2T). We extensively explore multiple approaches for the task and analyse different components of the pipeline. Starting from the choice of pretrained transformer model used, we explore the impact of different continued pretraining strategies. We also show that building cross lingual systems results in better performance than translation based approaches or multiple bi-lingual modes, thus validating the necessity of the proposed problem. We introduce novel techniques like fact-aware embeddings to further improve the generation quality. We demonstrate that these methods produce coherent and precise sentences.

Our experiments with the XF2T task lead to the observation that these generative models suffer from hallucination and due to the training setup, are only limited to generating a single sentence at a time. In order to mitigate these limitations, we extend the XF2T task to the problem of Cross-Lingual Fact to Long Text Generation (XFLT). The task involves generating descriptive and human readable long text in a target language from structured input data (such as fact triples) in a source language. XFLT is challenging because of (a) hallucinatory nature of the state-of-the-art NLG models, (b) lack of good quality training data, and (c) lack of a suitable cross-lingual NLG metric. Unfortunately previous work focuses on different related problem settings like monolingual graph to text and has made no specific efforts to handle hallucinations. Hence, we propose a novel solution to the XFLT task which addresses these challenges by training multilingual Transformer-based encoder-decoder models with coverage prompts and grounded decoding. Further, it improves on the XFLT quality by defining task-specific reward functions and training on them using reinforcement learning. On a dataset with over 64,000 paragraphs across 12 different languages, we compare this novel solution with several strong baselines using a new metric, cross-lingual PARENT.

Overall, we work on multiple related tasks aimed at automating the generation of encyclopedic articles and consolidating the factual information available in the form of natural language text from multiple LR languages to enrich structured knowledge bases.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this thesis, we primarily look at exploring ways of enriching the available structured and unstructured information over Wikidata and Wikipedia respectively. This chapter presents an introduction to the task at hand by first discussing the motivation behind it, followed by an overview of the subtasks. The chapter concludes by outlining the key contributions made in this thesis and providing an overview of the content flow across the subsequent chapters.

1.1 Motivation

In today's rapidly evolving technological landscape, the field of natural language generation has gained significant attention and importance. The ability to generate automated natural language text and extract factual information from it has become crucial for numerous applications in diverse domains. However, there are several key challenges and motivations driving the need for further exploration and advancements in this field. This section aims to highlight the underlying motivations that propel this thesis and shed light on the significance of addressing these challenges.

1.1.1 Information and resource divide among languages

The web exhibits a significant disparity in the availability of resources across different languages. This resource divide poses challenges in various domains, including education, technology, and access to information. This divide is also apparent in the realm of online knowledge repositories, with Wikipedia serving as a prominent example. Wikipedia is one of the largest online repositories of knowledge. While Wikipedia provides a wealth of information in widely spoken languages such as English, major gaps exist in the coverage of less widely spoken languages. This information divide results in limited access to knowledge and cultural representation for speakers of low-resource languages. It hinders their ability to contribute, access accurate information, and participate fully in the digital world. Bridging the information

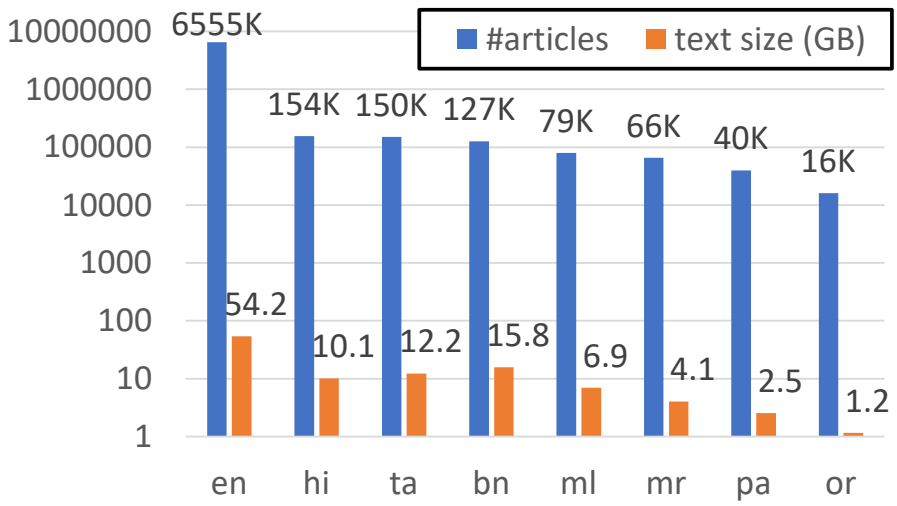

Figure 1.1: Comparison of Number of Wikipedia articles and text size between English and 5 Indian languages (from the September 2022 Wikipedia dump.)

divide across languages is essential to ensure equal access to knowledge and promote linguistic diversity in the online landscape. Figure 1.1 highlights the gap in the size of English Wikipedia and that of a few Indian languages.

1.1.2 Relying on structured factual data

Structured factual data serves as a reliable foundation for generating contextually appropriate and informative text in applications like automated content creation and summarization. By incorporating structured data, text output can be tailored to specific domains, enriched with relevant facts, and aligned with underlying information. This ensures the production of high-quality, reliable, and informative text that meets diverse user expectations. Additionally, structured data enables cross-lingual text generation, facilitating effective communication and information dissemination in multiple languages.

Furthermore, knowledge graphs serve as valuable resources for fact-checking and verification. During the generation of encyclopedic articles, the structured data can be leveraged to validate the accuracy of the generated text. By comparing the information against trusted sources within the knowledge graph, we can ensure that the articles align with established facts, thus minimizing the risk of hallucinations or the propagation of misinformation.

By using knowledge graphs and structured data in the generation of encyclopedic articles, we can address the challenges posed by large language models' potential for hallucination and inaccuracies. Leveraging structured information promotes reliability, fact-checking, coherence, and comprehensiveness, offering a robust framework for generating authoritative and trustworthy encyclopedic content.

For these reasons we make efforts to enrich the availability of structured data over Wikidata for entities which have their information present in multiple low resource languages and utilise the information present in the form of structured data to generate natural language articles.

1.1.3 The need for cross lingual automated generation

The first sub sections highlights the need for generating articles in native Indian languages and the second sub section provides the motivation for using structured data as a possible source of information for doing so. In this subsection, we explain the reasons for relying on cross lingual generation.

The two possible alternatives to a cross lingual generation could be to either use translations of articles from high resource languages or use monolingual data to text approaches. Using translation from high resource languages results in a dependency on the availability of content in those high resource languages. This way, generating articles for entities which may be notable for the readers of a particular low resource language but not notable universally would be extremely difficult. Furthermore, translations is likely to add loss of information and erroneous generations.

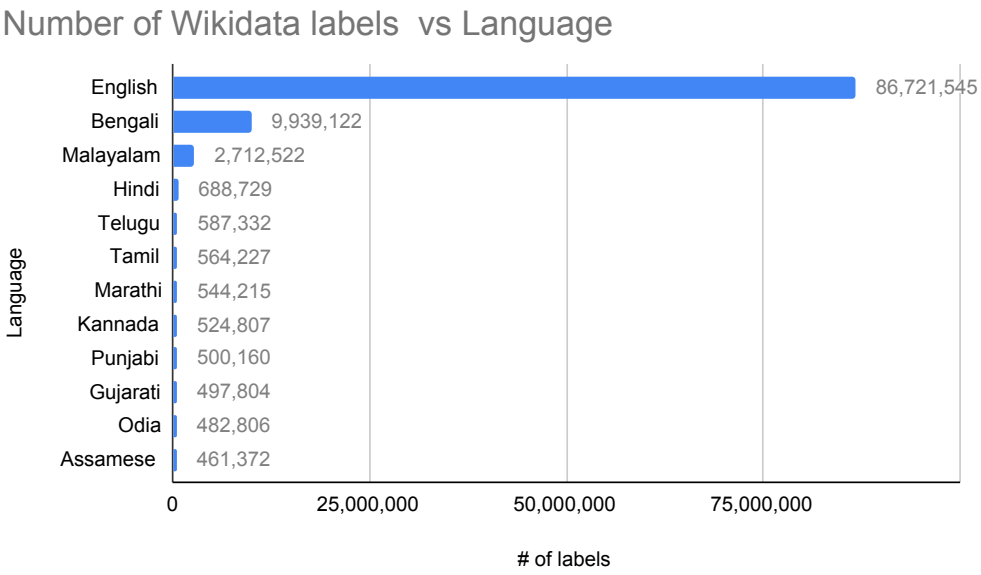

Figure 1.2: Comparison of Number of Wikidata labels between English and 11 Indian languages

The other alternative of multiple monolingual data to text systems could become a good choice, however the availability of this structured data in the low resource languages is in itself a problem. Figure 1.2 shows the discrepancy in the amount of labels available over Wikidata

in English vs that in Indian languages. Thus we choose to utilise these English labels in order to cross lingually generate articles for the multiple Indian languages.

1.2 Problem Description

This section describes the various components of the problem and provides a brief overview for the same.

1.2.1 Cross-lingual fact extraction

Large-scale knowledge graphs, such as the widely recognized Wikidata, have emerged as comprehensive repositories striving to encapsulate global knowledge encompassing an extensive array of entities. Recent endeavors have been dedicated to the augmentation of these knowledge graphs through automated techniques that leverage textual sources. Nevertheless, it is worth highlighting that a substantial wealth of valuable information, present in the form of text from low resource languages, often remains overlooked and underutilized. To overcome this limitation, the field of cross-lingual information extraction endeavors to extract factual information, specifically in the form of English triples, from textual sources composed in low resource Indian languages. However, despite the promising potential of this cross-lingual extraction task, progress in this domain lags behind its monolingual counterpart, which predominantly caters to a few high resource languages. Consequently, the overarching objective of cross-lingual fact extraction extends beyond individual languages, aiming to systematically extract and consolidate factual information, harnessed from natural language text across multiple low resource languages, into a unified language-agnostic knowledge graph.

1.2.2 Cross-lingual fact-to-text generation

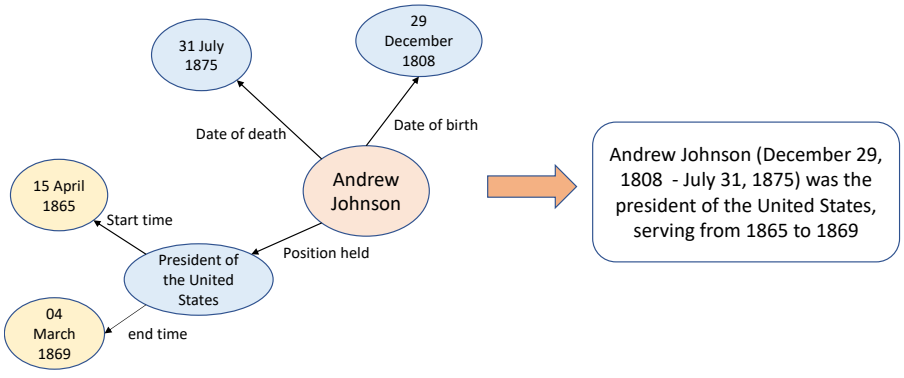

Figure 1.3: An example of the Fact to text task

The task of fact-to-text (F2T) generation [64] is centered around transforming structured data into natural language. F2T generation systems are indispensable in numerous downstream applications within Natural Language Processing (NLP), including automated dialogue systems [78], domain-specific chatbots [54], open domain question answering[83], and the creation of sports reports [33, 10], among others. Figure 1.3 provides an example of the fact to text task. However, most of such data to text systems are only available for English and not for low-resource (LR) languages. The problem of cross-lingual F2T generation (XF2T) deals with the setting where the factual input is in one language and the output is in a different language. In our case, the input is a set of English facts and output is a sentence capturing the fact-semantics in the specified LR language. Note that a fact is a triple composed of subject, relation and object. These triples can further have more sub-property information, called qualifiers.

In addition to the traditional cross-lingual data-to-text (XF2T) task, we embark on the ambitious endeavor of generating longer text pieces, encompassing entire articles, in a single automated process. This extended task tackles the challenge of generating multiple sentences that are coherent and follow a natural language text order, leveraging all available factual information about a given entity in the English language. By automating the complete pipeline of article generation, this problem necessitates additional steps, including organizing the input facts and devising a content plan. It is noteworthy that even state-of-the-art large language models exhibit limitations when it comes to handling longer outputs, as their performance tends to deteriorate with increasing length of the generated text. Therefore, addressing these challenges in the context of generating comprehensive and lengthy articles via cross lingual fact to long text generation (XFLT) requires innovative approaches and novel techniques.

1.2.3 Aligning structured data and natural language text across languages

The task of cross lingual fact to text generation, and the task of cross lingual fact extraction both depend on the availability of a structured dataset which is well-aligned with semantically equivalent textual data. Generating a high-quality fact-to-text (F2T) dataset of sufficient scale through manual creation is a daunting task that requires human supervision. To address this challenge, various automatic alignment approaches have been proposed. These approaches involve techniques such as aligning Wikipedia sentences with Infoboxes [42], utilizing distant supervision [22], and identifying lexical overlap between textual and structural entities [32], [26]. However, it is worth noting that the majority of existing F2T datasets are currently available only in English. For low resource (LR) languages, the number of structured Wikidata entries for person entities is significantly limited compared to English, resulting in a scarcity of data. Moreover, LR languages tend to have a smaller average number of facts per entity compared to English. As a result, the development of monolingual F2T datasets for LR languages faces challenges due to data sparsity. Thus, we propose transfer learning and distant supervision

based methods for cross-lingual alignment in order to create a high quality cross lingual dataset which can be used for cross lingual fact extraction and fact-to-text generation.

1.3 Challenges

Cross-lingual data-to-text generation poses several challenges that need to be overcome to ensure effective and accurate results. One of the primary challenges is the scarcity of parallel data, especially for low-resource languages. Building large-scale, high-quality datasets that align factual information with corresponding text in different languages is a laborious and time-consuming task. Another challenge lies in the structural and linguistic variations across languages. Each language has its own syntax, grammar rules, and semantic nuances, making it difficult to directly transfer information from one language to another. Additionally, the lack of comprehensive resources and tools for cross-lingual processing further complicates the generation process.

These complications are further amplified when we delve into the domain of encyclopedic text generation. Wikipedia is an encyclopedia and has specific guidelines and writing styles which are not the same as any other text generation. These nuances have to be carefully considered while creating any automated tool for the platform. Wikipedia articles have a greater need to be factually correct and grounded, as compared to any other generation and this is one of the major problems in existing large language models.

1.4 Contributions

The primary contributions of this thesis are as follows :

1. We propose cross-lingual approaches of text generation and information extraction as a possible solution to the scarcity of resources in low-resource languages. Our focus lies in tackling the alignment of low resource language text and structured data specifically in the context of encyclopedic text thus leading to the creation of the XAlign dataset.

2. We highlight the importance of generation grounded on structured data and the need for accumulating the information present in natural language text from multiple languages in the form of unified structured knowledge graphs. For this purpose, we propose the task of cross lingual fact extraction and explore novel approaches for the same.

3. We explore the task of generating sentences in low resource languages using structured data thus introducing the task of cross lingual fact to text generation (XF2T) and proposing novel approaches and strong baselines for the same.

4. We further extend the task of generating sentences to generating longer pieces of text with specific focus on reducing hallucination by utilising reinforcement learning based techniques. In order to accomplish this task, we also create the XLAlign dataset and the develop reliable metrics tailored to the cross lingual fact to long text (XFLT) task using partially aligned data.

These will be elaborated further in subsequent details. In addition to our main contributions, we provide an account of the experiments conducted that did not yield successful results. We believe that sharing these negative experiments can also provide valuable insights into the modeling approaches used in this particular domain.

1.5 Thesis Organisation

The thesis is structured into seven chapters, and a brief overview of each chapter is provided as follows:

1. Chapter 1 (Introduction) presents the motivation behind the work done as a part of the thesis and discusses the sub tasks explored. In this chapter, we introduce the problem statement and provide a brief summary of the major contributions of the thesis.

2. Chapter 2 (Related work) presents a survey of the prior literature related to the tasks explored in this thesis.

3. Chapter 3 focuses primarily on the creation of the XALIGN dataset by exploring techniques to align natural language text from various low resource languages with structured factual data.

4. Chapter 4 introduces the task of cross lingual fact extraction in order to consolidate factual knowledge present across various languages in the form of knowledge graphs

5. Chapter 5 explores the task of cross lingually generating sentences using structured data (XF2T). It defines the methods tried for the task and also explaines the baselines used to compare their performace.

6. Chapter 6 extends the generation towards longer text with specific focus on generating grounded text and tackling the problems of hallucination arising due to partially aligned data.

7. Chapter 7 serves as the concluding chapter of the thesis, offering a summary of the covered work and exploring potential avenues for future expansion and development based on the findings.

In addition, Appendix A provides a succinct summary of further research undertaken during the course of this thesis. This supplementary section highlights the effectiveness of pretrained transformer models, which hold a significant position in this thesis, in diverse tasks.

Chapter 2

Related work

The related works chapter delves into the existing literature and research in the field of natural language generation, with a specific focus on the interplay of structured data and text generation in various contexts. This chapter provides a comprehensive review of studies, methodologies, and techniques that have contributed to aligning sentences with their corresponding facts, extracting factual information from text, and automatically generating natural language sentences. By examining the advancements made in these areas, we aim to build upon the existing knowledge and identify gaps that our thesis aims to address. Through this exploration of related works, we gain valuable insights into the state-of-the-art approaches and identify potential avenues for further research and innovation.

2.1 Fact to text datasets

Training F2T or Information Extraction models requires aligned data with adequate content overlap. In recent times, significant efforts have been dedicated to the creation of automated datasets that convert structured data into text in diverse domains. Some previous studies like WebNLG [27] collected aligned data by crowd-sourcing while others have performed automatic alignment by heuristics like TF-IDF. The WebNLG dataset encompasses 15 distinct categories. They implemented a content selection module to extract fact triples of varying relevance, coherence, and relation from DBpedia. In each category, a graph was constructed by utilizing 500 seed entities and exploring edges up to 5 hops away from them. The graph was then used to train bi-gram models, incorporating different triple relations. Ultimately, the process of content selection was formulated as a linear programming problem, aiming to choose a sub-tree within the category graph for a given entity, maximizing the bi-gram probability while accommodating varying fact sizes. After the content selection for a given entity is done, the author uses crowd-source annotators to verbalise fact triples into a sentence.

Considerable endeavors have been dedicated to the development of automated datasets, like Lebret et al. [42] developed the WikiBio dataset, which aligns opening sentences with infoboxes

found in English Wikipedia articles about individuals. This approach has been extended to generate datasets in different domains [59] and languages [53] by aligning Wikipedia text with infoboxes. Some previous works have also proposed aligning knowledge graph triples from Wikidata with opening sentences in Wikipedia [26] in order to generate domain independent datasets. They achieve this alignment by performing a match over the named entities in a sentence and those in the corresponding Wikidata triples. However these approaches can only work when the source triplets and the text to be aligned are in the same language. Additionally, a dataset has been introduced that incorporates sub-property information in the form of quadruples, expanding beyond the use of Wikidata triples alone [49]. To facilitate alignment between structured data and natural text across various domains, a sequential pipeline strategy involving data collection, data filtering, entity linking, and alignment has been proposed [22, 32]. Some of these dataset creation pipelines also incorporate manual annotation in order to create the test set or in some cases, the entire dataset. Table 2.1 shows basic statistics of popular F2T datasets. The XAlign dataset created as a part of this work is the only cross lingual fact to text dataset with 12 languages and more than half a million samples.

Dataset	Languages	A/M	|I|	F/I	|P|	|T|	X-Lingual
WikiBio	en	A	728K	19.70	1740	26.1	No
E2E	en	M	50K	5.43	945	20.1	No
WebNLG 2017	en	M	25K	2.95	373	22.7	No
fr-de Bio	fr, de	A	170K, 50K	8.60, 12.6	1331, 1267	29.5, 26.4	No
TREX	en	A	6.4M	1.77	642	79.8	No
WebNLG 2020	en, ru	M	40K, 17K	2.68, 2.55	372, 226	23.7	Yes
KELM	en	A	8M	2.02	663	21.2	No
WITA	en	A	55K	3.00	640	18.8	No
WikiTableT	en	A	1.5M	51.90	3K	115.9	No
GenWiki	en	A	1.3M	1.95	290	21.5	No
XAlign	en + 7 LR	A	0.45M	2.02	367	19.8	Yes
XAlignV2	en + 11 LR	A	0.55M	1.98	374	19.7	Yes

Table 2.1: Statistics of popular Fact-to-Text datasets: WikiBio [42], E2E [54], WebNLG 2017 [27], WebNLG 2020 [24], fr-de Bio [53], KELM [2], WITA [26], WikiTableT [11], GenWiki [32], TREX [22], XAlign [1], and XAlignV2 (ours). Alignment method could be A (automatic) or M (manual). |I|=number of instances. F/I=number of facts per instance. |P|=number of unique relations. |T|=average number of tokens per instance.

2.2 Cross lingual fact extraction

Considerable progress has been made in addressing the challenge of extracting structured information from unstructured text. T-REx [22], for example, employs entity linking, co-reference resolution, and string matching techniques to establish connections between facts found in DBPedia [43] abstracts and Wikidata [76] triples. On the other hand, REFCOG [36] operates in a cross-lingual context and surpasses the performance of pipeline-based approaches. However, it should be noted that these approaches have their limitations as they primarily focus on fact linking and require a predefined set of facts as input.

To address this challenge, OpenIE [3] utilizes the linguistic structure to facilitate information extraction in open domains. In contrast to previous open domain information extraction systems such as Ollie [50], which rely on a large set of patterns to extract facts comprehensively, OpenIE employs a smaller set of patterns that are specifically designed for canonically structured sentences. This approach proves effective in extracting relevant information. However, it should be noted that the facts generated by these open domain information extractors often exhibit excessively long and overly specific relations, rendering them unsuitable for constructing knowledge graphs.

Certain applications like [87] [72] address the challenge of information extraction by employing neural models that simultaneously extract entities and their relationships from input text, without relying on preexisting repositories of facts. These approaches have demonstrated the ability to extract open information from text, as evidenced by their performance on the WebNLG dataset. However, it is important to note that these models are typically designed for monolingual settings and are thus restricted to extracting knowledge from text in a single language. Furthermore, many existing relation extraction models heavily rely on exact entity matches in the source text, which poses difficulties when adapting them for the cross-lingual fact extraction task.

Cross-lingual fact extraction, which involves extracting facts from source text written in different languages, has not received the same level of attention as monolingual fact extraction. While previous work, such as the study conducted by Zhang et al [84], focused on this task within a single language, the reported highest F1 score reached only 33.67. Furthermore, fact extraction from low resource languages, particularly Indic Languages, has not been explored extensively. To address these gaps in information extraction, our work aims to bridge the divide by proposing systems specifically designed for cross-lingual subject-centric fact extraction in low resource Indic Languages.

2.3 Fact to text generation

Recently there has been a lot of work on cross-lingual NLG tasks like machine translation [13, 48], question generation [14], news title generation [47], and summarization [88] thanks to models like XNLG [14], mBART [48], mT5 [82], etc. Initial F2T methods were template-based and were therefore proposed on domain-specific data like medical [7], cooking [15], person [21], etc. They align entities in RDF triples with entities mentioned in sentences, extract templates from the aligned sentences, and use templates to generate sentences given facts for new entities. Template-based methods are brittle and do not generalize well.

In recent times, Seq-2-seq neural methods [42, 51] have also become popular for F2T. These include vanilla LSTMs [75], LSTM encoder-decoder model with copy mechanism [70], LSTMs with hierarchical attentive encoder [53], pretrained Transformer based models [65] like BART [44] and T5 [61]. Vougiouklis et al. [75] proposed a method which uses feedforward neural networks to encode RDF triples and concatenate them as the input of the LSTM decoder. Variations of LSTM encoder-decoder model with copy mechanism [70] or with hierarchical attentive encoder [53] have also been proposed. Pretrained Transformer based models like BART [44] and T5 [61] have been applied for mono-lingual English Fact-to-Text [65]. Richer encoding of the input triples has also been investigated using a combination of graph convolutional networks and Transformers [86], triple hierarchical attention networks [12], or Transformers with special fact-aware input embeddings [12]. Some recent work also explores specific F2T settings like plan generation when the order of occurrence of facts in text is available [86]. Like our work, some studies [11, 58, 80] also perform fact to long text generation. However, all of these methods focus on English F2T only.

Another line of work which is of great interest to us, specifically focuses on reducing hallucinations in natural language generation. A recent publication [26] works on the specific setting of partially aligned F2T when the text covers more facts than those mentioned in the input. They incorporate special techniques to deal with partially aligned data and reduce hallucination during generation. This is closely related to our work since our training data is also partially aligned and not all the information present in a sentence may be present in the aligned input facts. Tian et al. [74] also propose confident decoding in order to achieve more faithful fact to text generation. They postulate that hallucination can stem from an encoder-decoder model generating content phrases without attending to the source. As a remedy, they propose the inclusion of a confidence score to ensure that the model focuses on the source when necessary. Furthermore, they introduce a variational Bayes training framework, enabling the model to learn the score from the provided data. Lai et al. [41] explore rewarding pretrained models with custom reward functions in order to achieve improved formality style transfer, similarly reward based approaches can be explored to generate more grounded text as well.

Our work is most related to fact verbalization tasks [52, 28] where the focus is to use facts to generate short text. Gardent et al. [28] proposed the WebNLG dataset which contains data

for English and Russian where each instance has 2.6 facts per instance and 23.7 words in the output text on average. Ferreira et al. [25] further enriched the corpus to include German as well. Moussallem et al. [52] verbalize RDF data to German, Russian, and English using the enriched WebNLG data, and experiment with an encoder-decoder architecture.

As against these, we also propose Cross lingual Fact to Long Tect (XFLT) where the focus is on *long* text generation in a cross-lingual manner. Further, from a knowledge graph (KG) and text linking perspective, our work is related to tasks like entity linking (link mention in a sentence to a KG entity) [8] and fact linking (linking sentence to a set of facts) [37]. As against this, XFLT is the problem of generating a paragraph given a set of facts.

2.4 Text generation metrics and evaluation

Sai et al. [67] provide a survey of evaluation metrics used for NLG systems. The most common metrics for tasks like F2T come from the class of reference dependent metrics. Metrics like BLEU [55], METEOR [6] and chrF++ [57] depend entirely on the reference text and evaluate the generations based on their overlap with the provided reference. The BLUE metric is a precision-oriented measurement that calculates the degree of overlap between the reference and the hypothesis based on n-grams. Specifically, it quantifies the ratio of n-grams that are shared between the two texts to the overall count of n-grams present in the hypothesis. Another source dependent metric METEOR [6] highlights that the problems with exact word match and aims to mitigate it by also using a match with potential synonyms, however extending this to all languages is not possible since the wordnets for low resource languages may not be available. Chrf is a metric that operates at the character level. It calculates precision and recall based on the character n-grams for different n values (up to 6). These precision and recall scores are then combined using arithmetic averaging to obtain the overall precision and recall respectively. Additionally, chrF++ extends the analysis to include word unigrams and bigrams in addition to character n-grams.

Another set of metrics for generation utilise embeddings from transformer based models in order to compute similarity with the reference text. This mitigates the problems of exact word match and rewards semantic similarities between the reference text and the predicted output. BERTScore [85] is one such metric which omputes cosine similarity of each hypothesis token with each token in the reference sentence using contextualized embeddings. We also utilise a modification of this using the LABSE [23] embeddings for evaluation some of our models.

2.4.1 Source-Dependent Text Generation Metrics

Evaluation metrics for text generation like BLEU and ROUGE rely on the reference text. This is problematic when the reference and the source do not align entirely. Datasets for fact

to text tasks are partially aligned, i.e., the reference text may have extra information not specifically mentioned in the input text. Hence, a source-dependent metric is suitable for fact to text tasks. Dhingra et al. [20] proposed PARENT as an NLG source-dependent metric that aligns n-grams from the reference and generated texts to the input text before computing their precision and recall. They show that PARENT correlates with human judgments better than other text generation metrics like BLEU, ROUGE, METEOR, CIDEr and CIDErD. However, PARENT works for monolingual tasks only since it relies on string matching. XFLT involves cross-lingual modeling and hence needs an adaptation of the PARENT metric for cross-lingual scenario. Hence, we propose XPARENT, which is a modified version of PARENT adapted for cross-lingual settings.

Chapter 3

Constructing the XAlign dataset

3.1 Overview

This chapter provides a detailed description of the construction pipeline for the XAlign dataset. As highlighted in section 1.1, we construct a cross lingual dataset for the task of fact to text generation and information extraction. We start by first constructing a dataset for 8 languages and then extend in to 4 more languages. Thus, we collect the English triples from Wikidata and natural language sentences in 12 language from their corresponding Wikipedia. We align individual sentences to the triples which express the same factual information as the sentence. This chapter, describe in detail the pipeline for data collection, pre-processing and alignment followed by an analysis of the constructed dataset.

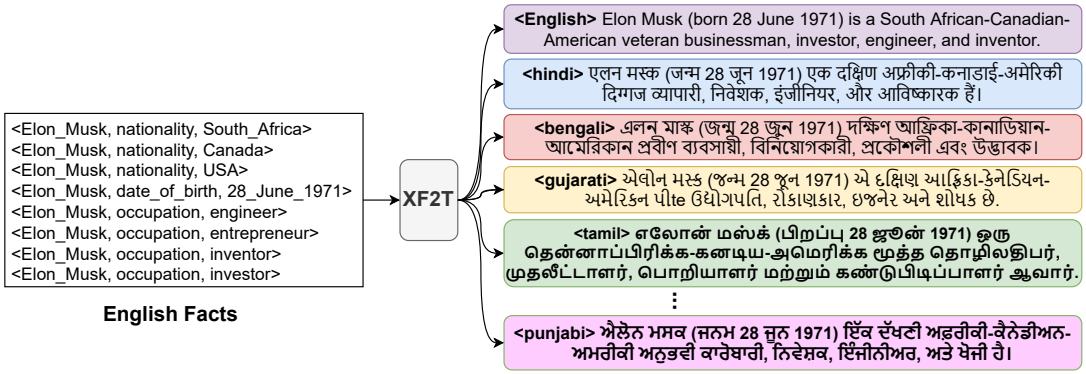

Figure 3.1: Examples of aligned English facts and LR natural language sentences

Figre 3.1 Provides an example of sentences from different languages aligned to a set of English facts.

Lang.	WikiData entries	Facts	Average facts per entity	Wikipedia articles
hi	26.0K	271.0K	10.43	22.9K
mr	16.5K	174.0K	10.56	15.9K
te	12.4K	142.2K	11.49	7.8K
ta	26.0K	280.4K	10.77	25.6K
en	1.3M	30.2M	22.8	627.9K
gu	3.5K	37.8K	10.88	1.9K
bn	36.2K	501.9K	13.87	29.0K
kn	7.5K	83.6K	11.1	4.5K

Table 3.1: Statistics of Wikidata and Wikipedia for the person entities across 8 languages

3.2 Data collection and pre-processing

3.2.1 Processing Wikidata facts

We obtain the facts corresponding to the chosen entities from the December 2020 Wikidata dump. For each entities we obtain all the facts where the entity appears as either the subject or the object. This also results in the presence of redundant facts due to the existence of backward relations. For example for the entity 'Amitabh Bachchan' the facts < Amitabh Bachchan, Son, Abhishek Bachchan > and <Abhishek Bachchan, Father, Amitabh Bachan> represent the same information. To tackle this redundancy, we use the WikiData API [1] to filter out such backward relations if the corresponding forward relation is already present in the list of the facts.

Furthermore, we filter out relations which do not provide information that can be verbalised into natural language sentences. These include relations like unique resource ids such as WikibaseItem, Time etc and social media URLs. We also specifically filter out relations based on our manual analysis if it is observed that they bring redundancy. For example we filter out relations like "native language", "writing language", "language known" etc if the relation "language written, spoken or signed" is also present and has the same object. This is done because these relations lead to the same verbalisation. Similarly if 'country of citizenship' and 'country of sport' had the same object for an entity, only 'country of citizenship' was retained. A few other specific redundancies related to the relation 'occupation' are also removed. The relation 'gender' is also removed for the purpose of alignment since that particular relation is never explicitly verbalised but expressed in certain languages through linguistic variations of pronouns and verbs.

In cases where there is supplementary supporting information linked to a given fact triple, we preserve it as a fact qualifier. As a result, we have successfully extracted approximately 1 million facts for approximately 120 thousand entities across all languages in our dataset.

[1] https://query.wikidata.org/

3.2.2 Processing Wikipedia Sentences

For all entities belonging to the person's domain, we extract the local language Wikipedia of 12 different languages in order to obtain sentences belonging to a given entity which can later be aligned with the extracted English facts from Wikidata. We use the May 2021 Wikipedia XML dump to obtain the local language Wikipedia articles for our entities of interest. We use the Wikiextractor tool [4], in order to extract clean text from the Wikipedia article. This tool automatically removes figures, tables, references and URLs from the XML version of the Wikipedia article.

Once we have the clean text, the next task is to tokenise the text into individual sentences. In order to do so, we use the sentence tokenizers from the IndicNLP library [40]. However to further process the sentences, we apply extra heuristics to account for special punctuation and sentence delimiters used in Indian languages. After tokenisation, the next step is to filter out sentences which may not contain any factual information or may adversely affect the quality of training data. To begin with, we remove any sentence with less than 5 or more than 100 words. For some of the low resource languages, it is common for the article to contain sentences of words from some other language. Such sentences are filtered out by applying a threshold over the results obtained from the Polyglot language detector[2]. If the confidence score for the detected language for any sentence is less than 95%, then the particular sentence is removed.

We also aim to remove sentences which do not contain any potential factual information. To filter out sentences that lack factual information, we employ part-of-speech (POS) tagging and retain only those sentences that contain at least one noun or verb. For POS tagging, we utilize different tools such as Stanza[60] for English, Hindi, Malayalam, Telugu, Tamil, Marathi, and Punjabi; LDC Bengali POS Tagger[5] for Bengali; and Patel et al's tool[56] for Gujarati. For the rest of the languages, no off the shelf POS tagging tools were available. To ensure a comprehensive compilation of entities in these languages, we construct a backup list by tracking Wikipedia articles that either cite or are cited by other pages. This combined list serves as a robust inventory of entities in the target language. Furthermore, we incorporate the native language labels of these entities from WikiData. The sentences can also represent factual information without containing any noun by using pronouns. To account for sentences containing pronouns, which may be overlooked by the previous list, we manually generate a set of pronouns for the target language. If there is an overlap between the entity pronoun list and the words present in a provided phrase, we retain the corresponding sentences.

Additionally, we also retain the section information associated with each sentence and the relative position of the sentence in the original article, extracted from the respective Wikipedia URL. This information would also be useful in the later sections of this thesis. All the sentences are finally processed using a script normaliser[3] since the scripts for Indian languages might have

[2]https://polyglot.readthedocs.io/en/latest/Detection.html

[3]https://indic-nlp-library.readthedocs.io/en/latest/indicnlp.normalize.html

discrepancies which may affect the performance of the model trained on such data. All sentences are then translated to English using the IndicTrans translator [63] which was trained on the Samanantar[63] data. The translations may be useful for the annotator during the annotation process and might be used in some of the approaches tried.

3.3 Manual annotation for test set

In order to construct a high quality test set, we get the sentences manually annotated from the native speakers of each of the languages. For this purpose, we construct an annotation tool for the manual annotators, in order to make the annotation process simpler.

The annotators were chosen from the list of annotators compiled by the National Translation Mission [4]. All annotators were required to have education equivalent to graduation and fluency in their native language and English. Each annotator was first given a set of 100 questions as a test and the annotators who perform above the expected threshold are chosen for further annotations.

Lang	κ	|A|	|I|	avg/min/max |T|	avg/min/max |F|
hi	0.81	4	842	11.1/5/24	2.1/1/5
mr	0.61	4	736	12.7/6/40	2.1/1/8
te	0.56	2	734	9.7/5/30	2.2/1/6
ta	0.76	2	656	9.5/5/24	1.9/1/8
en	0.74	4	470	17.5/8/61	2.7/1/7
gu	0.50	3	530	12.7/6/31	2.1/1/6
bn	0.64	2	792	8.7/5/24	1.6/1/5
kn	0.54	4	642	10.4/6/45	2.2/1/7

Table 3.2: Annotation statistics of test data for XAlign. |A|=#Annotators, |I|=#instances, |T|=word count, |F|=fact count, =avg Kappa score

Once the annotations were complete, only the sentences where complete information of the sentence was presented in the facts were retained as a part of the test set. Partially covered sentences, or those with no factual match are discarded. Table 3.2 provides the annotation statistics.

Now we describe the details of the annotation tool and the instructions provided to the annotators in order to construct the test set.

[4] https://www.ntm.org.in/

3.3.1 Annotation tool

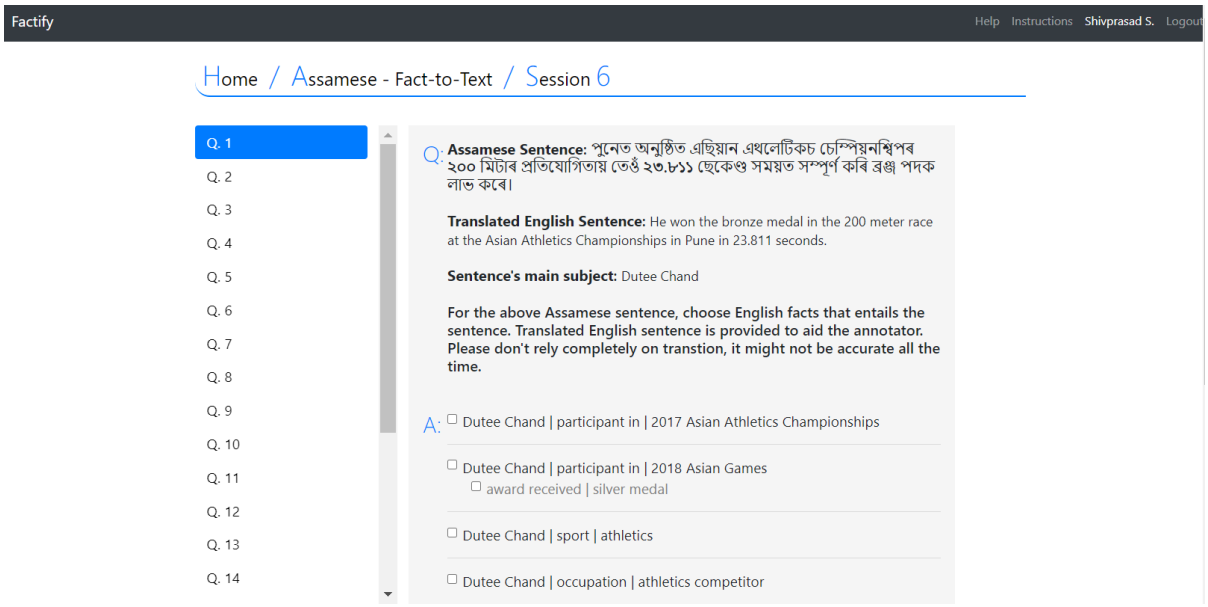

Figure 3.2: A screenshot of the annotation tool depicting a sample to be annotated with the native language sentence, translated sentence and the facts associated with the entity .

An annotation tool was created using the React framework[5] to aid the process of annotation. The tool presents the users with sentences about an entity from the native language Wikipedia, its English translation and the facts associated with that entity. The user was asked to mark for each of the facts whether the information present in the fact was represented in the sentence or not. The annotators were also asked an additional question after marking the facts that whether the chosen facts represent the complete information in the sentence or not, in other words, whether the sentence is completely or partially covered by the chosen facts. The annotators could view the annotation instructions and track their progress on the website.

3.3.2 Instructions

A detailed set of guidelines were compiled for the annotators to assist them in the process of annotation. These included general instructions and examples of specific cases which might be confusing for the annotators. Some of the provided instructions were as follows :

- Do not mark the fact as entailed if the fact contradicts the information present in the sentence. For example if the sentence talks about the date of birth, and there is a fact mentioning the date of birth of the person, do not mark the fact if the dates mentioned in the fact and the sentence do not match

[5]https://react.dev/

- Do not mark a fact if any amount of world knowledge is needed to infer the information present in the fact from the sentence. For example if the sentence mentions the place of birth as Ahmedabad, one should not mark a fact mentioning the place of birth as Gujrat even though it is correct.
- If there are two facts presenting the same information, choose the more appropriate one
- There were also specific instructions provided regarding how to deal with abbreviations or facts related to certain specific relations.

All these instructions were provided in much more detail with examples and an explanation video which were also linked in the annotation tool.

3.4 Aligning facts and sentences

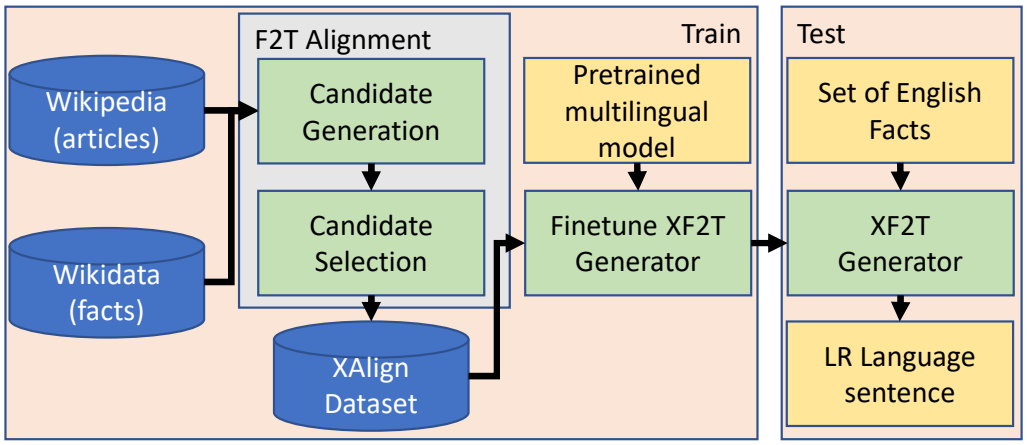

Figure 3.3: XAlign F2T Alignment System Architecture

This section describes the detailed process of automatically aligning the Wikidata facts with the Wikipedia sentences corresponding to a given entity. The proposed pipeline for this uses a two phase architecture. Here the first phase is to build a maximum recall system where all those facts which can possibly be linked to a given sentence are filtered out from the set of all facts. We call this stage *Candidate Generation*. The primary objective of this phase is to make sure that no aligned fact is left out, while it is okay to still let in some of the facts which may not be aligned as these can be filtered out in the next stage. The second stage called *Candidate selection* is responsible for selecting precisely the generations which are aligned to the given sentence. Figure 3.3 depicts the alignment pipeline and how the dataset can be utilised for the purpose of Fact to Text generation.

3.4.1 Candidate generation

The candidate generation step is a maximum recall step designed to filter out the facts which have no possibility of being aligned to the sentence from the set of all facts related to an entity. In order to filter out the facts, we develop a system to rank all the facts in the order of likeliness to be aligned with the sentence. Once all facts are ranked, we choose the top k (here k=10) facts for each sentence. The score for ranking is calculated on the basis of the syntactic and semantic similarity of the fact with the given sentence.

Given a fact f_i and the sentence s_j in target language l, to establish a syntactic match, we employ TFIDF by either translating the fact into the language of the sentence or translating the sentence into English. For translating the sentence to English the translations described in Section 3.2.2 are used. In order to translate the facts, if the Wikidata entity label is present in the desired language, we use that, otherwise we use the translation obtained from the IndicTrans module[62].

For assessing semantic alignment, we calculate the cosine similarity between the representations of the fact and the sentence using MuRIL[35]. Alternatively, we compute the similarity between their translations. While exploring different models like mBERT[19], XLM-R[16], and LaBSE[23], we discovered that MuRIL outperformed the rest when evaluating a small dataset of 500 examples specifically annotated for Stage-1 quality assessment. For each (fact, sentence) pair, we obtain a similarity score, denoted as $sim(fi, sj)$, ranging from 0 to 1.

Thus the final similarity score is a sum of the following four components:

- $TFIDF - cos(translate(fi,l), s_j)$: cosine similarity between the TF-IDF vectors of the fact translated to the target language of the sentence and the sentence, contributing to syntactic match

- $TFIDF - cos(fi, translate(sj, English))$: cosine similarity between the TFIDF vectors of the sentence translated to English and the fact e, contributing to syntactic match

- $MuRIL - cos(f_i, s_j)$: cosine similarity of the MuRIL embeddings of English facts and the native language sentence, contributing to semantic match

- $MuRIL - cos(translate(fi, l), translate(sj, English))$: cosine similarity of the MuRIL embeddings of the fact translated to target language and the sentence translated to English, contributing to the semantic match

To filter out sentences which do not possibly align with any of the given facts, we set a threshold τ, where any sentence for which the similarity score with the best matched fact is lower than τ is excluded. After manual inspection, we set τ to 0.65. From the remaining sentences, we retain a maximum of top-K facts, sorted based on their scores.

3.4.2 Candidate selection

The candidate selection stage takes as input the output of the candidate generation pipeline i.e. each sentence with its top k facts. In order to filter and retain only the strongly the strongly aligned sentence we propose the second stage of our aligner. This aligner treats the alignment procedure as a classification problem by looking at each fact separately and classifying if it is entailed in the given sentence. For this purpose we propose two different approaches. They are described in the following sections.

3.4.2.1 Transfer learning from NLI

The natural language inference task [9] is the task where given a premise sentence and a hypothesis sentence, the task is to predict whether the hypothesis entails, contradicts or is neutral to the given premise. The NLI task is very similar to the task of aligning facts with sentences. Each fact can be considered analogous to the hypothesis and the sentence to be analogous to the premise. For every sentence fact pair, we aim to predict if the sentence entails the fact, contradicts the fact or is neutral to the fact.

Multiple multi-lingual language models have been made publicly avaibale after finetuning on the popular Cross-Lingual Natural Language Inference(Xtreme-XNLI)[17] dataset. We conduct experiments using several multi-lingual NLI models, namely XLM-R, mT5, and MuRIL. We leverage their Xtreme-XNLI finetuned checkpoints obtained from Huggingface and evaluate their performance on the alignment problem between facts and sentences. During inference, we provide the input "sentenceSEPfact" to these models. If the model predicts entailment, we consider the pair of (fact, sentence) to be aligned; otherwise, they are considered not aligned. Consequently, we select a subset of facts from the output of the candidate generation module for each sentence.For evaluating each model, the selected fact list is then compared against the golden fact list, which is human-annotated specifically for the given sentence.

3.4.3 Distant supervision based approaches

Distant supervision refers to the process of generating training data by making use of an already existing database. The idea is to convert the data from this existing database into a format which mimics the task at hand. For this purpose we use the Knowledge Enhanced Language Modelling (KELM) [49] dataset. KELM is a distantly supervised dataset with automatically aligned (Wikipedia sentence, Wikidata facts) for English language. For a Wikipedia page corresponding to Wikidata entity e, a sentence s is aligned with a Wikidata fact $f = e, r, e$ if s contains subject e and object e.

In order to modify this dataset for our use-case, we model our alignment task as a binary classification task where given a (English fact, low resource language sentence) pair, we train a binary classifier to predict if the sentence entails the information present in the given fact or

not. The input to the binary classifier is a string "sentence<SEP>subject|predicate|object and the output is one of the two classes. The classifier architecture utilises one of the pretrained transformer models in order to obtain the embeddings for the input string which is then passed through a neural classifier head.

A positive instance is generated for each sentence in the dataset for every fact that is aligned with that sentence. For instance, if sentence "s" has two aligned facts "f_1" and "$f2$", we create two positive instances accordingly. In addition to the positive instances, we also create corresponding negative instances as follows. Since it is desirable for the negative instances to be hard negatives and not trivial ones where the facts are already too different from the sentence, we devise a pipeline to carefully choose the negative instances. We sort all the other sentences on the same Wikipedia page as "s" based on their semantic similarity in a descending order. This semantic similarity is calculated based on the cosine similarity of the Distil-Bert-Base[68] embeddings of the Wikipedia sentences as the sentences are only in English. Since finding the top k similar sentences can be very compute intensive, we used FAISS [6] maximum inner product search (MIPS) package to find sentence similar to the given sentence. From the top 10 sentences, we randomly select a sentence "s " while excluding the top two sentences to avoid potential similarities and accidentally labelling an actual positive sample as negative. The fact extracted from sentence "s " is combined with the original sentence "s" to form the negative instance. The dataset is then divided into a 90:10 ratio for training and validation purposes. Overall, the dataset consists of 1,177,636 instances for training (54% positive, 46% negative) and 130,849 instances for validation (54% positive, 46% negative).

The following sections describes the results obtained by each of the methods described above.

3.4.4 Results

Table 3.3 Shows the F1 scores for various candidate selection approaches, compared against some strong baselines. In addition to our proposed models based on transfer learning and distant supervision, we also compare our results with the alignment baselines (KELM-style and WITA-style). For the TF-IDF-based aligner, we utilize the candidates generated during the initial stage of the process. As for the KELM and WITA-style aligners, we strictly follow the ranking algorithm outlined in their respective papers without employing the stage-1 aligner. All experiments are conducted on a machine equipped with four 10GB RTX 2080 GPUs. During the fine-tuning process, we train the models for five epochs while incorporating an L2-norm weight decay of 0.001 and a dropout rate of 0.1. The learning rates are set at 1e-5, 2e-5, and 1e-3 for XLM-RoBERTa, MuRIL, and mT5, respectively. The batch sizes are configured as 32, 32, and 16 for XLM-RoBERTa, MuRIL, and mT5, respectively. Notably, our observations

[6] https://github.com/facebookresearch/faiss

	hi	mr	te	ta	en	gu	bn	kn	Avg.
Baselines									
KELM-style [2]	0.493	0.426	0.368	0.451	0.41	0.372	0.436	0.338	0.411
WITA-style [26]	0.507	0.574	0.517	0.459	0.602	0.500	0.535	0.530	0.528
Stage-1 + TF-IDF	0.750	0.685	0.693	0.718	0.737	0.701	0.787	0.647	0.715
Distant supervision based methods									
MuRIL	0.763	0.684	0.74	0.755	0.705	0.785	0.624	0.677	0.717
XLM-Roberta	0.781	0.69	0.765	0.739	0.765	0.785	0.669	0.724	0.740
mT5	0.79	0.714	0.776	0.786	0.766	0.8	0.698	0.705	0.754
Transfer learning based methods									
MuRIL	0.716	0.717	0.765	0.751	0.734	0.787	0.795	0.718	0.748
XLM-Roberta	0.772	0.767	0.78	0.812	0.79	0.805	0.831	0.727	0.786
mT5	0.902	0.831	0.841	0.886	0.845	0.851	0.751	0.785	0.837

Table 3.3: Stage-2 (Fact, Sentence) Candidate Selection F1 Scores across different methods

reveal that mT5 with transfer learning showcases the most optimal performance among the evaluated models.

3.5 Dataset Analysis

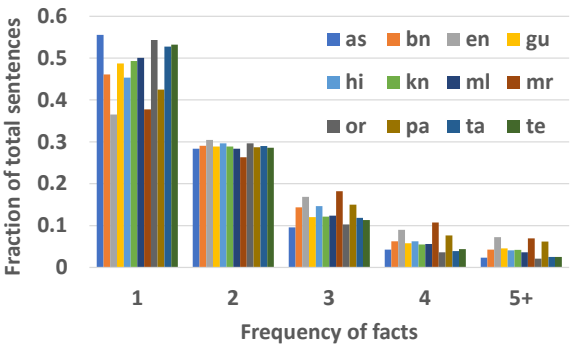

Figure 3.4: Fact Count Distribution across languages

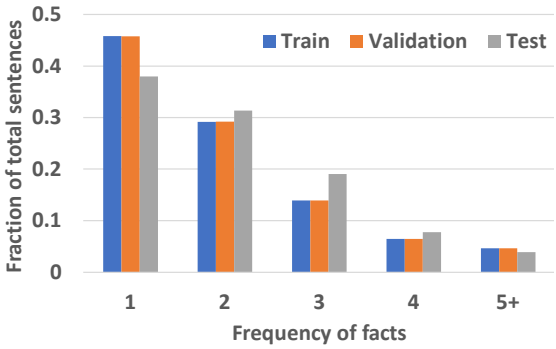

Figure 3.5: Fact Count Distribution across data subsets

In this section we present an analysis of the contsructed dataset along various axes. The dataset comprises of text from 12 different languages in total aligned to English facts. These languages include (in alphabetic order) : Assamese(as), Bengali (bn), English(en), Gujarati (gu), Hindi (hi), Kannada (kn), Malayalam, (ml) Marathi (mr), Odia (Or), Punjabi(pa), Tamil (ta), Telugu(te). The total dataset contains more than 0.55 million automatically aligned sentence

hi	occupation, date of birth, position held, cast member, country of citizenship, award received, place of birth, date of death, educated at, languages spoken written or signed
mr	occupation, date of birth, position held, date of death, country of citizenship, place of birth, member of sports team, member of political party, cast member, award received
te	occupation, date of birth, position held, cast member, date of death, place of birth, award received, member of political party, country of citizenship, educated at
ta	occupation, position held, date of birth, cast member, country of citizenship, educated at, place of birth, date of death, award received, member of political party
en	occupation, date of birth, position held, country of citizenship, educated at, date of death, award received, place of birth, member of sports team, member of political party
gu	occupation, date of birth, cast member, position held, award received, date of death, languages spoken written or signed, place of birth, author, country of citizenship
bn	occupation, date of birth, country of citizenship, cast member, member of sports team, date of death, educated at, place of birth, position held, award received
kn	occupation, cast member, date of birth, award received, position held, date of death, performer, place of birth, author, educated at
pa	occupation, date of birth, place of birth, date of death, cast member, country of citizenship, educated at, award received, languages spoken, written or signed, position held
as	occupation, date of birth, cast member, position held, date of death, place of birth, country of citizenship, educated at, award received, member of political party
or	occupation, date of birth, position held, cast member, member of political party, place of birth, date of death, award received, languages spoken, written or signed, educated at
ml	occupation, cast member, position held, date of birth, educated at, award received, date of death, place of birth, author, employer

Table 3.4: Top-10 frequent fact relations across languages.

fact pairs. In total there are more than 300 unique predicates from the set of English facts making a model trained on this data to easily generalise on the task of text generation for the given domain.

On average, each sentence in the dataset contains slightly more than 2 aligned facts. Figure 3.4 represents the distribution of aligned facts per sentence across languages. As it can be seen, Assamese has the highest proportion of sentences which have only one have aligned whereas English has the lowest. Among all languages English also has the highest number of sentences with more than 5 facts. Figure 3.5 shows the distribution of fact count across the dataset partitions. As it can be seen, the proportion is very similar with the test set having the least fraction of sentences with just one aligned fact. This could possibly be because the test set

contains very high quality manually annotated samples where all the information in the sentence is necessarily covered by the aligned facts (complete alignments).

Table 3.4 provides the top 10 fact relations across the 12 chosen languages. As it can be seen, occupation and date of birth are the two most common relations in the entire dataset followed by other relations like position held, cast member etc.

3.6 Summary and Conclusion

In conclusion, the chapter highlights the pressing need for automated generation of descriptive text in low resource (LR) languages from English fact triples, particularly in critical scenarios such as Wikipedia text generation based on English Infoboxes. Previous research efforts have primarily focused on English fact-to-text (F2T) generation, leaving a significant gap in cross-lingual alignment and generation for LR languages.

Addressing this gap, our work proposes two unsupervised methods for cross-lingual alignment, aiming to bridge the language barrier between English structured facts and LR sentences. We also emphasised on gathering data, followed by meticulous data pre-processing to guarantee the inclusion of high-quality samples. Furthermore, a thorough analysis of the data was conducted, focusing on key parameters for deeper insights. Our investigations revealed that the most optimal performance was achieved by employing mt5 with transfer learning, fine-tuned specifically for the Natural Language Inference (NLI) task. Based on this discovery, we proceeded to utilize this model to curate our final dataset.

We contribute the XALIGN dataset, comprising a substantial corpus of XF2T pairs across eight languages, including 5402 pairs that have been manually annotated. This dataset, holds immense significance within the field of text generation and information extraction in Natural Language Processing (NLP) and stands as a noteworthy contribution. This chapter sets the stage for further exploration and refinement of cross-lingual F2T systems, laying the groundwork for more effective and efficient text generation and information extraction in diverse linguistic contexts. The work explained in the upcoming chapters utilises the constructed dataset for multiple tasks like Cross lingual fact extraction CLFE, cross lingual fact to text generation XF2T and cross lingual fact to long text generation XFLT

Chapter 4

Cross Lingual Fact Extraction

4.1 Overview

The rise of knowledge graphs as extensive and structured sources of information has sparked significant research interest in automating their construction and enrichment [29], [89]. With a vast collection of more than 99 million entities, Wikidata [76] stands as one of the largest publicly accessible knowledge graphs. Its expansive nature has facilitated its utilization across various applications, including text generation [38] and question answering [71], [46], among others.

A knowledge graph comprises interconnected facts, wherein each fact is typically represented as a triplet encompassing two entities and a semantic relation that connects them. This information is encoded as a triple $< h, r, t >$ where h, r and t represent the subject entity, the relation and the tail entity, respectively.

This chapter addresses the challenge of extracting factual information from low resource languages and proposes the task of multi-lingual and cross-lingual fact to text extraction (CLFE) for seven Low Resource (LR) Indian Languages and English. The aim of CLFE is to extract English triples, representing factual information, from text written in Indian languages. To the best of our knowledge, this is the first endeavor to perform multilingual and cross-lingual fact extraction specifically from LR Indian Languages.

Through this chapter, we highlight the following contributions :

- This chapter proposes the problem of cross-lingual and multi- lingual fact extraction for LR Indian languages.
- The chapter describes an end-to-end generative approach for extracting subject centric factual information from LR Indian language text, which shows significant improvements over classification based pipelines.
- We train multiple multi-lingual CLFE models which lead to an overall F1 score of 77.46

Multilingual Texts	Head Entity	Extracted Facts
Elon Reeve Musk (born 28 June 1971) is a South African-Canadian-American veteran businessman, investor, engineer, and inventor.	Elon Musk	< nationality, South_Africa> < nationality, Canda> < nationality, USA> < date_of_birth, 28_June_1971> < occupation, engineer> < ocuupation, entrepreneur> < occupation, inventor> < occupation, investor>
एलन मस्क (जन्म 28 जून 1971) एक दक्षिण अफ्रीकी-कनाडाई-अमेरिकी दिग्गज व्यापारी, निवेशक, इंजीनियर, और आविष्कारक हैं।	Elon Musk	< nationality, South_Africa> < nationality, Canda> < nationality, USA> < date_of_birth, 28_June_1971> < occupation, engineer> < ocuupation, entrepreneur> < occupation, inventor> < occupation, investor>
এলন মাস্ক (জন্ম 28 জুনজু 1971) দক্ষিণ আফ্রিকা-কানাডিয়ান-আমেরিকান প্রবীণ ব্যবসায়ী, বিনিয়োগকারী, প্রকৌশলী এবং উদ্ভাবক।	Elon Musk	< nationality, South_Africa> < nationality, Canda> < nationality, USA> < date_of_birth, 28_June_1971> < occupation, engineer> < ocuupation, entrepreneur> < occupation, inventor> < occupation, investor>
એલોન મસ્ક (જન્મ 28 જૂન 1971) એ દક્ષિણ આફ્રિકા-કેનેડિયન□અમેરિકન પીte ઉદ્યોગપતિ, રોકાણકાર, ઇજનેર અનેશોધક છે.	Elon Musk	< nationality, South_Africa> < nationality, Canda> < nationality, USA> < date_of_birth, 28_June_1971> < occupation, engineer> < ocuupation, entrepreneur> < occupation, inventor> < occupation, investor>

Figure 4.1: Example Inputs and outputs of CLFE task. Text from any language along with entity of interest(head entity) is provided as input to extract English Facts(relation and tail entity pairs). The same sentence may or may not be present in all languages.

The remaining chapter is organised as follows : Section 4.2 describes the usage of the dataset in context of the CLFE task. Section 4.3 describes the methods applied for the task of cross lingual fact extraction. The following sections discuss the results achieved ans provides a conclusive summary to the chapter.

4.2 Dataset

Originally designed for cross-lingual data-to-text generation, due to its richly cross-lingual and multilingual nature, the XAlign dataset consists of 0.45 million pairs across eight languages, and proves to be a valuable resource for our task. Of these 0.45 million pairs, 5,402 pairs having undergone manual annotation, and have served as our golden test set. The sentences in the XAlign dataset are extracted from Wikipedia articles written in Indian languages, pertaining to entities classified under the human category.

However, the repurposing of the XAlign dataset for CFLE introduced a multitude of challenges. If we were to treat each relation as a distinct class in classification-based approaches, we observe a high level of class imbalance. Among approximately 367 unique relations (classes), the most frequent class alone accounts for 27 % of the data, while the top 20 classes contribute to 90 % of the dataset. On average, each sentence in the dataset is associated with 2.02 aligned facts.

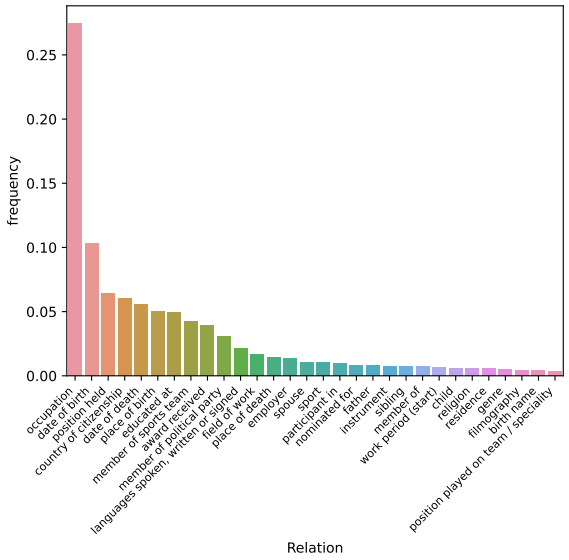

Figure 4.2: Distribution of Top 30 most frequent relations in the dataset

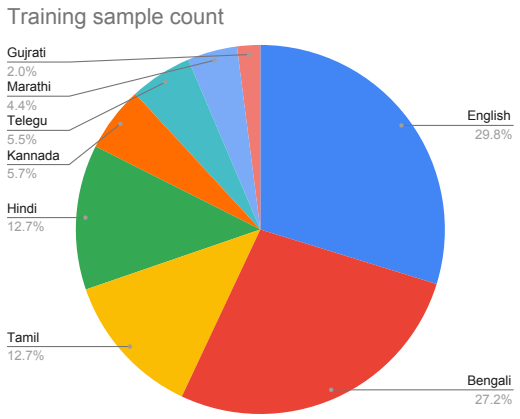

Figure 4.3: Distribution of the 8 languages in the training set

Moreover, the dataset presents the challenge of partial alignment. While the sentences in the test set possess complete coverage in terms of aligned facts, the aligned facts do not encompass the entire information present in the sentences from the training set. This characteristic of the dataset can potentially penalize the model during training even for generating accurate facts, which subsequently impacts the recall scores during testing. The distribution of the top 30 most frequent relations in the dataset is visualized in Figure 4.2. Additionally, Figure 4.3 illustrates the distribution of languages in the dataset. It is evident from the figures that the dataset exhibits a high degree of imbalance, both in terms of relations and languages.

4.3 Methodology

We put forward two distinct approaches for the CLFE task: a classification-based approach that initially extracts the tails and subsequently predicts the relation, and a generative approach that concurrently performs both tasks in a single step.

4.3.1 Tail Extraction and Relation Classification(TERC)

The TERC pipeline (Figure 4.4) encompasses a two-step process. In the initial step, we extract the tails of facts from the source language text. To achieve this, we employ the IndicTrans [62] translation model to convert the input text into English.

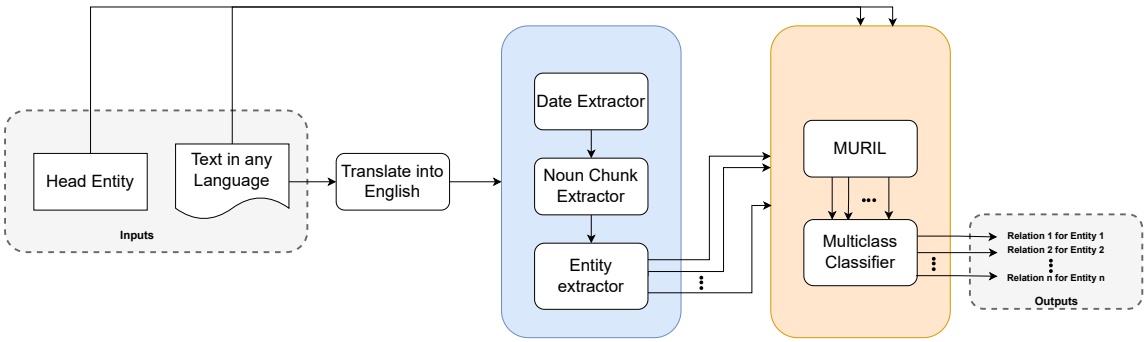

Figure 4.4: Pipeline Architecture for CLFE

Additionally, we identify any date expressions within the text and normalize them to a standardized format. To prevent dates from being considered as entities, we replace them with a placeholder token in the original text. As tail entities are limited to nouns or proper nouns, we utilize the noun chunk extractor from the spaCy library [31] to extract all noun chunks. The process of selecting tail entities from the set of noun chunks involves the following steps:

- Removal of entities that match the head: Entities that have a high lexical overlap with the head entity are excluded from consideration. This step ensures that only distinct tail entities are retained in this stage of the pipeline.
- Filtering out pronouns: Noun chunks with pronoun roots are discarded to eliminate pronouns from the selection. Tails in the dataset are never represented as pronouns, so any noun phrases containing pronoun heads are pruned.
- Selection of continuous spans with ADJ and PROPN PoS tags: Continuous spans of tokens with adjective (ADJ) and proper noun (PROPN) parts of speech (PoS) tags are chosen as individual entities. Since tails can consist of multiple words and may include adjectives within their span, PoS tags are used to identify maximal spans for each recognized proper noun.
- Selection of the root of the noun chunk: If the PoS tag of the noun chunk's root is NOUN, it is selected as a separate entity. This step ensures that the primary noun in the noun chunk is considered as an individual entity in addition to any other detected entities.

The subsequent step in the pipeline involves predicting a relation for each extracted tail entity. To accomplish this, we employ a pretrained MuRIL model [35] to generate a joint representation that incorporates the head entity, tail entity, as well as the source language input text, which is then provided as input to a classifier, which predicts the relation between the head and tail

entities. During training, given a sentence and a <head, tail> pair, the classifier learns to predict the relation by considering the ground truth tails as input. We employ weights based on the **inverse logarithm of the class distribution** in the loss function, to systematically address the class imbalance issue. This weighting approach outperforms both the standard inverse class distribution and unweighted loss methods. When evaluating the performance of the pipeline architecture, the tails extracted from the translated input text are aligned with the ground truth tails. This alignment entails assigning one ground truth tail entity to each extracted entity without duplication. We disregard some extracted entities that do not have any overlap with the ground truth. Additionally, certain ground truth entities may not be assigned to any of the extracted entities, resulting in a lower recall. The assignment process relies on a similarity score and a threshold. The similarity score between two entities is computed as the sum of cosine similarities of GloVe vectors and the intersection over union of terms. Using a threshold of 0.7, we achieve a precision of 0.54 and a recall of 0.77. For these aligned tails, we make predictions, while also calculating the evaluation metrics based on this alignment.

4.3.2 End to End Generative extraction

Prior research in the field of monolingual fact extraction has demonstrated that a model that jointly performs tail extraction and relation prediction tends to outperform a disjoint approach [45]. Unlike the pipeline approach mentioned earlier, this joint approach benefits from a two-way interaction between tail extraction and relation prediction, leading to improved performance as these tasks are not independent of each other. In line with this, we propose an end-to-end approach to the fact extraction problem that can simultaneously extract tails and their corresponding relations with the head entity.

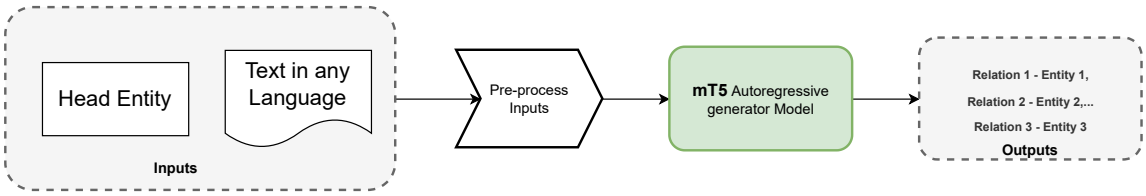

Figure 4.5: End to end architecture for CLFE

Framing this problem as a text-to-text task, we utilize the mT5 [81] auto-regressive sequence-to-sequence model to generate relations and tails given the head entity and input text as inputs, training this model using cross-entropy loss. We employ a generative approach, enabling a more flexible and unrestricted information extraction process, where the set of relations and tails is not constrained.

We conduct experiments with three variations of this pipeline, with the facts linearized and represented as the target text by concatenating the head and tail entities with special tokens.

For a given sentence S, hence, if the corresponding i facts are $[h, r_1, t_1], [h, r_2, t_2]....[h, r_i, t_i]$, the target text would be $< R > r_1 < T > t_1 < R > r_2 < T > t_2.... < R > r_i < T > t_i$.

The first variation of our experiments involves fine-tuning the pretrained mt5 model for the fact extraction task across all languages. In the second experiment, we employ script unification by transliterating the input text of all languages (except English) into the Devanagari script. This approach leverages the high vocabulary overlap among multiple Indian languages to facilitate model training. In the third variation, we train separate bi-lingual fact extraction models for each language.

For the Two-Phase approach, we train the relation prediction block of the model. The classifier is based on the MURIL encoder model from Google, which consists of 12 encoding layers with an output dimension of 768. During training, the 12th layer of MURIL and the layers in the feed forward network are unfrozen. Using the Adam optimizer with an initial learning rate of 1e-4, step scheduling with a step size of 2 and a gamma value of 0.3, we train batches of 16 facts to optimize the Cross Entropy Loss. As mentioned earlier, we use inverse log frequency of classes as weights for the cross entropy loss to effectively combat the class imbalance issue. The training process for relation prediction takes approximately 5 hours on one GPU card.

For the Generative approach, we utilize the pretrained mT5 model and finetune it for 5 epochs in all experiments. The learning rate is set to 0.001 with a weight decay of 0.01. To mitigate overfitting, a dropout rate of 0.1 is applied during training. We use the Adafactor optimizer to optimize the Cross Entropy Loss during generation.

4.4 Results

Table 4.1 presents the summarized results of the different fact extraction approaches discussed earlier.

The findings indicate that the open-ended approach achieves the highest F1 score, offering greater flexibility in terms of possible entities and relations. Another notable observation is that training separate bilingual models performs better than using a combined model for English and Bengali, which are the two most prevalent languages in the dataset, accounting for 54.44 % of the training data. This suggests that multilingual training proves advantageous due to shared learning across Indian languages. Additionally, script unification, specifically transliterating input scripts to Devanagari, benefits the Dravidian languages (te, ta, kn) in the dataset.

Since the current evaluation criteria require an exact word match between the predicted and ground truth tails to determine correctness, it should be duly acknowledged that the model's actual performance might be better than the reported numbers. However, this approach fails to account for cases where the predicted and ground truth tails are synonymous. For instance, if the model predicts 'writer' as the occupation while the ground truth label is 'author', both

	te	bn	ta	gu	mr	en	hi	kn	All languages		
	F1	F1	F1	F1	F1	F1	F1	F1	P	R	F1
Classification with GT Tails	69.19	67.50	89.44	85.74	51.38	72.87	87.10	79.74	79.04	77.93	75.37
TERC	43.66	41.96	52.19	40.30	44.59	50.80	50.46	42.57	40.45	53.71	46.15
E2E Cross-lingual Generative Model	71.82	75.56	82.82	**72.36**	**77.79**	76.28	**86.62**	68.04	74.09	**81.15**	**77.46**
E2E generation w script unification	**72.51**	75.38	**85.21**	72.04	77.19	74.56	83.44	**70.46**	78.49	76.15	77.29
Bilingual Models	70.94	**78.01**	83.71	67.84	71.91	**76.64**	86.49	63.19	**79.79**	71.63	75.49

Table 4.1: Precision, recall and F1 scores of various methods applied on all languages in the Test set. Note that "Classification with GT Tails" uses tails from ground truth as input for the Relation Prediction model and hence does not represent a complete pipeline

terms convey the same meaning but would not be considered a match under the strict evaluation scheme.

4.5 Summary and Conclusion

This chapter focuses on the extraction of factual information from low resource languages and presents the task of multi-lingual and cross-lingual fact to text extraction (CLFE) for seven Low Resource (LR) Indian Languages along with English. Notably, this research has been the first attempt to perform multilingual and cross-lingual fact extraction specifically from LR Indian Languages. By undertaking this research, the coverage of existing knowledge graphs is significantly expanded by leveraging the factual knowledge present in Indic texts. The study reveals that while script-unification benefits certain languages, a single multilingual end-to-end generative pipeline yields superior performance with an overall F1 score of 77.46. This work lays the groundwork for future research in knowledge extraction from LR Indic language text. Subsequent chapters build upon the tools and findings presented here to advance research in fact-to-text generation.

Chapter 5

Approaches for Cross Lingual Fact to Text Generation

5.1 Overview

Fact-to-text generation systems have predominantly focused on English development primarily because of the abundance of suitable datasets. Conversely, due to the scarcity of datasets for low-resource languages, these systems have been exclusively developed for English. This chapter focuses on the problem of crosslingual fact-to-text (XF2T) generation, which involves automatically generating descriptive human-readable text from structured input data in multiple languages. The XF2T generation task tackled in this chapter takes a set of English facts as input and aims to generate a sentence that accurately captures the semantic meaning of the facts in the specified target language. By exploring different models, training setups, and strategies, the chapter aims to build a robust XF2T system that can bridge the gap between English-centric fact-to-text generation and the need for crosslingual capabilities. In Fig. 5.1, an illustration is presented, demonstrating an instance where a collection of English Wikidata facts is employed to generate a sentence across different languages.

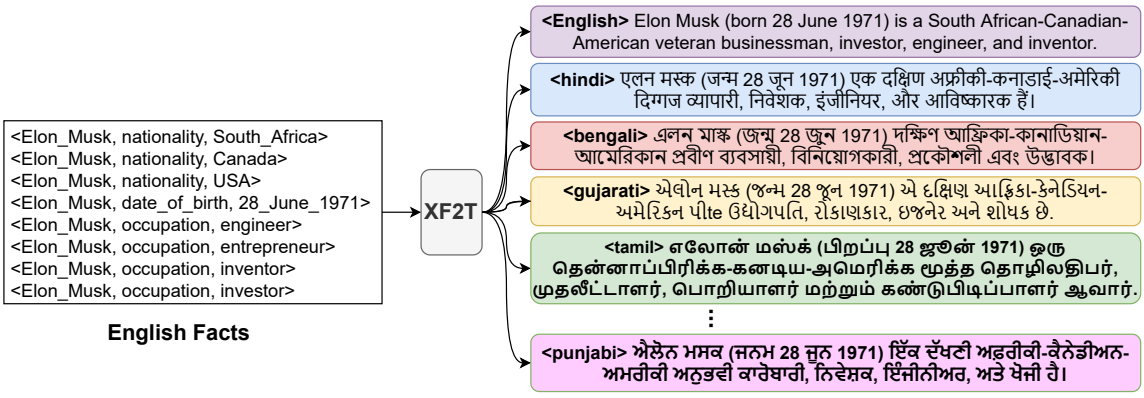

Figure 5.1: Example showing generation of natural language sentences from English facts

This chapter delves into the XF2T problem itself using the extended version of the XAlign dataset, XAlignV2 as described in Chapter 3. We explore the different Transformer-based multi-lingual encoder-decoder models such as the vanilla Transformer, IndicBART, and mT5. The chapter investigates various training setups and strategies for improving XF2T generation, including multi-lingual data-to-text pre-training, fact-aware embeddings, fact ordering and structure-aware input encoding.

Experimental results are presented, comparing the performance of different models and strategies across the twelve languages covered in the XALIGNV2 dataset. The chapter reveals that the mT5 model with fact-aware embeddings and structure-aware input encoding achieves the best results on average. Evaluation metrics such as BLEU, METEOR, LABSE score and chrF++ are used to assess the quality of the generated crosslingual text.

5.2 XF2T Approaches

This section provides a detailed account of the approaches tried for the task of cross lingual fact to text generation. We experiment with multiple components involved in the process of generation, like the encoding of the input, the pretrained transformers used, variations of extended pretraining and more. We describe them in detail in the following subsections.

5.2.1 Encoding of Input for the transformer models

In order to feed our input of set of facts into the transformer model, we need to come up with an encoding strategy for the input which can convert the set of facts into a logical string

Every input instance comprises several facts $F = f_1, f_2, \ldots, f_n)$ and a section title (t). A fact (f_i) consists of subject (s_i), relation (r_i), object (o_i), and (m) qualifiers $(Q = q_1, q_2, \ldots, q_m)$. Qualifiers offer supplementary details pertaining to the fact. Each of these qualifiers (q_j) can be associated with the fact through a fact-level property known as qualifier relation (qr_j).

The encoding of each fact f_i involves representing it as a string, while the complete input entails the concatenation of these strings from all facts within the set F. The string representation for a fact f_i is denoted by "$\langle S\rangle s_i \langle R\rangle r_i \langle O\rangle o_i \langle R\rangle qr_{i1} \langle O\rangle q_{i1} \langle R\rangle qr_{i2} \langle O\rangle q_{i2} \ldots \langle R\rangle qr_{im} \langle O\rangle q_{im}$", where $\langle S\rangle$, $\langle R\rangle$, and $\langle O\rangle$ corresponds to special tokens which are added specifically to the vocabulary of the encoder. Ultimately, the comprehensive input, encompassing n facts, is acquired through the following process: "generate [l]: $f_1\ f_2\ \ldots f_n\ \langle T\rangle\ [t]$" where "[l]" represents one of the 12 available languages, $\langle T\rangle$ signifies the token used as a delimiter for section titles, and t represents the specific title of the section.

5.2.2 Standard Transformer-Based Baselines

As our first set of experiments, we explore multiple pretrained standard transformer based baselines in order to evaluate and compare their respective performance. In the XF2T generation process, we employed popular multi-lingual text generation models, namely mT5-small and IndicBART [18], and finetuned them on the Train+Validation portion of the XAlign dataset. We trained a single model in cross lingual fashion using data from multiple languages without the requirement of translation,

As it can be observed from Table 5.1, mT5 performs the best overall. Hence for all the following experiments we use the mT5-small model as our base model.

5.2.3 Monolingual, Bilingual, Multilingual and Translation-based models

In order to analyse the practicality of our cross lingual training setup, we conduct experiments using various training setups to explore their effectiveness. Firstly, we focused on constructing bilingual models in which the input was always in English, while the output could vary among the 12 languages under consideration. Thus this training setup requires 12 individual bi-lingual models to be constructed. This approach is based on the observation that bilingual models tend to exhibit improved accuracy for certain language pairs in cross-lingual scenarios.

It is important to note that our particular case maintains English as the consistent input language, necessitating the training of separate bilingual models for each target language.

The requirement to manage multiple models, one for each language, which can be quite cumbersome in practice, is a drawback of this approach, nonetheless.

Additionally, we also trained two translation-based models to explore alternative strategies. In the "translate-output" setting, we developed a single English-only model that processes English facts as input and generates text in English as output. During the testing phase, the generated English output is then translated to the desired language using the IndicTrans [63] translation tool.

Conversely, in the "translate-input" setting, we opted to translate the English facts into the respective target language and utilized the translated version as input for training a single multilingual model across all languages. When performing the translation process, if any mapped strings for entities were available in Wikidata, we directly employed them. However, both of these approaches introduce a limitation: the need for translation during the testing phase.

Our default setting remains training a single cross lingual model for all the 12 languages without any translations on either of the sides. This arrangement benefits from shared knowledge and vocabulary across different languages. Furthermore, training data from different high resource languages helps improve the generation for low resource languages which lack sufficient training data. This approach also mitigates the propagation of loss added due to the translation module.

5.2.4 Continued Pre-training

Pretraining has become a widely used approach to achieve highly effective models, even when working with limited labeled data. Furthermore, the application of domain-specific and task-specific pretraining has demonstrated additional performance improvements [30]. In our study, we explore four distinct pretraining strategies that are implemented on top of an already pretrained encoder-decoder model, prior to fine-tuning it on the xalignv2 dataset.

The first strategy is **multi-lingual pretraining** [77], which involves leveraging a larger, albei't noisy, corpus of data obtained from Wikipedia for the English F2T (Fact-to-Text) task. This dataset consists of 542,192 data pairs spread across 15 categories and is constructed by combining English Wikipedia [77] data with Wikidata triples. To generate the multi-lingual pretraining data, we translate English sentences extracted from the Wikipedia-based dataset into our low-resource (LR) languages. As a result, the multi-lingual pretraining data encompasses approximately ∼6.5M data pairs. For the translation process, we utilize IndicTrans.

The second strategy focuses on **translation-based pretraining**, recognizing that translation serves as a fundamental task in facilitating effective cross-lingual NLP. In this approach, we pretrain the mT5 model on translation data specific to English-to-other-language pairs, with ∼0.25M instances available per language.

The third strategy combines the two aforementioned methods in a two-stage pretraining approach. Initially, we conduct translation-based pretraining during the first stage. Subsequently, in the second stage, we proceed with multi-lingual pretraining. We call this **Multi stage pretraining**

Lastly, the fourth strategy involves **multi-task pretraining**, encompassing training for both translation and XF2T (cross-lingual Fact-to-Text) tasks simultaneously. Differing from the two-stage method where pretraining is performed separately for translation and XF2T tasks, the multi-task pretraining approach leverages a joint multi-task learning setup to simultaneously address both tasks.

5.2.5 Fact-aware Embeddings

The mT5 model receives input comprising both token embeddings and position embeddings. In the case of XF2T (Cross-lingual Fact-to-Text) generation, the input consists of a collection of facts, wherein each fact comprises distinct semantic units that fulfill different roles, namely subject, relation, and object. To enhance the mT5 model's understanding of these facts, we augment the standard input by incorporating specific role embeddings, known as fact-aware role embeddings.

To be more specific, we introduce four role IDs: ROL1 for subjects, ROL2 for relations and qualifier relations, ROL3 for objects and qualifier tokens, and ROL0 for all other tokens that do not fall into the aforementioned categories (see Fig. 5.2). These role embeddings are randomly

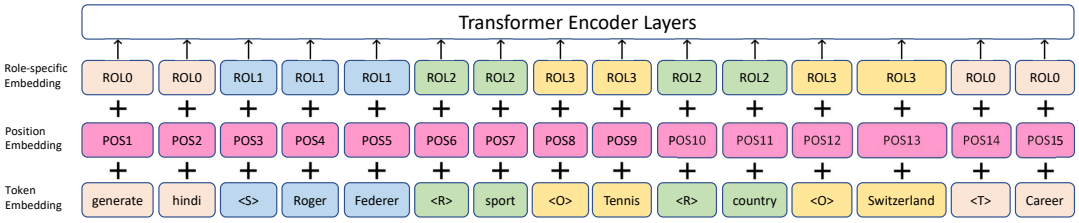

Figure 5.2: English facts being passed as input to mT5's encoder with token, position and (fact-aware) role embeddings.

initialized and learned during the training process. By explicitly indicating the role played by each token within the input facts, we anticipate that the model will gain a deeper understanding of the underlying semantics, thereby facilitating improved XF2T generation.

Furthermore, we conducted additional experiments involving (1) separate role embeddings for qualifier relations and qualifiers, as well as (2) the inclusion of fact ID embeddings. In the latter case, if the input contains K facts, each fact is assigned a unique fact ID, and all tokens corresponding to a specific fact receive the same fact ID embedding. However, the results obtained from these experiments did not yield any significant improvements, and therefore, we have chosen not to include them in our reported findings.

5.3 Results

In this section we compare the results from the multiple experiments described so far.

5.3.1 Metrics

We evaluate our scores based on one or more of these 4 metrics : BLEU [55], METEOR [6], chrF++[57], and LaBSE Score. While BLEU, METEOR and chrF++ are standard metrics and consider only the syntactic similarity of the generated content with respect to the reference, we define LaBSE score to measure the semantic similarity as well.

LaBSE score works on the similar concept as BERT score [85]. We simply take the cosine similarity of the LaBSE[23] embeddings of the generated and reference sentences.

5.3.2 Standard Transformer-Based Baselines

As seen in Table 5.1, mT5 outperforms other pretrained models. Both pretrained models drastically outperform the vanilla transformer on all languages. As it can be seen, IndicBART [18] performs much better than the mt5[81] model over Bengali across all metrics, however performs significantly worse on English, possible due to the differences in their pretraining.

	Vanilla Transformer				IndicBART				mT5			
	BLEU	METEOR	chrF++	LaBSE	BLEU	METEOR	chrF++	LaBSE	BLEU	METEOR	chrF++	LaBSE
hi	35.04	63.46	60.85	86.04	40.44	66.41	66.27	88.51	**44.65**	**68.58**	**68.49**	**90.28**
mr	18.28	50.66	49.87	77.73	**28.08**	55.35	57.73	82.79	26.47	**56.85**	**59.17**	**85.25**
te	6.95	36.17	41.70	77.44	**15.67**	41.52	50.40	80.46	14.46	**43.45**	**52.58**	**83.47**
ta	14.67	44.64	53.03	80.63	**19.37**	45.78	56.63	83.07	18.37	**46.15**	**57.42**	**84.53**
en	37.12	65.32	59.69	80.05	10.47	42.35	34.35	66.38	**46.94**	**70.60**	**65.20**	**84.91**
gu	15.66	47.70	46.29	79.46	19.16	47.92	49.30	78.64	**22.69**	**50.31**	**51.36**	**84.36**
bn	48.55	74.18	75.68	87.97	**55.90**	**79.29**	**80.51**	**91.44**	40.38	61.71	68.71	84.17
kn	4.78	28.96	37.60	71.90	10.30	**33.55**	46.65	77.13	**10.66**	32.58	**46.92**	**80.45**
ml	16.29	50.84	47.26	80.13	**27.41**	56.27	56.80	86.08	26.22	**56.71**	**57.01**	**86.53**
pa	17.76	50.27	44.73	77.82	22.32	53.20	50.74	81.34	**26.96**	**54.82**	**52.33**	**84.11**
or	39.94	61.09	62.79	81.33	22.16	53.76	58.30	77.56	**47.17**	**67.82**	**71.20**	**86.05**
as	8.08	29.27	31.24	60.18	**14.07**	**34.25**	**38.87**	63.58	12.61	32.93	36.91	**65.84**
Avg	21.93	50.21	50.89	78.39	23.78	50.80	53.88	79.75	**28.13**	**53.54**	**57.27**	**83.33**

Table 5.1: Comparison of different pretrained transformer models

5.3.3 Monolingual, Bilingual, Multilingual and Translation-based models

	Bi-lingual (12 models)				Translate-Output (1 model)				Translate-Input (1 model)				Multi-lingual (1 model)			
	BLEU	METEOR	chrF++	LaBSE	BLEU	METEOR	chrF++	LaBSE	BLEU	METEOR	chrF++	LaBSE	BLEU	METEOR	chrF++	LaBSE
hi	41.07	66.15	65.57	88.40	24.88	55.91	54.48	83.85	41.98	66.14	66.47	89.32	**44.65**	**68.58**	**68.49**	**90.28**
mr	16.74	49.36	48.40	78.00	20.62	46.87	52.23	82.59	24.90	54.56	57.25	83.14	**26.47**	**56.85**	**59.17**	**85.25**
te	12.23	37.85	44.94	78.93	14.13	38.69	50.36	82.78	13.11	40.83	49.64	81.19	**14.46**	**43.45**	**52.58**	**83.47**
ta	18.37	**46.57**	57.10	83.50	8.36	30.41	46.35	74.76	**19.23**	45.68	**57.54**	**84.73**	18.37	46.15	57.42	84.53
en	45.79	69.90	63.79	84.64	**50.81**	70.47	**65.43**	**85.73**	45.12	69.88	64.11	83.55	46.94	**70.60**	65.20	84.91
gu	12.49	38.73	37.01	72.69	18.23	42.25	46.27	79.97	20.84	48.71	49.30	82.10	**22.69**	**50.31**	**51.36**	**84.36**
bn	**53.61**	**75.42**	**78.12**	**89.87**	20.57	46.58	56.60	78.62	40.56	67.75	71.36	86.65	40.38	61.71	68.71	84.17
kn	8.71	31.02	41.16	75.62	7.93	27.58	44.47	78.97	7.75	30.82	41.44	75.96	**10.66**	**32.58**	**46.92**	**80.45**
ml	24.28	55.37	55.49	85.35	18.60	47.39	51.47	82.86	26.16	56.49	**57.22**	**87.36**	**26.22**	**56.71**	57.01	86.53
pa	21.92	51.10	47.82	80.64	26.24	53.18	51.57	83.19	24.42	51.64	49.28	80.95	**26.96**	**54.82**	**52.33**	**84.11**
or	45.53	62.91	65.30	82.09	9.37	29.40	37.80	75.41	43.43	64.12	65.20	83.67	**47.17**	**67.82**	**71.20**	**86.05**
as	9.76	26.48	29.80	56.93	7.15	25.25	32.19	62.38	10.89	30.27	35.00	64.05	**12.61**	**32.93**	**36.91**	**65.84**
Avg	25.88	50.91	52.88	79.70	18.91	42.83	49.10	79.30	26.53	52.24	55.32	81.90	**28.13**	**53.54**	**57.27**	**83.33**

Table 5.2: Comparison of different training setup

As it can be seen in Table 5.2, a single multilingual model performs better than translation based approaches, or even utilising 12 different bilingual models due to transfer of knowledge across languages.

5.3.4 Continued Pre-training strategies and fact aware embeddings

As it can be seen in Table 5.3, multilingual pretraining performs the best among all pre-training strategies. On BLEU score however, fact aware embeddings outperform the training strategies. Table 5.4 further shows the detailed language wise comparison of multi-lingual pretrained mT5 and mT5 with fact-aware embedding models with the vanilla mt5 baseline

No.	Method	BLEU	METEOR	chrF++
1	Multi-lingual mT5 (No pretraining, no fact-aware embeddings)	28.13	53.54	57.27
2	Multi-stage Pretraining	27.70	51.87	55.32
3	Multi-task Pretraining	28.45	51.87	55.20
4	Translation-only Pretraining	27.53	50.67	53.71
5	Multi-lingual Pretraining	28.71	**53.83**	**57.58**
6	Fact-aware embeddings	**29.27**	53.64	57.30

Table 5.3: XF2T scores on XAlignV2 test set using different pretraining strategies and fact-aware embeddings for the mT5 model

	Vanilla mT5			Multi-lingual Pretraining			Fact-aware embeddings		
	BLEU	METEOR	chrF++	BLEU	METEOR	chrF++	BLEU	METEOR	chrF++
hi	44.65	68.58	68.49	43.32	68.19	68.21	42.72	67.49	68.03
mr	26.47	56.85	59.17	27.64	56.34	57.74	29.06	55.40	57.97
te	14.46	43.45	52.58	15.94	42.71	52.40	16.21	42.14	51.25
ta	18.37	46.15	57.42	16.68	42.32	54.88	19.07	43.65	56.01
en	46.94	70.60	65.20	46.61	70.45	65.33	48.29	70.75	65.42
gu	22.69	50.31	51.36	21.39	47.98	50.14	23.27	50.00	50.64
bn	40.38	61.71	68.71	50.89	75.62	77.43	49.48	73.03	76.19
kn	10.66	32.58	46.92	11.61	33.00	47.18	11.57	33.44	46.66
ml	26.22	56.71	57.01	27.38	56.63	57.35	29.04	57.15	57.60
pa	26.96	54.82	52.33	26.04	54.17	52.50	28.65	55.19	53.38
or	47.17	67.82	71.20	44.97	66.49	70.64	41.75	63.77	67.96
as	12.61	32.93	36.91	12.00	32.04	37.15	12.16	31.61	36.44
Avg	28.13	53.54	57.27	28.71	53.83	57.58	29.27	53.64	57.30

Table 5.4: XF2T scores on XALIGNV2 test set using vanilla mT5, multi-lingual pretrained mT5 and mT5 with fact-aware embedding models

5.4 Conclusion and summary

In summary, this chapter provided a detailed analysis of various experiments tried for the task of cross lingual fact to text generation. We conclude that fact aware embeddings result in the best performing model with significant gains over the proposed baselines. While the work explored in this chapter resulted in fluent systems for generating natural language text, we must note that they have few limitations. The primary error in the xf2t systems is the presence of extraneous information in generation or in other words, hallucination. Furthermore, the approaches mentioned in this section can only generate a single sentence at a time, thus still needing manual intervention in order to generate the complete article. Upcoming chapters will focus on addressing these limitations.

Lang.	Input	Reference Text	Generated Text
hi	generate Hindi <S> Asha Nautiyal <R> member of political party <O> Bharatiya Janata Party <R> date of birth <O> 25 June 1969 <R> occupation <O> politician <R> country of citizenship <O> India <T> introduction	आशा नौटियाल (जन्म २५ जून, १९६९) एक भारतीय राजनीतिज्ञ हैं जो भारतीय जनता पार्टी से हैं ।	आशा नौटियाल (जन्म २५ जून, १९६९) एक भारतीय राजनीतिज्ञ और भारतीय जनता पार्टी की सदस्य हैं ।
en	generate English <S> Kedarnath Singh <R> date of death <O> 19 March 2018 <R> date of birth <O> 07 July 1934 <R> occupation <O> poet <R> languages spoken, written or signed <O> Hindi <R> country of citizenship <O> India <T> introduction	Kedarnath Singh (7 July 1934 - 19 March 2018) was an Indian poet who wrote in Hindi.	Kedarnath Singh (7 July 1934 - 19 March 2018) was a Hindi poet from Uttar Pradesh, India.
mr	generate Marathi <S> Théodore de Banville <R> date of death <O> 13 March 1891 <R> date of birth <O> 14 March 1823 <R> occupation <O> writer <R> country of citizenship <O> France <T> introduction	थेओदोर दि बॅनव्हिल (मार्च १४, इ. स. १८२३ – मार्च १३, इ. स. १८९१) हा फ्रेंच साहित्यिक होता.	थॉडेर द बॅनव्हिल (मार्च १४, इ. स. १८२३ – मार्च १३, इ. स. १८९१) हा फ्रेंच लेखक होता.
te	generate Telugu <S> Sushmita Sen <R> date of birth <O> 19 November 1975 <R> place of birth <O> Hyderabad <T> introduction	ఈమె 1975 నవంబరు 19 న హైదరాబాదులో జన్మించింది.	సుష్మితా సేన్ 1975 నవంబరు 19 న హైదరాబాదులో జన్మించింది.
ta	generate Tamil <S> Kirti Kumari <R> member of political party <O> Bharatiya Janata Party <R> date of birth <O> 13 August 1967 <R> date of death <O> 28 August 2017 <R> occupation <O> politician <R> country of citizenship <O> India <T> introduction	கீர்த்தி குமாரி (13 ஆகத்து 1967 - 28 ஆகத்து 2017) பாரதீய ஜனதா கட்சியின் இந்திய அரசியல்வாதி ஆவார்.	கீர்த்தி குமாரி (13 ஆகத்து 1967 - 28 ஆகத்து 2017) ஓர் இந்திய அரசியல்வாதியும், பாரதிய ஜனதா கட்சியின் முன்னாள் சட்டமன்ற உறுப்பினரும் ஆவார்.
kn	generate Kannada <S> Barry C. Barish <R> award received <O> Henry Draper Medal <R> point in time <O> 2017 <T> awards and honors	ಮತ್ತು ಬ್ಯಾರಿಷ್ ಅವರಿಗೆ ೨೦೧೭ ರ ಹೆನ್ರಿ ಡ್ರೇಪರ್ ಪದಕವನ್ನು ನೀಡಲಾಯಿತು.	೨೦೧೭ ರಲ್ಲಿ ಅವರು ಹೆನ್ರಿ ಡ್ರೆಪರ್ ಪದಕವನ್ನು ಪಡೆದರು.
bn	generate Bengali <S> Jim Pothecary <R> member of sports team <O> South Africa national cricket team <R> occupation <O> cricketer <T> introduction	দক্ষিণ আফ্রিকা ক্রিকেট দলের অন্যতম সদস্য ছিলেন তিনি ।	দক্ষিণ আফ্রিকা ক্রিকেট দলের অন্যতম সদস্য ছিলেন তিনি ।
gu	generate Gujarati <S> Krishnalal Shridharani <R> date of birth <O> 16 September 1911 <R> date of death <O> 23 July 1960 <R> occupation <O> poet <R> occupation <O> playwright <R> languages spoken, written or signed <O> Gujarati <T> introduction	કૃષ્ણલાલ શ્રીધરાણી (૧૬ સપ્ટેમ્બર ૧૯૧૧ – ૨૩ જુલાઇ ૧૯૬૦) ગુજરાતી ભાષાના કવિ અને નાટ્યકાર હતા.	કૃષ્ણલાલ શ્રીધરાણી (૧૬ સપ્ટેમ્બર ૧૯૧૧ – ૨૩ જુલાઇ ૧૯૬૦) ગુજરાતી કવિ, નાટ્યકાર અને નાટ્યકાર હતા.
pa	generate Punjabi <S> Orhan Pamuk <R>award received <O> Nobel Prize in Literature <R> point in time <O> 2006 <R> date of birth <O> 07 June 1952 <R> occupation <O> novelist <R> languages spoken, written or signed <O> Turkish <T> introduction	ਓਰਹਾਨ ਪਾਮੋਕ (ਜਨਮ 7 ਜੂਨ 1952) ਇੱਕ ਤੁਰਕੀ ਨਾਵਲਕਾਰ ਹੈ ਜਿਸ ਨੇ 2006 ਵਿੱਚ ਸਾਹਿਤ ਲਈ ਨੋਬਲ ਇਨਾਮ ਹਾਸਿਲ ਕੀਤਾ.	ਓਰਹਾਨ ਪਾਮੋਕ (ਜਨਮ 7 ਜੂਨ 1952) ਇੱਕ ਤੁਰਕੀ ਨਾਵਲਕਾਰ ਹੈ ਜਿਸ ਨੂੰ 2006 ਵਿੱਚ ਸਾਹਿਤ ਲਈ ਨੋਬਲ ਪੁਰਸਕਾਰ ਨਾਲ ਸਨਮਾਨਿਤ ਕੀਤਾ ਗਿਆ .
ml	generate Malayalam <S> Naomi Scott <R> date of birth <O> 06 May 1993 <R> place of birth <O> London <R> country of citizenship <O> United Kingdom <T> introduction	1993 മെയ് 6 ന് ഇംഗ്ലണ്ടിലെ ലണ്ടനിലാണ് സ്കോട്ട് ജനിച്ചത്.	1993 മെയ് 6 ന് ഇംഗ്ലണ്ടിലെ ലണ്ടനിലാണ് സ്കോട്ട് ജനിച്ചത്.
or	generate Odia <S> Ajay Swain <R> award received <O> Odisha Sahitya Akademi Award <R> point in time <O> 2012 <T> introduction	ସେ ୨୦୧୨ ମସିହାରେ ଓଡ଼ିଶା ସାହିତ୍ୟ ଏକାଡେମୀ ପୁରସ୍କାର ଲାଭ କରିଥିଲେ।	୨୦୧୨ ମସିହାରେ ସେ ଓଡ଼ିଶା ସାହିତ୍ୟ ଏକାଡେମୀ ପୁରସ୍କାର ଲାଭ କରିଥିଲେ।
as	generate Assamese <S> Harishankar Parsai <R> date of death <O> 10 August 1995 <R> date of birth <O> 22 August 1922 <R> occupation <O> writer <R> country of citizenship <O> British India <R> country of citizenship <O> Dominion of India <R> occupation <O> author <T> introduction	হৰিশংকৰ পৰসাঈ (২২ আগষ্ট, ১৯২৪ - ১০ আগষ্ট, ১৯৯৫) আছিল হিন্দী সাহিত্যৰ এগৰাকী প্রসিদ্ধ লেখক আৰু ব্যংগকাৰ ।	হৰিশংকৰ পৰসাঈ (২২ আগষ্ট, ১৯২২ - ১০ আগষ্ট, ১৯৯৫) এজন ভাৰতীয় লেখক ।

Table 5.5: Examples of generation

Chapter 6

Exploring Techniques for Generating Cross Lingual Factually Grounded Long Text

6.1 Overview

In the previous chapters we have explained the creation of the XAlign dataset along with its use for the task of fact extraction and fact to text generation. However, the fact to text generation procedure described so far suffers from the limitation of being able to generate only a single sentence at one time. Because of this limitation, significant manual efforts are needed when aiming to generate complete articles. The fact to text generation system demands that all facts about an entity to be clustered together into logical groups such that each group represents a sentence. Since the XF2T systems are trained to generate shorter pieces of text, multiple problems like hallucination, repetition of information etc arise when attempting to use these models with a lot of input facts.

In order to mitigate these limitations and automate the process of complete article generation, we introduce the task of Fact to long text generation. This task involves taking as input, all facts about a particular entity and the output is a paragraph in another target language which is expected to capture all the semantic information in English facts without hallucination. The solution is also expected to group related semantic information from facts into coherent sentences which appear in an appropriate order with smooth transitions. This chapter discusses in detail, the process of clustering the facts into logical groups and then performing text generation on top of them. Since our analysis from the previous chapters resulted in the realisation that one of the significant problems with the XF2T systems is hallucination, we make special efforts during our generation techniques in order to tackle hallucination and generate more faithful and grounded sentences. Figure 6.1 provides an example of the task discussed in this chapter.

Cross-lingual fact to *long* text (XFLT) systems could be useful across several business domains like healthcare, sports, travel, education, and reporting. In healthcare, English medical

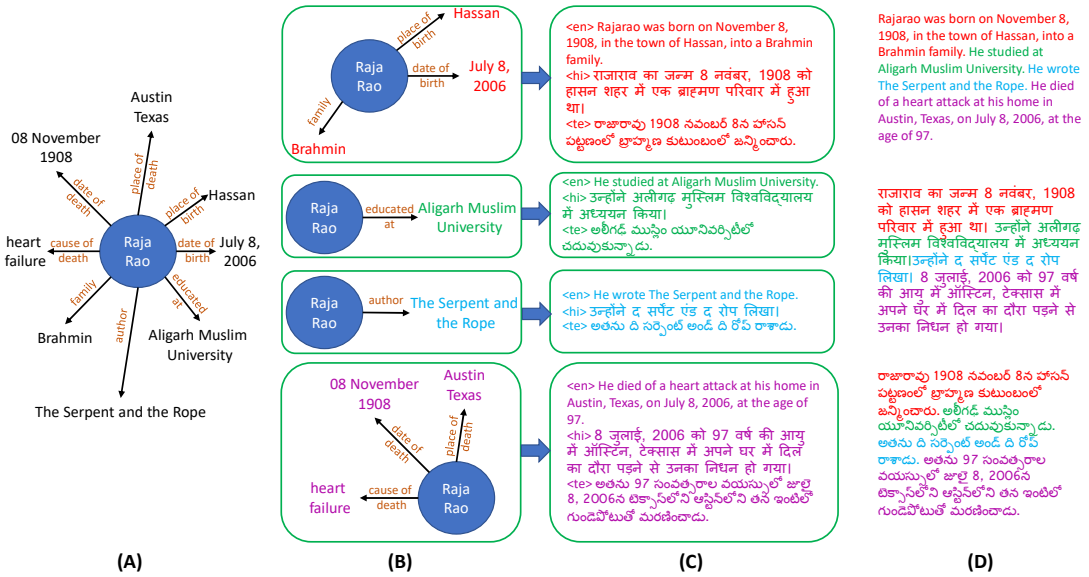

Figure 6.1: XFLT example: Generating English, Hindi and Telugu paragraphs to capture semantics from English facts

records can be used to generate patient summaries in regional languages. Drug information leaflets can be curated in different languages from English ingredients and effects. Summary of health insurance policies can be generated in different languages from English terms and conditions. English facts and warnings can be used to create public health alerts and advisories in different languages. Similarly, in sports, English statistics about events and players can be used to compose match reports, sports news, athlete biographies, and sports history essays in different languages. In tourism and travel, XFLT tools could help generate travel guides, hotel reviews, travel itinerary summary, travel blogs, travel advisories, travel-related news across languages given English facts.

Only 10% of the sentences in the dataset have complete coverage with respect to their corresponding facts. Leveraging, such a dataset for cross-lingual fact to *long* text (XFLT) brings its own challenges. Lastly, while there exist source-dependent metrics like BLEURT [69] and PARENT [20], they are defined only for monolingual scenarios where input and output are in the same language. How do we define a similar source-dependent metric for our cross-lingual setting? Since we are dealing with a dataset where the reference text might be diverging from the input facts (, we also introduce a new evaluation metric - XPARENT, specifically designed for handling diverging references in the domain of XF2T.

Overall, through this chapter we highlight the following contributions.

- We propose a novel problem: Cross-lingual fact to long text generation (XFLT).

- We propose a modular approach which uses coverage prompts and grounded decoding to reduce hallucination and deep reinforcement learning to improve quality.

- Our best model achieves a BLEU of 23 and cross-lingual PARENT score of 56. We make our code and data publicly available[1].

The remainder of the chapter is organized as follows. We discuss the details of the dataset and its reaction in Section 6.2. We discuss details of the two modules of our proposed system in Section 6.3. We discuss experiments and results in Section 6.4. Finally we conclude with a brief summary in Section 6.5.

6.2 Dataset

We derive our dataset, XLAlign, from the existing dataset, XAlignV2 [66] (which is a revised version of XAlign [1]). The details of the construction of these datasets can be found in Chapter 3. Example pairs corresponding to the same entity from XAlignV2 are combined to obtain example (English facts, target language paragraph) pairs for our dataset, XLAlign. The combination is done by a union of the English facts of corresponding XAlignV2 examples, and a concatenation of sentences as per their order in the original Wikipedia article to create multi-sentence descriptions. In total, the XLAlign dataset contains 125,106 paragraphs across 12 different languages. This is summarized in Table 6.1 which shows average number of facts, sentences, words per instance and instance counts in the train, validation, test splits. Compared to existing cross-lingual fact to short text datasets which contain one sentence per example, XLAlign contains 2.9 sentences and 47.7 words on average.

Each example in the overall dataset has two properties.

- Degree of Sentence-level Coherence: Since all the sentences for an entity in XAlignV2 may not be present contiguously in the source article, there is a variance in the coherence levels of the corresponding paragraph in XLAlign created by concatenation of such potentially non-contiguous sentences. Since no classifier for predicting coherence exists for Indian languages, with an aim to measure coherence levels of the corresponding paragraph in XLAlign, a coherence classifier (transfer-learned from pretrained MuRIL) was trained using the next sentence prediction task. Sentence pairs were extracted from featured Wikipedia articles, with contiguous sentence pairs chosen as positive samples and randomly permuted sentence pairs as negative samples. This classifier leads to a F1 of 0.71. Fig. 6.2 shows the variation of coherence across samples in XLAlign. Note that several examples in the dataset have one sentence paragraphs which have a default sentence-level coherence of 1.

[1] `https://tinyurl.com/CrossLingual-FLT`

Language	Instance Counts			Avg	Avg	Avg
	Train	Val	Test	#Facts	#Sents	#Words
Assamese (as)	799	159	111	7.0	4.3	66.9
Bengali (bn)	14,858	2,968	1,984	7.5	3.8	59.0
English (en)	32,176	6,427	4,292	5.3	2.4	41.2
Gujarati (gu)	901	179	121	6.0	3.3	55.6
Hindi (hi)	9,266	1,850	1,239	5.2	2.6	51.9
Kannada (kn)	2,026	404	273	6.6	3.7	51.1
Malayalam (ml)	8,363	1,671	1,117	6.0	3.2	40.4
Marathi (mr)	5,394	1,077	722	4.5	2.0	31.6
Odia (or)	1,742	348	237	6.9	4.1	63.0
Punjabi (pa)	5,454	1,085	731	6.5	3.1	84.1
Tamil (ta)	10,026	2,004	1,340	4.8	2.8	37.1
Telugu (te)	2,820	563	379	6.2	3.7	46.3
All	93,825	18,735	12,546	5.8	2.9	47.7

Table 6.1: Dataset statistics for the XLAlign dataset.

- Degree of alignment: XAlignV2 contains examples with varying level of alignment between English facts and labeled target language sentences. This means that some semantics in the sentence is not captured by the corresponding facts. In order to quantify this partial alignment, we use scores from the coverage classifier described in Section 6.3.2.1 and illustrated in Fig. 6.2. This classifier was trained on binary labels obtained for 4376 examples. The classifier leads to a micro-averaged F1 of 0.9.

We split the dataset into train:validation:test in the ratio 75:15:10 as follows. To create a high-quality test and validation sets, the examples in XLAlign were partitioned such that in the test and validation set, the ground truth target language paragraph is coherent and contains least amount of extra information which is not covered by corresponding English facts. The train, validation and test split for each of the languages was also stratified based on the number of sentences per entity in the ground truth so that each of the splits contains equal proportion of paragraphs of different lengths.

Fig. 6.3 and 6.4 show the distribution of number of facts and sentences respectively across various languages in the XLAlign dataset. Note that the dataset contains sizeable number of instances across various languages. Also, while creating the dataset we ensured that the number of sentences per example is limited to a maximum of 10 which leads to $\sim$1.6% examples with 20+ facts.

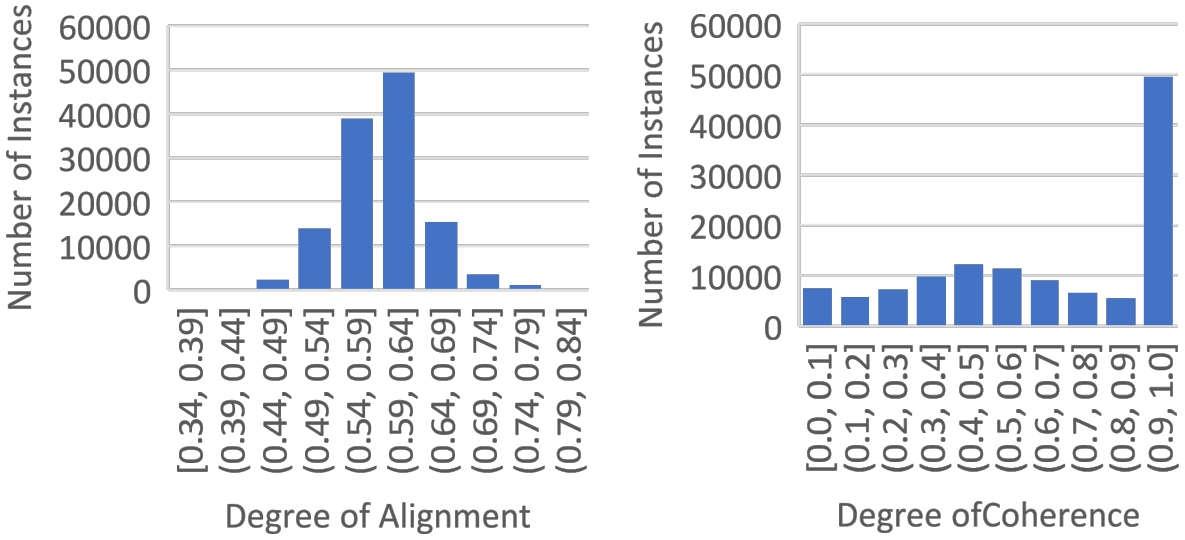

Figure 6.2: Distribution of degree of alignment and degree of coherence across dataset instances in XLAlign

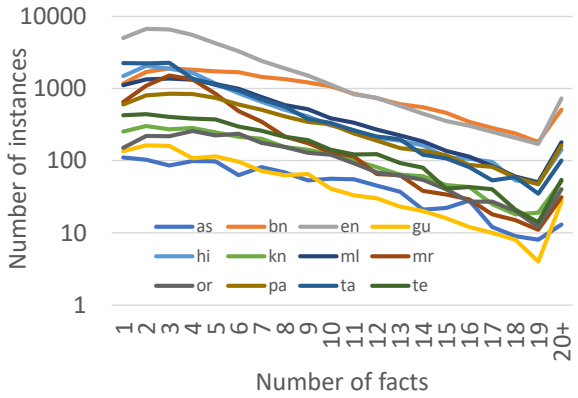

Figure 6.3: Distribution of number of facts across various languages in the XLAlign dataset

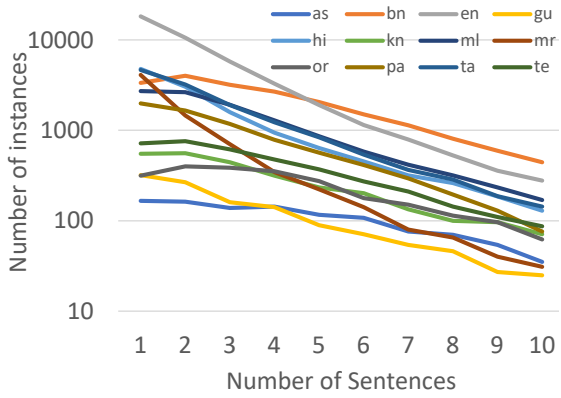

Figure 6.4: FDistribution of number of sentences across various languages in the XLAlign dataset

6.3 The Proposed Cross Lingual Fact to Long Text Generation System

Our dataset D containing N instances can be represented as $D = \{F_i, T_i, l_i\}_{i=1}^{N}$ where each instance D_i contains a set of $|F_i|$ English facts $F_i = \{f_j\}_{j=1}^{|F_i|}$ and an ordered list of aligned $|T_i|$ target sentences $T_i = [t_k]_{k=1}^{|T_i|}$ in the desired language l_i. A fact f_j is a tuple

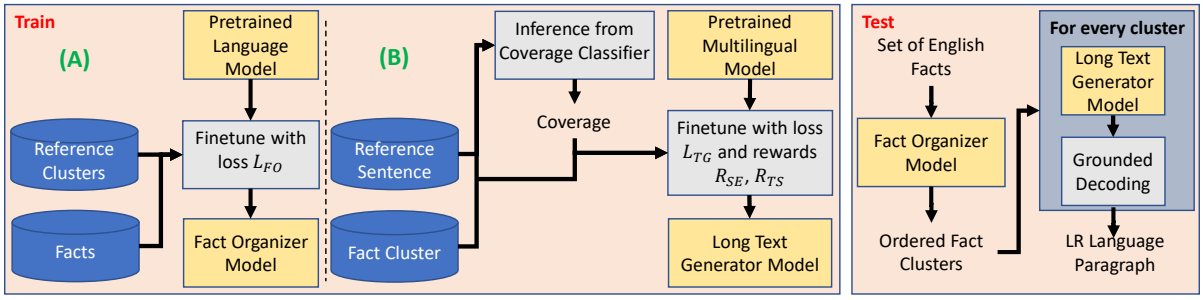

Figure 6.5: Proposed pipeline for cross-lingual fact to long text generation. Training involves finetuning (A) Fact Organizer Model and (B) Long Text Generation Model.

composed of subject s_j, relation r_j, object o_j and m qualifiers $Q = q_1, q_2, \ldots, q_m$. Each qualifier provides more information about the fact. Each of the qualifiers $\{q_j\}_{j=1}^{m}$ can be linked to the fact using a fact-level property which we call as qualifier relation qr_j. For example, consider the sentence: "Narendra Modi was the Chief Minister of Gujarat from 7 October 2001 to 22 May 2014, preceded by Keshubhai Patel and succeeded by Anandiben Patel." This can be represented by a fact where subject is "Narendra Modi", relation is "position held", object is "Chief Minister of Gujarat" and there are 4 qualifiers each with their qualifier relations as follows: (1) q_1="7 October 2001", qr_1="start time", (2) q_2="22 May 2014", qr_2="end time", (3) q_3="Keshubhai Patel", qr_3="replaces", and (4) q_4="Anandiben Patel", qr_4="replaced by". Further, the alignment between every target sentence t_k and set of English facts f_j is also provided as part of the dataset. We represent the aligned set of facts for target sentence t_k by $A(t_k)$.

Given the dataset D, with partially aligned cross-lingual facts and sentences, our approach consists of two main modules: fact organizer and long text generator. Fact organizer clusters facts into logical groups and also predicts a sequence order over these groups. The long text generator is a multilingual Transformer-based encoder-decoder model with the following training recipe. The coverage prompts and grounded decoding tricks help us address the hallucination problem to a significant extent. Further, we obtain better quality output with deep reinforcement learning (RL) using task-specific reward functions which motivate the model to generate outputs which are (a) syntactically aligned to ground truth output and (b) semantically aligned to input English facts. Fig. 6.5 shows the broad architecture of our proposed pipeline. We discuss details of these modules in this section.

6.3.1 Fact Organizer Training

For every instance $D_i \in D$, fact organizer clusters its facts $\{f_j\}_{j=1}^{|F_i|}$ into an ordered list of logical groups $G_i = g_1, g_2, \ldots, g_{|G_i|}$. Facts that align with a target sentence t_k, i.e., $A(t_k)$ should belong to the same logical group. Thus, ideally, there should be a logical group corresponding

to each target sentence, i.e., $|G_i| = |T_i|$. Each logical group can consist of different number of facts. Also, each fact can belong to multiple logical groups.

We use an English Transformer-based encoder-decoder pretrained model for modeling the fact organizer. Each fact f_j is encoded as a string and the overall input consists of a concatenation of such strings across all facts in F_i. The string representation for a fact f_j is "$\langle S\rangle s_j \langle R\rangle r_j \langle O\rangle o_j \langle R\rangle qr_{j_1} \langle O\rangle q_{j_1} \langle R\rangle qr_{j_2} \langle O\rangle q_{j_2} \ldots \langle R\rangle qr_{j_m} \langle O\rangle q_{j_m}$" where $\langle S\rangle$, $\langle R\rangle$, $\langle O\rangle$ are special tokens. The overall input with F_i facts is obtained as follows: "cluster: f_1 f_2 ... $f_{|F_i|}$". The overall output with G_i logical groups is obtained as follows: "$g_1 \langle BR\rangle g_2 \langle BR\rangle \ldots \langle BR\rangle g_{|G_i|}$" where each group g is a concatenation of constituent facts. Overall, the model is trained using the standard categorical cross-entropy loss L_{FO}.

The grouping of facts and the order in which these groups appear in the text is used as input for the long text generation.

6.3.2 Long Text Generator Training

The long text generator is a multilingual Transformer-based encoder-decoder model with the following training recipe. It uses coverage prompts to address the partially aligned nature of the training data. Further, it uses RL based training with reward functions to encourage grounded generations.

6.3.2.1 Coverage prompts to Reduce Hallucination

Ideally, in every instance of the dataset D, each target sentence t_k should contain the same semantic information as in its aligned set of facts $A(t_k)$. But practically, the set of aligned facts $A(t_k)$ may not cover the entire semantics of the target sentence t_k. We refer to this problem as partially aligned nature of the labeled data. If we train on such partially aligned data, the long text generator is encouraged to generate extraneous information beyond the semantics present in the input facts, leading to hallucination.

To address this problem, we first train a coverage classifier that estimates the degree to which the set of aligned facts $A(t_k)$ cover the semantics of the target sentence t_k. To train this classifier, we obtain coverage annotations for a part D_{cov} of the dataset D. Each target sentence t_k for every instance in D_{cov} is labeled with one of the two classes: complete coverage or partial coverage. The coverage classifier is a multilingual Transformer-based encoder with a classifier head which takes t_k and a string representation of $A(t_k)$ separated by a [SEP] token. Based on a threshold applied on confidence score with which the classifier predicts a fact-reference pair as completely aligned, we determine a coverage class (one of low, medium or high) for each of our training samples such that there are equal number of training instances for each of the classes per language.

While training the long text generator, we also incorporate the predicted coverage class as part of the input. Each training instance for long text generator model consists of a sentence t_k across all samples from D. At train time, we use the ground truth set of English facts aligned with t_k as input rather than using logical groups obtained from fact organizer. Overall, the input format for the long text generator is "generate l_i c_{ik}:" followed by a linearized string of facts in $A(t_k)$, where l_i is the target language of the sentence t_k and c_{ik} is the coverage class predicted using the coverage classifier. The long text generator is trained using the standard categorical cross-entropy loss L_{TG}. At inference time, we expect to generate sentences with high coverage and hence, we pass c always as "High" at inference time.

6.3.2.2 Reinforcement Learning for Improved Generation Quality

Further, we obtain better quality output with deep reinforcement learning using task-specific reward functions which motivate the model to generate outputs which are (a) syntactically aligned to ground truth output and (b) semantically aligned to input English facts.

Source Entailment Reward (R_{SE}): Given an instance with input as $A(t_k)$ and reference text t_k, source entailment reward measures the semantic similarity between the generated text and source English facts $A(t_k)$. The English fact tokens are not directly comparable with generated target language tokens. To bridge this gap, we introduce the notion of entailment probability, which is based on the probabilities that the presence of ngrams in the generated text is "correct" given the associated English facts. Estimating this probability is in itself a challenging language understanding task. Let y_k be the generated sentence text. Let y_k^n denote the list of all ngrams of y_k of order n. Let b denote one of such ngrams. Further, consider every token w in an ngram b. First, we compute entailment probability of token w being entailed by the source as the maximum of its probabilities of being entailed by each lexical item (subject, relation, object, or qualifier) v of a fact in the source.

$$P(w \impliedby A(t_k)) = \max_{v \in A(t_K)} P(w \impliedby v) \tag{6.1}$$

where $P(w \impliedby v)$ is estimated by using similarity scores from MuRIL embeddings of the token w and lexical item v. Using this, we compute the entailment probability of ngram b being entailed as the geometric average of entailment probabilities of each of the constituent tokens as follows.

$$P(b \impliedby A(t_k)) = \left(\prod_{w \in b} P(w \impliedby A(t_k)) \right)^{1/|b|} \tag{6.2}$$

where $|b|$ is the order of the ngram b. Lastly, entailment score of generated sentence y_k for ngrams of order n with respect to the aligned ground truth facts is obtained by taking mean of entailment probabilities of each of the constituent ngrams as follows.

$$ES^n(y_k, A(t_k)) = \frac{\sum_{b \in y_k^n} (P(n \impliedby A(t_k)))}{|y_k^n|} \tag{6.3}$$

where $|y_k^n|$ denotes the number of ngrams in y_k^n. Lastly, entailment score $ES(y_k, A(t_k))$ of generated sentence y_k with respect to the aligned ground truth facts is obtained by taking geometric mean of $ES^n(y_k, A(t_k))$ across all orders. The final source entailment reward is given by $R_{SE} = \lambda_{SE} \times ES(y_k, A(t_k))$ where λ_{SE} is a tunable hyperparameter controlling the importance of this reward in the overall objective to be optimized.

Target Similarity Reward (R_{TS}): This measures the syntactic similarity between the generated text y_k and reference text t_k. We measure this similarity using the BLEU metric. Thus, $R_{TS} = \lambda_{TS} \times BLEU(y_k, t_k)$ where λ_{TS} is a tunable hyperparameter controlling the importance of this reward in the overall objective to be optimized.

The rewards are used for policy learning. We employ the policy gradient algorithm [79] to maximize the expected reward (source entailment and/or target similarity) of the generated sequence y_k, whose gradient with respect to the parameters ϕ of the neural network model is estimated by sampling as follows.

$$\Delta_\phi J(\phi) = E[R.\Delta_\phi \log(P(y_k|x;\phi))] \tag{6.4}$$

where R is the R_{SE} reward and/or the R_{TS} reward, y_k is sampled from the distribution of model outputs at each decoding time step, x (which includes $A(t_k)$, language ID l_i and the coverage prompt) is the input to the model, and ϕ are the parameters of the long text generation model. The overall objectives for ϕ are the loss of the base model L_{TG} and the policy gradient of the different rewards.

6.3.3 Grounded Decoding during Inference

To reduce hallucination, at inference time, we use a decoding strategy that reduces the generation of text that is unsupported by the source, similar to [73]. This is based on the intuition that every word generated by the model should be entailed by the source facts, as long as the word captures some semantics from the source facts. Wrongly associating a content phrase (e.g. France) to the language model, simply because it seems more fluent (e.g. Paris France is fluent), might be a major cause of hallucination; since the facts may be discussing about the city of Paris in Texas, USA.

We encode this intuition in the decoding process as follows. At time t, while decoding the text y_k, we choose the top k tokens w based on their language modeling probabilities $P(w|y_{k[1:t-1]}, x; \phi)$. For each of these tokens w, we compute entailment probabilities $P(w \impliedby A(t_k))$ using Eq. 6.1. Then, we perform beam search using a combination of these two probabilities as follows: $P(w|y_{k[1:t-1]}, x; \phi) \times P(w \impliedby A(t_k))^{\lambda_{EF}}$ instead of just using the original language modeling probabilities.

6.3.4 Overall XFLT Inference

To summarize, the overall inference pipeline of our proposed system for XFLT works as follows. Given a set of English facts F_i for the i-th test instance, our fact organizer model outputs ordered fact clusters $G_i = g_1, g_2, \ldots, g_{|G_i|}$. Each fact cluster $\{g_k\}_{k=1}^{|G_i|}$ is then processed individually by our long text generator module along with grounded decoding to generate the output sentence y_k. Finally, these sentences are concatenated to generate the prediction paragraph $Y_i = \mathit{concat}(y_1, y_2, \ldots, y_k)$. Hyper-parameter details of various methods are in the Appendix.

6.4 Experiments and Results

6.4.1 Metrics

We use two standard natural language generation metrics: BLEU [55][2] and chrF++ [57]. But these metrics rely on the reference text. This is problematic because in XFLT, the reference and the source do not align entirely, i.e., the reference text may have extra information not specifically mentioned in the input text. Hence, a source-dependent metric is suitable for XFLT. Further, since the task involves cross-lingual modeling, we propose XPARENT, which is a modified version of PARENT adapted for cross-lingual settings.

Given generated text y, target reference text t and corresponding source facts $A(t)$, we define XPARENT$(y, t, A(t))$ as the F1 score (or harmonic mean) of entailed precision (EP) and entailed recall (ER) which in turn are defined as follows.

Entailed precision (EP) is computed as geometric average of entailed precision EP^n for ngrams of order n=1 to n=4. EP^n is further calculated as follows. Let y^n and t^n denote the list of all ngrams of order n of y and t respectively. Let b denote one of such ngrams in y^n. We consider the ngram b to be correct either if it occurs in the reference t, or if it has a high probability of being entailed by the source facts $A(t)$. Let $P(b \in t^n) = \min(\#(b, y^n), \#(b, t^n))/\#(b, y^n)$ where $\#(b, \circ)$ indicates number of times b occurs in $\circ$. Entailed precision EP^n for ngrams of order n is given by:

$$EP^n = \frac{\sum_{b \in y^n} [[P(b \in t^n) + P(b \notin t^n) P(b \impliedby A(t))] \times \#(b, y^n)]}{\sum_{b \in y^n} \#(b, y^n)} \tag{6.5}$$

In words, an ngram receives a reward of 1 if it appears in the reference, with probability $P(b \in t^n)$, and otherwise it receives a reward of $P(b \impliedby A(t))$ which is computed using Eq. 6.2. Both numerator and denominator are weighted by the count of the ngram in y^n. $P(b \in t^n)$ rewards an ngram for appearing as many times as it appears in the reference, not more.

[2]Specifically, we use the implementation provided at `https://github.com/mjpost/sacrebleu`

Entailed recall (ER) is computed against both the reference ($ER(t)$), to ensure proper sentence structure in the generated text, and the input facts ($ER(A(t))$), to ensure that texts which mention more information from the facts get higher scores. These are combined using a geometric average as follows.

$$ER = ER(t)^{\lambda_R} ER(A(t))^{1-\lambda_R} \tag{6.6}$$

The parameter λ_R trades-off how much the generated text should match the reference, versus how much it should cover information from the facts.

Entailed recall $ER(t)$ with respect to reference t is computed as geometric average of $ER^n(t)$ for ngrams of order n=1 to n=4. We compute $ER^n(t)$ as follows.

$$ER(t) = \frac{\sum_{b \in t^n} [\min(\#(b, y^n), \#(b, t^n)) P(b \impliedby A(t))]}{\sum_{b \in t^n} [\#(b, t^n) P(b \impliedby A(t))]} \tag{6.7}$$

Entailed recall $ER(A(t))$ with respect to source facts $A(t)$ is computed at a word level as follows.

$$ER(A(t)) = \frac{\sum_{w \in A(t)} [I[P(w \impliedby y) > \tau] \times \#(w, A(t))]}{\sum_{w \in A(t)} \#(w, A(t))} \tag{6.8}$$

where τ is a threshold tuned by manual inspection, w is a unique word in the concatenated string representation of facts in $A(t)$, $I[c]$ is the indicator function which takes a value of 1 if the condition c is true, else 0, and $P(w \impliedby y)$ is computed using Eq. 6.1.

6.4.2 Fact Organizer Quality Evaluation

For our fact organizer, we use mT5-small. It provides a micro-F1 score of 0.595 and an MSE of 1.28 on average for prediction of the number of logical groups. For comparison, we also trained a MuRIL-base multi-class classifier to predict number of logical groups on XLAlign train set using categorical cross-entropy loss. This method provides much lower micro-F1 score of 0.245 and an MSE of 4.67 . Further, Fig. 6.6 shows the heatmap comparing actual versus predicted number of logical groups using the proposed fact organizer (left) and MuRIL-base classifier (right). From the heatmap as well as the micro-F1 and MSE values it is clear that a MuRIL-base classifier is poor at predicting the number of clusters.

Further, we wished to evaluate the quality of the discovered clusters using our fact organizer. We compute the quality as follows. First, given the discovered clusters and ground truth clusters, we compute 1:1 correspondence between them by modeling this as a linear sum assignment problem[3] and solve it using the Hungarian Method [39]. If number of discovered clusters is different from the number of ground truth clusters, the extra clusters on either side remain unassigned. Post the assignment, one can measure accuracy based on number of data points

[3] `https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.linear_sum_assignment.html`

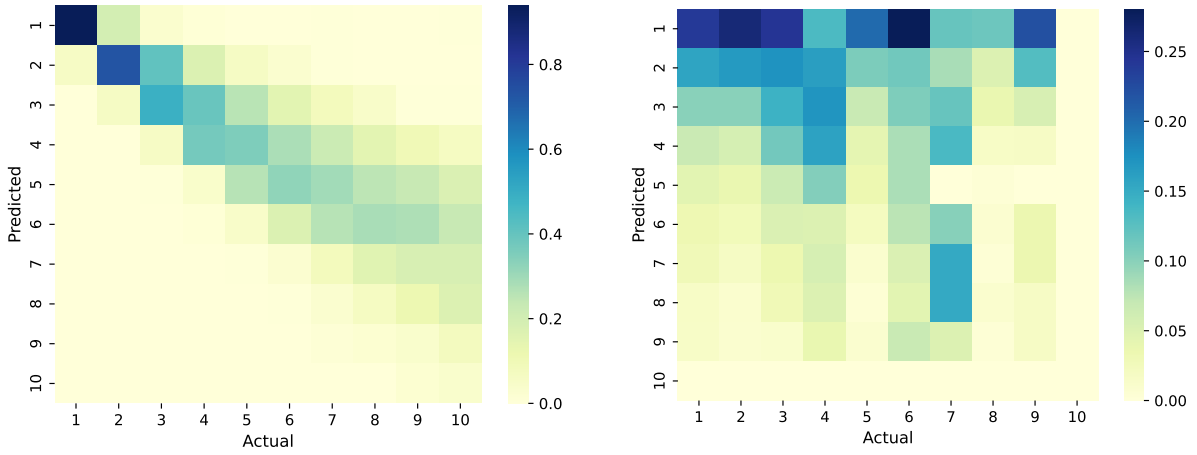

Figure 6.6: Heatmap comparing actual versus predicted number of logical groups using the proposed fact organizer(left) and MuRIL-base classifier(right).

accurately clustered compared to ground truth. For our fact organizer, the average accuracy across test instances with $\geq$2 sentences turns out to be 81.49% which implies that our fact organizer is extremely effective at clustering facts into the expected logical groups.

Lastly, our fact organizer is also responsible for ordering the logical groups. To measure the quality of this ordering of logical groups, we can compare with the ground truth ordering of sentences. We perform this comparison using Kendall rank correlation coefficient (τ) [34] which is in the range [0,1] – higher the better. We find that the average Kendall-τ across test instances with $\geq$2 sentences turns out to be 0.696. This implies that our fact organizer not just discovers the right clusters but also sequences them in the expected order effectively.

6.4.3 Long Text Generator Quality Evaluation

For the long text generation, we use pretrained mT5-small as the base model architecture.

Baselines: Our work is closest to Cross-Lingual Fact to Short Text (XFST) methods. Hence, we compare our proposed method with two baseline approaches both of which also use the same base model architecture: Single-Sentence XFST and Multi-Sentence XFST. Multi-Sentence XFST is finetuned on XLAlign dataset where the input consists of a large number of English facts and the model is trained to generate multiple native language sentences. For training Single-sentence XFST model, we first split each instance in XLAlign train set such that each instance in the split dataset contains one native language sentence paired with the correspondence set of English facts. Single-Sentence XFST is then finetuned on this split dataset.

Ablations: Our full proposed method (Fact Organizer+CP+RL+GD) consists of several components: mT5 for clustering, coverage prompts, RL for improved generation quality and

Lang	Single-Sentence XFST						Fact Organizer+CP+RL+GD					
	All Test Instances			Test Instances with ≥2 sentences			All Test Instances			Test Instances with ≥2 sentences		
	BLEU	chrF++	XPARENT	BLEU	chrF++	XPARENT	BLEU	chrF++	XPARENT	BLEU	chrF++	XPARENT
as	5.092	34.406	26.786	5.035	34.062	26.613	8.119	43.359	40.311	7.232	43.538	41.362
bn	16.456	51.106	43.501	16.230	50.815	42.506	25.216	58.769	62.993	22.645	58.710	62.495
en	22.211	50.862	56.545	19.578	49.263	54.245	30.647	53.916	68.670	25.703	52.771	67.574
gu	6.621	32.977	29.204	6.109	32.454	28.235	13.598	40.644	43.824	10.578	39.945	45.501
hi	14.544	44.457	43.320	16.504	44.631	41.274	25.951	48.260	58.999	20.972	47.214	58.461
kn	4.280	31.220	21.893	4.200	30.769	21.428	7.551	36.216	39.051	6.426	36.141	40.650
ml	6.550	37.892	24.741	6.724	37.479	24.342	10.507	41.386	37.125	9.113	41.284	39.084
mr	22.529	41.051	40.656	12.057	33.124	32.993	29.859	51.130	56.449	18.502	45.948	51.947
or	17.632	52.457	42.941	18.114	52.218	42.990	26.598	60.014	50.528	26.848	60.352	52.334
pa	10.939	35.286	37.206	10.062	34.522	35.458	15.837	39.778	52.493	12.220	39.276	50.600
ta	6.637	42.681	22.951	5.850	41.774	21.592	11.912	44.941	36.687	9.124	45.140	37.933
te	3.863	29.620	24.246	4.118	29.391	23.887	8.488	39.591	38.409	7.112	39.465	40.101
All	15.515	45.410	42.202	14.059	44.171	40.301	23.010	50.142	56.555	19.036	49.318	56.132

Table 6.2: Language-wise Performance Comparison of the baseline XFST method and our proposed method.

	All Test Instances			Test Instances with ≥2 sentences		
	BLEU	chrF++	XPARENT	BLEU	chrF++	XPARENT
Single-Sentence XFST [1, 52]	15.515	45.410	42.202	14.059	44.171	40.301
Multi-Sentence XFST	18.660	37.621	50.338	15.873	37.067	50.327
Fact Organizer+Single-Sentence XFST	20.395	44.136	52.679	18.227	43.366	52.628
Fact Organizer+CP	22.060	48.821	55.271	18.442	48.119	55.074
Fact Organizer+CP+RL	22.663	49.532	55.328	18.760	48.717	54.966
Fact Organizer+CP+RL+GD	**23.010**	**50.142**	**56.555**	**19.036**	**49.318**	**56.132**

Table 6.3: Performance Comparison of various methods for XFLT task.

grounded decoding. To evaluate the importance of each component, we evaluate multiple ablations as follows:

- Fact Organizer+Single-Sentence XFST: Coverage prompts, RL for improved generation quality and grounded decoding are removed.
- Fact Organizer+CP: RL for improved generation quality and grounded decoding are removed.
- Fact Organizer+CP+RL: Grounded decoding is removed.

Main Results: Table 6.3 shows performance comparison between the baselines, our proposed method and its ablations, on the XLAlign test set. We show BLEU, chrF++ and XPARENT for two settings: all test instances, and test instances with $\geq$ 2 sentences. While "all test

	Punjabi			English			Hindi			Marathi			Telugu		
	F	R	C	F	R	C	F	R	C	F	R	C	F	R	C
Ours	53	65	64	42	33	31	46	45	52	42	55	59	21	54	68
Multi-Sentence XFST	31	19	15	26	15	19	35	35	35	29	30	31	53	19	8
Both equal	16	16	22	32	52	50	19	21	13	29	15	10	26	27	24

Table 6.4: Human Evaluation: Percent times each method was preferred when compared to Multi-Sentence XFST baseline. F=Fidelity, R=recall, C=coherence.

instances" contain ∼33% instances with one sentence only (and is therefore similar to XFST setting), the "test instances with $\geq$ 2 sentences" is truly an XFLT setting.

We make the following observations from Table 6.3. (1) Results for the "test instances with $\geq$ 2 sentences" setting are typically lower compared to "all test instances" setting as expected. (2) Multi-sentence XFST is better than single-sentence XFST on BLEU and XPARENT. chrF++ is better for single-sentence XFST since its generations are relatively shorter and precise. (3) Fact Organizer helps improve the results for single-sentence XFST by a large margin. (4) Finetuning mT5 long text generator with coverage prompts leads to gains across all metrics. (5) RL based reward functions make the long text generator training more effective leading to gains across all metrics except XPARENT in the "test instances with $\geq$ 2 sentences" setting. We found that this minor decrease was because of a large decrease in entailed recall against the reference (ER(t)) for Tamil. We see consistent improvements across all metrics when using RL across all other languages. We also tried ablations using the two reward functions one by one, and found that both are needed for best results. (6) Finally, grounded decoding leads to the most accurate model. (7) All improvements for our full method (Fact Organizer+CP+RL+GD) are statistically significant compared to all baselines and ablations as measured using repeated measures ANOVA test with p-value < 0.05.

Language-wise Detailed Results for the Best Method: We show detailed language-wise results for the baseline XFST method and our proposed method (Fact Organizer+CP+RL+GD) on the XLAlign test set in Table 6.2. We observe that (1) Results with our proposed method (Fact Organizer+CP+RL+GD) are drastically better compared to the XFST method clearly showing that XFLT entails unique challenges different from XFST. (2) In the "All Test Instances" setting, BLEU improves relatively by 48.3%. On the other hand, in the "Test Instances with $\geq$2 sentences" setting, XPARENT sees the maximum relative improvement of 39.3%. (3) The biggest relative performance improvements are seen in Telugu, Gujarati and Kannada across metrics. Even in languages where XFST performed well, Fact Organizer+CP+RL+GD improves the metrics improves by >∼1.5x.

6.4.4 Qualitative Results

Human evaluation results: For five languages, we take 100 random test samples, we compare Multi-Sentence XFST baseline and our best proposed method. Table 6.4 shows the preference percentages based on fidelity, recall and coherence. Fidelity captures lack of hallucination. Recall captures how much of the semantics from facts were encoded in the generated output. Coherence (or fluency) assimilates how well the sentences are connected and how smooth is the flow of concepts in the output. We observe that in most cases, outputs from our proposed system are preferred over the best baseline.

Error Analysis: We manually examine 50 examples with low scores using our best method, to analyse the source of possible errors. We found that the most common source was the model repeating a set of words multiple times in a loop. Other sources included missing out facts from the input in the representation and generating extraneous information. Diverging references also lead to lower BLEU and chrF++ scores. Finally, we observed that the model has learned fact association patterns strongly. For example, even if the input facts do not have death cause but just have date of death, the model hallucinates the death cause. Since the model does not have any knowledge about the position of the sentence in the paragraph, in some cases, it generates pronouns in the first sentence and referent nouns in later sentences. This could be solved by passing in relative positional information as part of the model input in the future.

6.4.5 Experiment Setting

To enable better learning from training instances belonging to multiple languages, we perform script unification by transliterating instances from related languages to a representative script. By analyzing vocabulary overlap, we chose three scripts: Roman for English, Malayalam script for Dravidian languages (te, ta, ml, kn), and Devanagiri for the remaining languages.

All experiments were performed on a machine with 4 NVIDIA V100s. Unless otherwise mentioned the hyper-parameters were chosen either based on validation set or tuned based on manual inspection.

For training all of our generative models, we use AdamW optimizer with learning rate of 1e-3 for non-RL methods and 2e-5 for RL methods. These models were trained for a maximum of 30 for non-RL methods, and further 5 epochs for RL methods starting from best checkpoint. Best checkpoint was chosen based on validation loss.

For RL methods, λ_{SE} and λ_{TS} are set to 1. For XPARENT, λ_R is set to 0.5. For grounded decoding, we set λ_{EF} to 0.5.

6.4.6 Examples of Generations using our Best Method

Tables 6.5 6.6 6.7 show the examples of the text generated by our best performing methof for all languages.

Lang.	Source	Reference	Generated
en	generate english high :<H> unnikrishnan puthoor <R> date_of_birth <T> 15 july 1933 <R> date_of_death <T> 02 april 2014 <R> languages_spoken,_written_or_signed <T> malayalam <R> occupation <T> novelist <R> occupation <T> writer <R> award_received <T> kerala sahitya akademi award <R> date_of_death <T> 02 april 2014 <R> place_of_death <T> chavakkad	Unnikrishnan Puthur (15 July 1933 – 2 April 2014) was a Malayalam–language novelist and short story writer . He was awarded Kerala Sahitya Akademi Award for the novel " Balikkallu " in 1968 . He died on 2 April 2014 at a private hospital in Chavakkad.	Unnikrishnan Puthoor (15 July 1933 - 2 April 2014) was an Indian novelist and short story writer in Malayalam . He was awarded the Kerala Sahitya Akademi Award for his contribution to literature . Unnikrishnan Puthoor died on 2 April 2014 in Chavakkad.
hi	generate hindi high : <H> madhav mantri <R> date_of_birth <T> 01 september 1921 <R> date_of_death <T> 23 may 2014 <R> member_of_sports_team <T> india national cricket team <R> occupation <T> cricketer <R> date_of_birth <T> 01 september 1921 <R> place_of_birth <T> nashik <R> member_of_sports_team <T> india national cricket team <QR> start_time <QT> 1951 <QR> end_time <QT> 1955 <R> occupation <T> cricketer <R> cause_of_death <T> myocardial infarction <R> date_of_death <T> 23 may 2014	माधव कृष्णाजी मंत्री (1 सितम्बर 1921 - 23 मई 2014) भारतीय क्रिकेट खिलाड़ी थे जिन्होंने 1951 से 1955 तक चार टेस्ट मैच खेले । मंत्री का जन्म 1 सितम्बर 1921 को महाराष्ट्र के नासिक में हुआ । मंत्री ने भारतीय क्रिकेट टीम के सदस्य के रूप में 1950 के दशक विकेट कीपर और बल्लेबाज के रूप क्रिकेट खेला । 23 मई 2014 को दिल का दौरा पड़ने से उनका निधन हो गया ।	माधव माधव मिश्रा (1 सितंबर 1921 - 23 मई 2014) एक पूर्व भारतीय क्रिकेट खिलाड़ी थे । माधव मण्डी का जन्म 1 सितम्बर 1921 को महाराष्ट्र के नासिक में हुआ था । वह भारतीय क्रिकेट टीम के लिए १९५१ से १९५५ तक खेले थे । 23 मई 2014 को दिल का दौरा पड़ने से उनका निधन हो गया ।
as	generate assamese high : <H> ushadevi bhosle <R> date_of_birth <T> 30 march 1949 <R> occupation <T> mathematician <R> educated_at <T> tata institute of fundamental research	ড. ঊষাদেৱী নৱেন্দ্ৰ ভোছলে হৈছে এগৰাকী ভাৰতীয় গণিতজ্ঞ, শিক্ষাবিদ আৰু গৱেষক । ঊষাদেৱী নৱেন্দ্ৰ ভোছলেৰ জন্ম হৈছিল ১৯৪৯ চনৰ ৩০ মাৰ্চত ; ভাৰতৰ ঢোলেত । ১৯৭১ চনৰ পৰা টাটা মৌলিক গৱেষণা প্ৰতিষ্ঠানত তেখেতে স্নাতকোত্তৰ শিক্ষা আৰম্ভ কৰিছিল । ১৯৭১ চনৰ পৰা ১৯৭৪ চনলৈকে তেখেতে টাটা মৌলিক গৱেষণা প্ৰতিষ্ঠানত গৱেষণা সহযোগী হিচাপে কাৰ্যনিৰ্বাহ কৰিছিল ।	ঊষাদেৱী ভোঁচলে (জন্ম ৩০ মাৰ্চ ১৯৪৯) এগৰাকী ভাৰতীয় গণিতজ্ঞ। তেওঁ টাটা মৌলিক গৱেষণা প্ৰতিষ্ঠানৰ পৰা স্নাতক ডিগ্ৰী লাভ কৰিছিল ।
bn	generate bengali high : <H> ken eastwood <R> country_of_citizenship <T> australia <R> date_of_birth <T> 23 november 1935 <R> member_of_sports_team <T> australia national cricket team <R> occupation <T> cricketer <R> member_of_sports_team <T> australia national cricket team <QR> start_time <QT> 1971 <QR> end_time <QT> 1971 <R> occupation <T> cricketer <R> country_of_citizenship <T> australia <R> member_of_sports_team <T> australia national cricket team <R> occupation <T> cricketer	কেনেথ হামফ্রে ইস্টউড (; জন্মঃ ২৩ নভেম্বর , ১৯৩৫) নিউ সাউথ ওয়েলসের চ্যাটসউড এলাকায় জন্মগ্রহণকারী সাবেক অস্ট্রেলীয় আন্তর্জাতিক ক্রিকেটার । অস্ট্রেলিয়া ক্রিকেট দলের অন্যতম সদস্য ছিলেন তিনি । ১৯৭১ সালে সংক্ষিপ্ত সময়ের জন্যে অস্ট্রেলিয়ার পক্ষে আন্তর্জাতিক ক্রিকেটে অংশগ্রহণ করেছেন ।	কেনেথ ইস্টউড (; জন্মঃ ২৩ নভেম্বর , ১৯৩৫) ভিক্টোরিয়ার মেলবোর্নে জন্মগ্রহণকারী সাবেক অস্ট্রেলীয় আন্তর্জাতিক ক্রিকেটার । ১৯৭১ সালে সংক্ষিপ্ত সময়ের জন্যে অস্ট্রেলিয়ার পক্ষে আন্তর্জাতিক ক্রিকেটে অংশগ্রহণ করেছেন । অস্ট্রেলিয়া ক্রিকেট দলের অন্যতম সদস্য ছিলেন তিনি ।

Table 6.5: Some examples of generation using the best performing model in English, Hindi, Assamese and Bengali

Lang.	Source	Reference	Generated
gu	generate gujarati high : <H> shaktisinh gohil <R> country_of_citizenship <T> india <R> occupation <T> politician <R> date_of_birth <T> 04 april 1960 <R> place_of_birth <T> limbda <R> educated_at <T> bhavnagar university<R> member_of_political_party <T> indian national congress <R> occupation <T> politician	શક્તિસિંહ હરીશચંદ્રસિંહજી ગોહિલ એક ભારતીય રાજકારણી છે જેઓ હાલમાં ભારતની રાજ્યસભાના સંસદ સભ્ય અને ભારતીય રાષ્ટ્રીય કોંગ્રેસના નેતા છે . તેઓ બિહારની રાજ્ય બાબતોના પક્ષ પ્રભારી અને દિલ્હીના વચગાળાના પ્રભારી અને અખિલ ભારતીય કોંગ્રેસ સમિતિના રાષ્ટ્રીય પ્રવક્તા છે . શક્તિસિંહનો જન્મ ૪ એપ્રિલ ૧૯૬૦ના રોજ ભાવનગર જિલ્લાના લીમડા ખાતે , તત્કાલીન બૃહદ મુંબઈ રાજ્યમાં થયો હતો . શક્તિસિંહે ભાવનગર યુનિવર્સિટીમાંથી રસાયણશાસ્ત્ર વિશેષતા સાથે વિજ્ઞાનમાં સ્નાતકની પદવી અને સૌરાષ્ટ્ર યુનિવર્સિટીમાંથી કાયદામાં સ્નાતકોત્તર મેળવ્યો છે .	શકિતીસિંહ ગોહિલ ભારતના ગુજરાત રાજ્યના એક રાજકારણી છે .શકીપસંદ ગોહિલનો જન્મ ૪ એપ્રિલ ૧૯૬૦ના રોજ લીમડા ખાતે થયો હતો .તેમણે ભાવનગર યુનિવર્સિટીમાંથી એમ . ની પદવી મેળવી હતી .તેઓ ભારતીય રાષ્ટ્રીય કોંગ્રેસ પક્ષના સભ્ય
kn	generate kannada high : <H> sooraj pancholi <R> country_of_citizenship <T> india <R> date_of_birth <T> 1990 <R> occupation <T> actor <R> father <T> aditya pancholi <R> mother <T> zarina hoogb<R> cast_member <T> hero <R> occupation <T> actor <R> work_period_start <T> 2015<R> cast_member <T> hero <R> occupation <T> actor <R> work_period_start <T> 2015<R> cast_member <T> time to dance	ಸೂರಜ್ ಪಾಂಚೋಲಿ (ಜನನ ೯ ನವೆಂಬರ್ ೧೯೯೦) ಹಿಂದಿ ಚಿತ್ರಗಳಲ್ಲಿ ಕಾಣಿಸಿಕೊಳ್ಳುವ ಭಾರತೀಯ ಚಲನಚಿತ್ರ ನಟ . ಇವರು ರೋಮ್ಯಾಂಟಿಕ್ ಆಕ್ಷನ್ ಚಿತ್ರ ' ಹೀರೋ ' (೨೦೧೫) ರಲ್ಲಿ ಪಾದಾರ್ಪಣೆ ಮಾಡಿದರು . ಪಾಂಚೋಲಿ ೧೯೯೦ ರ ನವೆಂಬರ್ ೯ ರಂದು ಮುಂಬೈನ ಆದಿತ್ಯ ಪಾಂಚೋಲಿ ಮತ್ತು ಜರೀನಾ ವಹಾಬ್ ದಂಪತಿಗೆ ಜನಿಸಿದರು . ೨೦೧೫ ರಲ್ಲಿ ಅವರು ರೋಮ್ಯಾಂಟಿಕ್ ಆಕ್ಷನ್ ಚಿತ್ರ ' ಹೀರೋ ' ಚಿತ್ರದಲ್ಲಿ ತಮ್ಮ ನಟನೆಗೆ ಪಾದಾರ್ಪಣೆ ಮಾಡಿದರು . ಮುಂಬರುವ ಭಾರತೀಯ ನೃತ್ಯದಲ್ಲೂ ಅವರು ನಟಿಸಿದ್ದಾರೆ ಟೈಮ್ ಟು ಡ್ಯಾನ್ಸ್ ಚಿತ್ರ .	ಸೂರಜ್ ಪಂಚೋಲಿ (ಜನನ ೧೯೯೦) ಒಬ್ಬ ಭಾರತೀಯ ಚಲನಚಿತ್ರ ನಟ .ತಂದೆ ಆದಿತ್ಯ ಪಂಚೋಲಿ ಮತ್ತು ತಾಯಿ ಜಾರಿನಾ ಹೂಬ್ . ಸೂರಜ್ ಪಂಚೋಲಿ ೨೦೧೫ ರಲ್ಲಿ , ಚಲನಚಿತ್ರ ಹೀರೋ ಮೂಲಕ ಅಭಿನಯಕ್ಕೆ ಪಾದಾರ್ಪಣೆ ಮಾಡಿದರು . ಟೈಮ್ ಟು ಡ್ಯಾನ್ಸ್ ' ಚಿತ್ರದಲ್ಲಿ ಸುರಾಜ್ ಪಾಂಚುಲಿ ನಟಿಸಿದ್ದಾರೆ .
ml	generate malayalam high : <H> michael de sanctis <R> canonization_status <T> saint <R> date_of_birth <T> 29 september 1591 <R> date_of_death <T> 10 april 1625 <R> religion <T> catholic church <R> country_of_citizenship <T> spain <R> date_of_birth <T> 29 september 1591<R> canonization_status <T> saint <R> feast_day <T> april 10<R> occupation <T> priest<R> date_of_death <T> 10 april 1625	റോമൻ കത്തോലിക്കാ സഭയിലെ ഒരു വിശുദ്ധനാണ് മൈക്കൽ ദ സാങ്റ്റിസ് (സെപ്റ്റംബർ 29,1591 - ഏപ്രിൽ 10,1625). ഏപ്രിൽ 10 - നാണ് വിശുദ്ധന്റെ തിരുനാൾ ആഘോഷിക്കുന്നത് . 1591 സെപ്റ്റംബർ 29 - ന് സ്പെയ്നിലെ കാറ്റലോനിയായിലാണ് മൈക്കൽ ദ സാങ്റ്റിസിന്റെ ജനനം . പിന്നിട് വൈദികപഠനത്തിന് ശേഷം പൌരോഹിത്യം സ്വീകരിച്ചു . 1625 ഏപ്രിൽ 10 - ന് മുപ്പത്തിയഞ്ചാം വയസ്സിൽ മൈക്കിൾ അന്തരിച്ചു . 1862 - ൽ പിയൂസ് ഒമ്പതാമൻ മാർപ്പാപ്പ മൈക്കിളിനെ വിശുദ്ധനായി നാമകരണം ചെയ്തു .	മൻ കത്തോലിക്കാ സഭയിലെ ഒരു വിശുദ്ധനാണ് മൈക്കൾ ഡി സെയിൻസ് (29 സെപ്റ്റംബർ 1591 - 10 ഏപ്രിൽ 1625) . 1591 സെപ്റ്റംബർ 29 - ന് സ്പെയിനിൽ ജനിച്ചു . ഏപ്രിൽ 10 - ന് വേദപാരംഗതൻ എന്ന പദവി സ്വീകരിച്ചു .തുടർന്ന് അദ്ദേഹം പൌരോഹിത്യകാരനായി സേവനം അനുഷ്ഠിച്ചു . 1625 ഏപ്രിൽ 10 - ന് അദ്ദേഹം അന്തരിച്ചു .
mr	generate marathi high : <H> vandana shiva <R> country_of_citizenship <T> india <R> date_of_birth <T> 05 november 1952 <R> occupation <T> environmentalist <R> occupation <T> philosopher <R> place_of_birth <T> dehradun <R> date_of_birth <T> 05 november 1952 <R> place_of_birth <T> dehradun<R> educated_at <T> panjab university<R> award_received <T> right livelihood award <QR> point_in_time <QT> 1993 <R> award_received <T> sydney peace prize <QR> point_in_time <QT> 2010 <R> winner <T> sydney peace prize <QR> point_in_time <QT> 2010	वंदना शिवा (जन्मः ५ नोव्हेंबर १९५२ ; डेहरादून , उत्तराखंड , भारत) या भारतीय तत्त्वज्ञ , पर्यावरणवादी कार्यकर्त्या व लेखिका आहेत . इ . स . १९९३ मध्ये त्यांना राईट लाईव्हलीहूड पुरस्काराने आणि २०१० मध्ये त्यांना सिडनी शांतता पुरस्काराने सन्मानित करण्यात आले . वंदना शिवा यांचा जन्म ५ नोव्हेंबर १९५२ रोजी डेहरादून (उत्तराखंड) येथे झाला . त्यानंतर त्या चंदीगडच्या पंजाब विद्यापीठात उच्च शिक्षणासाठी गेल्या .	वणदा शिवा (जन्मः ५ नोव्हेंबर १९५२ , डेहरादून , उत्तराखंड , भारत) या एक भारतीय पर्यावरणवादी आणि सामाजिक कार्यकर्त्या आहेत . यांचा जन्म ५ नोव्हेंबर १९५२ रोजी डेहरादून , उत्तराखंड येथे झाला . ते पंजाब विद्यापीठातून मध्ये पदवी संपादन केली . १९९३ मध्ये त्यांना राईट लाइव्हलीहुड पुरस्कार व २०१० मध्ये सिडनी शांतता पुरस्काराने गौरविण्यात आले .

Table 6.6: Some examples of generation using the best performing model in Gujarati, Kannada, Malayalam and Marathi

Lang.	Source	Reference	Generated
or	generate odia high : <H> braja kishore mohanty <R> date_of_birth <T> 15 april 1931 <R> date_of_death <T> 15 june 2018 <R> occupation <T> politician <R> member_of_political_party <T> janata party <R> occupation <T> politician<R> member_of_political_party <T> janata party <R> occupation <T> politician<R> position_held <T> member of the seventh odisha legislative assembly <QR> elected_in <QT> 1977 odisha legislative assembly election <QR> electoral_district <QT> rourkela vidhan sabha constituency <QR> start_time <QT> 26 june 1977<R> date_of_death <T> 15 june 2018	ବ୍ରଜିତ ଭିତ୍ତିଆ (ଜନ୍ମଃ ୧୬ ମଇ ୧୯୫୬) ଜଣେ ଓଡ଼ିଆ ରାଜନୀତିଜ୍ଞ । ସେ ଓଡ଼ିଶା ରାଜନୀତିରେ ଜନତା ଦଳର ଜର୍ମକର୍ତା ଭାବରେ କାର୍ଯ୍ୟ କରୁଥିଲେ । ୧୯୯୦ ମସିହାରେ ହୋଇଥିବା ଓଡ଼ିଶା ବିଧାନ ସଭା ନିର୍ବାଚନରେ ସେ ଜଳସରା ବିଧାନ ସଭା ନିର୍ବାଚନ ମଣ୍ଡଳୀରୁ ୧୦ମ ଓଡ଼ିଶା ବିଧାନ ସଭାକୁ ନିର୍ବାଚିତ ହୋଇଥିଲେ । ବ୍ରଜିତ ଓଡ଼ିଶା ରାଜନୀତିରେ ଜନତା ଦଳର ଜର୍ମକର୍ତା ଭାବରେ କାର୍ଯ୍ୟ କରୁଥିଲେ ।	ବ୍ରଜ କିଶୋର ମହାନ୍ତି (୧୫ ଅପ୍ରେଲ ୧୯୩୧ - ୧୫ ଜୁନ ୨୦୧୮) ଜଣେ ଓଡ଼ିଆ ରାଜନୀତିଜ୍ଞ ଥିଲେ ।ବ୍ରଜ କିଶୋର ମହାନ୍ତି ଓଡ଼ିଶା ରାଜନୀତିରେ ଜନତା ଦଳର ଜର୍ମକର୍ତା ଭାବରେ କାର୍ଯ୍ୟ କରୁଥିଲେ ।ବ୍ରଜ କିଶୋର ମହାନ୍ତି ଓଡ଼ିଶା ରାଜନୀତିରେ ଜନତା ଦଳର ଜର୍ମକର୍ତା ଭାବରେ କାର୍ଯ୍ୟ କରୁଥିଲେ ।୧୯୭୭ ମସିହାରେ ହୋଇଥିବା ଓଡ଼ିଶା ବିଧାନ ସଭା ନିର୍ବାଚନରେ ସେ ରୋଉରକେଲା ବିଧାନ ସଭା ନିର୍ବାଚନ ମଣ୍ଡଳୀରୁ ୭ମ ଓଡ଼ିଶା ବିଧାନ ସଭାକୁ ନିର୍ବାଚିତ ହୋଇଥିଲେ ।୨୦୧୮ ମସିହା ଜୁନ ୧୫ତାରିଖରେ ତାଙ୍କର ପରଲୋକ ଘଟିଥିଲା ।
pa	generate punjabi high : <H> jimmy shergill <R> date_of_birth <T> 03 december 1970 <R> languages_spoken,_written_or_signed <T> hindi <R> name_in_native_language <T> jimmy shergill <R> occupation <T> film actor <R> name_in_native_language <T> jimmy shergill <R> place_of_birth <T> gorakhpur<R> educated_at <T> punjabi university<R> educated_at <T> punjabi university	ਜਿੰਮੀ ਸ਼ੇਰਗਿੱਲ (ਜਨਮ 3 ਦਸੰਬਰ 1970) , ਜਨਮ ਵੇਲੇ ਨਾਮ ਜਸਜੀਤ ਸਿੰਘ ਗਿੱਲ , ਇੱਕ ਭਾਰਤੀ ਅਦਾਕਾਰ ਅਤੇ ਫ਼ਿਲਮ ਨਿਰਮਾਤਾ ਹੈ ਜੋ ਹਿੰਦੀ ਅਤੇ ਪੰਜਾਬੀ ਫ਼ਿਲਮਾਂ ਵਿੱਚ ਕੰਮ ਕਰਦਾ ਹੈ । ਜਿੰਮੀ ਸ਼ੇਰਗਿੱਲ ਦਾ ਜਨਮ ਇੱਕ ਪੰਜਾਬੀ ਸਿੱਖ ਪਰਿਵਾਰ ਵਿੱਚ ਜ਼ਿਲ੍ਹਾ ਗੋਰਖਪੁਰ , ਦੇ ਦਿਓਕੇਹੀਆ ਪਿੰਡ ਵਿੱਚ ਉੱਤਰ ਪ੍ਰਦੇਸ਼ ਵਿਖੇ ਹੋਇਆ । ਇਸ ਨੇ ਪੰਜਾਬ ਪਬਲਿਕ ਸਕੂਲ , ਨਾਭਾ ਅਤੇ ਬਿਕਰਮ ਕਾਲਜ , ਪੰਜਾਬੀ ਯੂਨੀਵਰਸਿਟੀ , ਪਟਿਆਲਾ ਵਿਖੇ ਆਪਣੀ ਬਾਕੀ ਦੀ ਪੜ੍ਹਾਈ ਕੀਤੀ । ਇਸ ਨੇ ਆਪਣੀ ਗ੍ਰੈਜੂ . ਸ਼ਨ ਦੀ ਡਿਗਰੀ ਪੋਸਟ ਗ੍ਰੈਜੂ . ਟ ਸਰਕਾਰੀ ਕਾਲਜ - 11 , ਚੰਡੀਗੜ੍ਹ , ਪੰਜਾਬ ਯੂਨੀਵਰਸਿਟੀ ਤੋਂ ਪੂਰੀ ਕੀਤੀ ।	ਜਿੰਮੀ ਸ਼ੇਰਗਿੱਲ (ਜਨਮ 3 ਦਸੰਬਰ 1970) ਇੱਕ ਭਾਰਤੀ ਅਦਾਕਾਰ ਹੈ ਜੋ ਹਿੰਦੀ ਫਿਲਮਾਂ ਵਿੱਚ ਕੰਮ ਕਰਦਾ ਹੈ ।ਜਿੰਮੀ ਸ਼ੇਰਗਿੱਲ ਦਾ ਜਨਮ ਪੰਜਾਬ ਦੇ ਗੋਰਖਪੁਰ ਵਿੱਚ ਹੋਇਆ ।ਉਸਨੇ ਪੰਜਾਬੀ ਯੂਨੀਵਰਸਿਟੀ ਤੋਂ ਗ੍ਰੈਜੂ . ਸ਼ਨ ਕੀਤੀ ।ਉਸਨੇ ਪੰਜਾਬੀ ਯੂਨੀਵਰਸਿਟੀ ਤੋਂ ਗ੍ਰੈਜੂ .
ta	generate tamil high : <H> vishnu manchu <R> date_of_birth <T> 23 november 1981 <R> languages_spoken,_written_or_signed <T> telugu <R> occupation <T> actor <R> date_of_birth <T> 23 november 1981 <R> place_of_birth <T> chennai<R> occupation <T> actor<R> occupation <T> actor <R> sibling <T> manoj manchu <R> sibling <T> lakshmi manchu<R> occupation <T> actor	மஞ்சு விஷ்ணு ( ) (பிறப்புஃ 23 நவம்பர் 1981) ஒரு தெலுங்கு மொழி திரைப்பட நடிகர் , தயாரிப்பாளர் மற்றும் இயக்குநர் ஆவார் . இவர் பிரபல்யமான நடிகர் மோகன் பாபுவின் மகன் ஆவார் . இவரது பெயரில் விஷ்ணு என்ற திரைப்படத்தில் நடித்து , சிறந்த நடிகருக்கான பிலிம்பேர் விருது பெற்றவர் . இவர் 23 நவம்பர் 1981ஆம் ஆண்டு சென்னை , தமிழ்நாட்டில் பிறந்தார் . இவர் பிரபல்யமான நடிகர் மோகன் பாபுவின் மகன் ஆவார் . இவருக்கு மஞ்சு மனோஜ் என்ற ஒரு இளையசகோதரரும் மற்றும் மஞ்சு லட்சுமி என்ற ஒரு மூத்த சகோதரியும் உண்டு இவர்களும் நடிகர்கள் ஆவார் .	விஷ்ணு மஞ்சு (பிறப்புஃ 23 நவம்பர் 1981) ஒரு தெலுங்கு திரைப்பட நடிகர் ஆவார் .விஷ்ணு மஞ்சு 1981 நவம்பர் 23 அன்று சென்னையில் பிறந்தார் .விஷ்ணு மஞ்சு தென்னிந்தியத் திரைப்பட நடிகர் ஆவார் .இவர் நடிகர் மனோஜ் மஞ்சுவின் சகோதரர் ஆவார் .விஷ்ணு மஞ்சு தென்னிந்தியத் திரைப்பட நடிகர் ஆவார் .
te	generate telugu high : <H> mai masri <R> date_of_birth <T> 02 april 1959 <R> occupation <T> film director <R> date_of_birth <T> 02 april 1959<R> father <T> munib al-masri<R> educated_at <T> san francisco state university	మై మస్రీ (జననం ఏప్రిల్ 2,1959) యునైటెడ్ స్టేట్స్ లో చదువుకున్న పాలస్తీనా చిత్ర దర్శకుడాలు . మై మస్రీ 1959 , ఏప్రిల్ 2న జోర్డాన్ లో జన్మించింది . ఈవిడ తండ్రి మునిబ్ మస్రీ నబ్లూస్ కు చెందినవాడు , తల్లి టెక్సాస్ కి చెందిన అమెరికన్ . మై మస్రీ 1981లో శాన్ ఫ్రాన్సిస్కో స్టేట్ యూనివర్సిటీ నుండి పట్టభద్రురాలయింది .	మై మసరీ (జననంఃి 1959 ఏప్రిల్ 2) ఒక అమెరికన్ చలనచిత్ర దర్శకుడు , రచయిత .మై మసరీ 1959 , ఏప్రిల్ 2న జన్మించాడు .మామ్ మస్రీ తండ్రి మునిబ్ అల్ - మస్రీ .మై మస్రీ సాన్ ఫ్రాన్సిస్కో స్టేట్ యూనివర్సిటీలో గ్రాడ్యుయేషన్ పూర్తి చేసింది .

Table 6.7: Some examples of generation using the best performing model in Oriya, Punjabi, Tamil and Telugu

6.5 Conclusions

In this work we explored the XFLT problem for generation of multi-sentence paragraphs. We created a novel dataset, XLAlign, using the existing XAlignV2 dataset, with a high quality test partition. We explore different methods such as explicit clustering of facts, coverage prompting, grounded decoding and reinforcement learning each of which improve the quality of generation and address the problem of hallucination. These approaches can be used to directly generate Wikipedia like long text from structured data. We also define XPARENT score for evaluation of cross-lingual data-to-text problem which is of particular relevance for partially aligned ground truth text.

Chapter 7

Conclusion and Future work

In this study we propose methods for the purpose of enriching structured and unstructured content over the Wikipedia ecosystem. We aim to automate the pipeline of article generation over Wikipedia for low resource Indian languages using structure knowledge graphs from Wikidata as input. We also contribute methods of consolidating the factual information present in the Wikipedia articles from these low resource Indian languages and using that to enrich Wikidata.

We begin by drawing attention to the scarcity of available resources in low-resource languages and put forth the idea of employing cross-lingual methodologies for text generation and fact extraction in order to tackle this scarcity. For this purpose we introduced the problem of cross lingual fact to text alignment, cross lingual fact extraction and cross lingual fact to text generation. The XF2T problem further advances into the task of generating cross lingual long text which aims to generate the complete article from all facts about an entity.

Chapter 1 covered an exploratory analysis of the availability of resources across the languages explored in this thesis and provided a brief motivation for the work done. It also introduced the various sub tasks tackled in the thesis and a brief descriptions of the major challenges involved which make these problems worth exploring.

Chapter 2 covered the previous work done in related problems and the identify the gaps in literature which are filled by the current work. In this chapter we observed that while data-to-text generation and fact extraction are standard and well explored problems, there are not been enough work in the space of cross lingual fact to text generation and fact extraction. We also look at the widely popular metircs used to evaluate text generation tasks and provide a brief overview of their functioning and possible limitations.

The thesis also contributed by proposing parallel dataset for the tasks of cross lingual fact to text generation and extraction. **Chapter 3** describes in detail the process of creating the XAlign dataset which is used in subsequent chapters. The dataset contained aligned (sentence,facts) pairs for 12 languages constituting English and 11 other Indian languages. The sentences were obtained from the native language Wikipedia whereas the English facts came

from the Wikidata triples. We proposed a two stage pipeline for aligning the sentence to English facts, by first passing them through a maximum recall candidate generation stage followed by a final candidate selection phase. We also highlight the data cleaning, preprocessing and the setup for the manual annotations. We infer from the works of this chapter that cross lingual fact to text alignment is not a trivial task, however approaches utilising multilingual pretrained transformers in combination with transfer learning and distant supervision strategies can perform well.

Following the construction of the dataset, in **Chapter 4** we introduce the task of cross lingual fact extraction and propose strong baselines for the same. We observe a single end-to-end generative approach towards extraction performs better than a two-phase pipeline.

Chapter 5 provides a detailed account of multiple experiments for the task of cross lingual fact to text generation. We construct strong baselines by modifying the existing data-to-text systems for our task. We evaluate the performances of different components of the generative system like the choice of pretrained transformer model, the training setup or possible continued pretraining strategies. We observe that using multilingual pretrained transformers provide significant gains over vanilla transformers, further using a multilingual pretraining by translating existing related, but noisy datasets into our desired target languages. We also propose fact-aware embeddings which outperform the explored baselines.

Finally, we address the limitations of single sentence generation and the major problem of unfaithful generation, and explore the task of fact to long text generation with specific focus on reducing hallucination in the generated content in **Chapter 6** of this thesis. We efficiently modify the XAlign dataset in order to construct an appropriate dataset for the new task with high quality test partition. We observe that special techniques for organising the input facts are needed in order to incorporate a greater number of input facts and the existing systems trained to only generate a single sentence perform poorly when given the task of generating longer pieces of text. Thus we propose a fact organiser and utilise methods like coverage prompts and reinforcement learning in order to generate more faithful and improved content. Furthermore, we also address the lack of a reliable metric which can handle the diverging references and thus devise a source-dependent cross lingual metric for the XF2T task.

Overall, this thesis presented architecture and system

7.1 Future work

1. The XAlign dataset focused on the Persons' domain and contains encyclopedic style text. Since the entire alignment process is general and does not take domain specific measures, one could possibly extent the entire study to multiple other domains or languages and analyse the performance and scalability. Similarly, we can also exploit possible domain

specific features to improve the performance for the respective tasks in the current domain as well.

2. For the task of cross lingual fact to long text generation, we focus on reducing hallucination and grouping the facts in a logical order. However not enough has been explored regarding evaluating and improving the coherence of the generated sentences during the generation step. This involves ensuring proper use of pronouns across sentences and handling repetition of information. Currently these directions are bottlenecked by the availability of co-reference resolution and entity linking systems for the low resource Indian languages but can be explored in future.

3. The task of cross lingual fact extraction proposed here currently focuses on extracting facts centric to an entity, this can be expanded further by developing a more general CLFE system. This would also require creating new datasets for the same since currently none exist.

4. The task of cross lingual fact to text generation or cross lingual information extraction can be expanded into multiple modalities beyond text or facts. An example of this could be a text generation system which also incorporates images, graphs and tables etc alongside the facts. This could definitely take the process of automating the article generation one step further,

5. While the current models explained in this thesis can generate fluent text output, there is still some scope in improving the quality of generations. One of the common problems is the model getting stuck in a loop or repeating information. Such problems can be tackled to further improve the generations.

6. Regarding the experimental framework, our approaches primarily revolve around end-to-end neural network-based methods. However, it is worth noting that there are alternative problem settings that incorporate a combination of neural and rule-based approaches, which offer avenues for further exploration. Considering the challenges posed by neural models in ensuring factual accuracy, the aforementioned concept presents an intriguing pathway to explore. In this context, leveraging rule-based methods could prove advantageous, particularly in the data selection stages, as they tend to offer greater interpretability. One example could be exploring the problem of domain specific template generation where instead of automating the articles, domain specific templates are generated which can then be filled by rule based systems.

Overall, this thesis has made significant contributions to enrich the encyclopedic content via cross lingual approaches. However, there is still some scope to modify or extend the components described here to further enhance the quality and quantity of the generated content.

Appendix A

Effectiveness of Pretrained Transformer Architectures

In the appendix, we explore multiple problems in the domain of natural language processing which extensively make use of pretrained transformers. These tasks were done during the course of this thesis as a part of various shared tasks. The tasks differ greatly in terms of the domain, data used and the problems dealt.

We will be discussing our works in brief on four different problems:

1. Multilingual Tweet intimacy analysis
2. Identifying Human Values behind Arguments
3. Analysing disagreements between annotators
4. Citation Context Classification for scientific documents

A prevalent theme among the four problems addressed in this chapter is the utilization of pretrained transformer architectures. Throughout this chapter, we showcase the effectiveness of employing these architectures to achieve favorable outcomes in tackling various problems. These findings have guided us in making specific design choices for the architectures discussed in the preceding sections of this thesis.

A.0.1 Multilingual Tweet intimacy analysis

Intimacy in language refers to the degree of emotional closeness or familiarity between individuals, which is often reflected in the choice of words, tone, and context of communication. The problem statement involved scoring the amount of intimacy in tweets from 10 different languages in the form of a number between 1 to 5. Out of these 10 languages, minimal training data was provided for 6 languages, whereas zero-shot performance was evaluated for the other 4. We utilise domain-specific features and domain-adapted pre-trained models in order to improve the understanding of intimacy in tweets. We further utilise a translation-based data augmentation pipeline which proves effective in significantly improving the scores for unseen

languages. We also explore multiple ablations applied to our pipeline in order to better understand the contribution of the various components used. Our system achieved third rank for unseen languages with a Pearsons r score of 0.485 and tenth rank overall with a Pearsons r score of 0.592.

Figure A.1 shows the pipeline of the proposed architecture. As it can be seen, we augment the data using translations and pass it through a preprocessing stage. Here we extract the emojis and get their textual descriptions and vector embeddings. We use a pretrained transformer to get teh embeddings from the text. Finally, the embeddings of text and emojis are concatenated and passed through a neural network with a regression layer.

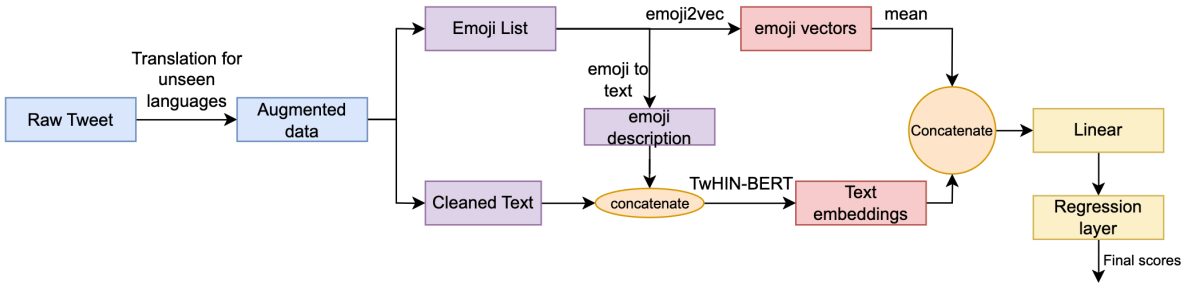

Figure A.1: The pipeline for the proposed architecture

Ablation		pretrained model				filtering	emojis		translation		others
	submitted system	distill-bert	mbert	TwHIN-bert	xlm-T	no cleaning	no emoji	emoji-2-text	no trans	trans test	no trans no emoji
English	0.706	0.602	0.636	0.688	0.706	0.704	**0.723**	0.706	0.704	0.702	0.715
Spanish	0.725	0.604	0.622	**0.727**	0.711	0.720	0.705	0.709	0.694	0.680	0.678
Portuguese	0.648	0.514	0.545	0.606	**0.676**	0.671	0.674	0.668	0.645	0.652	0.645
Italian	**0.727**	0.590	0.558	0.710	0.698	0.694	0.690	0.692	0.694	0.709	0.695
French	0.628	0.559	0.580	0.631	0.675	0.681	0.674	0.674	0.681	0.680	**0.692**
Chinese	0.698	0.666	0.664	0.721	0.714	0.717	0.720	**0.729**	0.720	0.677	0.708
Hindi	0.203	0.176	0.174	0.189	**0.217**	0.160	0.184	0.184	0.200	0.235	0.206
Dutch	0.591	0.488	0.487	0.567	**0.630**	0.608	**0.611**	0.603	0.602	0.604	0.592
Korean	0.307	0.277	0.269	**0.404**	0.358	0.306	0.372	0.359	0.322	0.319	0.374
Arabic	**0.644**	0.395	0.365	0.637	0.605	0.572	0.647	0.623	0.653	0.604	0.628
Seen Lang	0.684	0.591	0.612	0.687	0.704	**0.707**	0.699	0.702	0.694	0.684	0.693
Unseen Lang	0.485	0.410	0.384	**0.516**	0.477	0.471	0.484	0.477	0.434	0.367	0.420
Overall	0.592	0.512	0.510	**0.605**	0.602	0.601	0.601	0.600	0.573	0.535	0.570

Table A.1: The table shows the results for all the experiments and the ablation studies. The first column highlights our submitted system. All the other columns highlight different ablation experiments where one of the components of our pipeline is modified or removed

This work shows how carefully designed data augmentation techniques can help in better cross lingual transfer of learning and improved scores over unseen languages. The work also

highlights the importance of using efficient encoding strategies to include domain specific features like emojis for an improved understanding of the text. The results also show that, though efficient, the transformer based deep learning models are prone to variance, and just utilising the right set of hyperparameters can result in significant gains. Finally, we also see that domain-adapted pre-trained transformers can capture nuances of in-domain text, and when used with simple deep learning and machine learning models, could give competitive results.

A.0.2 Identifying Human Values behind Arguments

The task aimed at identifying the human values involved in a given premise, stance and conclusion triple. Humans often come to different conclusions given the same premise. This variation can be attributed to their values. Identifying the values behind the arguments is helpful in understanding the argument itself. Downstream tasks like supporting or opposing argument generation can benefit from value identification. In this task, we aim to identify 20 value categories in a given premise, stance and conclusion pair. We use DeBERTa, a pre-trained language model, that has shown remarkable success in various NLP tasks, including classification. The proposed method tokenizes the premise, stance and conclusion text using the pretrained tokenizer, and then concatenates them and feeds it into the LM, generates a representation of the combined text, and maps it to a set of values using a fully connected Neural Network. The model is trained on a Multi-margin loss function and evaluated on metrics such as accuracy, precision, recall, and F1 score. Figure A.2 depicts the hierarchy of the values. and Figure A.3 depicts the pipeline of the transformer

Test set / Approach	All	Self-direction: thought	Self-direction: action	Stimulation	Hedonism	Achievement	Power: dominance	Power: resources	Face	Security: personal	Security: societal	Tradition	Conformity: rules	Conformity: interpersonal	Humility	Benevolence: caring	Benevolence: dependability	Universalism: concern	Universalism: nature	Universalism: tolerance	Universalism: objectivity
DeBERTa	.26	**.62**	.11	.12	.16	.07	.07	.07	.04	.00	.01	**.72**	**.62**	**.55**	**.46**	.00	.17	.54	.56	**.31**	.04
DeBERTa+Extra Classes	**.43**	.43	**.56**	**.25**	**.39**	**.59**	**.25**	**.37**	**.20**	**.70**	**.60**	.41	.49	.15	.08	**.52**	.21	**.63**	**.70**	.14	**.48**
DeBERTa+All Levels	.24	.23	.41	.00	.00	.47	.16	.13	.00	.57	.41	.17	.39	.00	.00	.50	.25	.53	.00	.21	0.33
Hierarchichal	.25	.31	.41	.00	.00	.44	.19	.20	.06	.54	.49	.18	.36	.00	.05	.40	**.26**	.56	.06	.22	.33

Table A.2: F1 scores for classification across the different classes

We propose a method that uses a pre-trained language model, DeBERTa, to tokenize and concatenate the text before feeding it into a fully connected neural network. We also show that leveraging the hierarchy in values improves the performance by 0.14 F1 score compared to only using level 2 values.

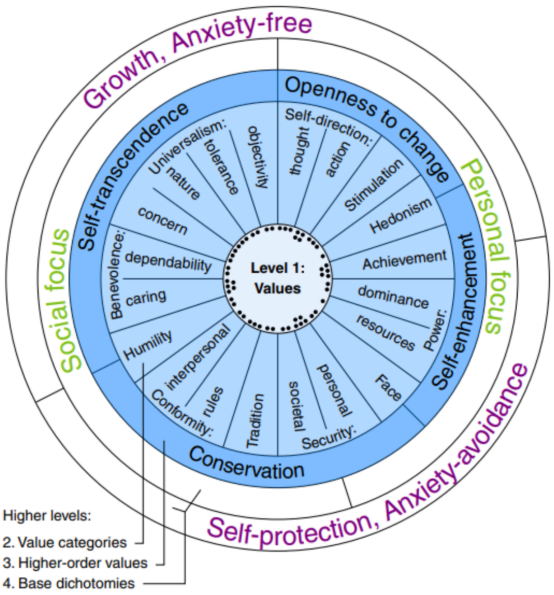

Figure A.2: Values in the data organized higher level to lower level

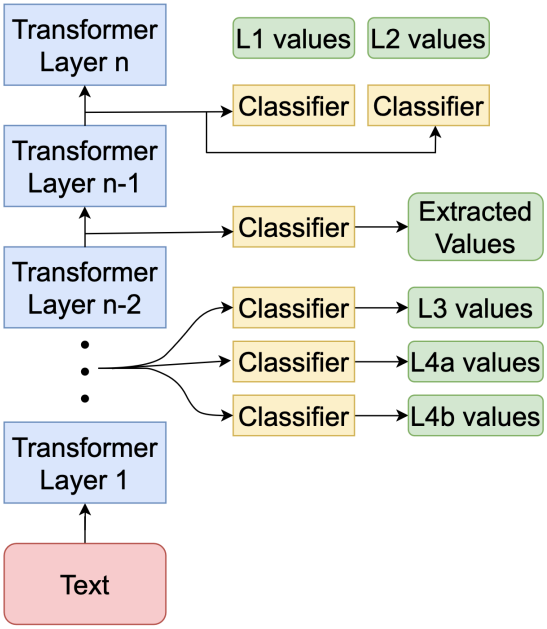

Figure A.3: Using Internal Hidden states to feed classifiers to exploit the Hierarchy in Values

We found that identifying the hierarchy in the values improves performance, and adding 5 high-level values to the existing 20 values significantly improved the models accuracy compared to just using 20. Hierarchical methods did not perform as expected due to missing high level values. The proposed approach has the potential to be an effective NLP model for identifying values in arguments.

A.0.3 Analysing disagreements between annotators

Natural language expressions, such as sentences and phrases, can often have multiple possible interpretations depending on the context in which they are used. This ambiguity arises due to languages inherent complexity and flexibility, which can lead to different interpretations of the same expression by different individuals. Additionally, subjective tasks can lead to disagreements between annotators with different perspectives or interpretations of the same text. The current Learning With Disagreements (Le-Wi-Di) task focuses entirely on such subjective tasks, where training with aggregated labels makes much less sense. In this task, we worked with three (textual) datasets with different characteristics in terms of languages (English and Arabic), tasks (misogyny, hate speech, offensiveness detection) and annotations methodology (experts, specific demographic groups, AMT-crowd).

Testing strategy	HS-Brexit				ArMIS				MD-Agreement			
	val		test		val		test		val		test	
	CE	F1	CE	F1	CE	F1	CE	F1	CE	F1	CE	F1
Majority baseline	2.71	0.89	5.62	0.89	8.23	0.60	8.91	0.57	7.74	0.65	7.38	0.67
Hard loss	0.47	0.86	0.75	0.84	4.55	0.57	**4.01**	0.58	7.50	0.51	9.92	0.42
Soft loss	0.65	0.88	1.07	0.86	3.82	0.58	4.70	0.56	6.42	0.57	8.73	0.50
Better mixers with multi-head attention	0.58	0.88	**0.58**	0.84	-	-	-	-	3.40	0.59	**3.70**	0.58

Table A.3: Results for cross entropy and micro F1 across the three datasets

We leverage this additional information in order to get more accurate estimates of each annotators annotation. All the datasets provide a multiplicity of labels for each instance. The focus is on developing methods able to capture agreements/disagreements rather than focusing on developing the best model. Since a "truth" cannot be assumed, "soft" evaluation is the primary form of evaluating performances, i.e. an evaluation that considers how well the models probabilities reflect the level of agreement among annotators.

Table A.3 summarizes the results from our experiments. Our architectural improvements, which included designing better mixers, gave better cross entropy and micro F1 results for both HS-Brexit and MD-Agreement datasets. Our results highlight the benefits of using soft loss over hard loss for such controversial cases. We also find that using better ways to combine multiple channels of information can lead to the best results by potentially helping us model the annotators and predict their choices. However, the deep learning models of today are primarily encouraged to focus on hard evaluation scores like F1 and disregard the noise in the data, which leads to excellent results in constrained lab environments but fail in real-world scenarios. Finding more ways to incorporate the subjectivity of real-world data and peoples opinions could help make these models more robust and generalizable

A.0.4 Citation Context Classification

The shared task focused on classifying citation context in research publications based on their influence and purpose and contained two different sub tasks. Subtask A aims at identifying the purpose of the citation. Subtask A involves a multiclass classification of citations into one of six classes: Background, Uses, Compare and Contrast, Motivation, Extension, and Future. Subtask B aims at identifying the importance of the citation. It is a binary classification of citations into one of two classes: Incidental and Influential. Our proposed system performed the best on the leaderboard and was awarded the best paper award.

We build a classifier using a transformer model for obtaining the embeddings followed by a classifier head. We experiment with various transformer models and observe that a domain adapted transformer model (SciBERT) performed the best. We also experiment with different classifier heads and observe that a neural classifier performs better than other alternatives.

Subtask A		Subtask B	
Model	**Macro F1**	**Model**	**Macro F1**
BERT-uncased + Linear	**0.4350 ± 0.00439**	BERT-uncased + Linear	0.6611 ± 0.0065
RoBERTa + Linear	0.4258 ± 0.0264	RoBERTa + Linear	0.6636 ± 0.0038
SciBERT-cased + Linear	0.4232 ± 0.0157	SciBERT-cased + Linear	0.6720 ± 0.0041
SciBERT-uncased + Linear	**0.4333 ± 0.0094**	SciBERT-uncased + Linear	**0.6778 ± 0.0098**
SciBERT-uncased + Bi-LSTM	0.4246 ± 0.01267	SciBERT-uncased + Bi-LSTM	0.6741 ± 0.0101
Random Forest	0.2742	Random Forest	0.6559
Title + Citation Context	0.4232 ± 0.01486	Title + Citation Context	**0.6781 ± 0.0077**

Table A.4: Results of subtask A and subtask B

Finally, in order to tackle the extreme class imbalance for the multi class classification task, we use weighted loss functions which result in significantly better performance over the macro-F1 metric. Table A.4 provides the results for the approaches tried. This work shows how domain adapted embeddings can capture nuances of scientific documents, and simple deep learning and machine learning models could give competitive results. Despite a small dataset, good results could be achieved.

A.1 Conclusion

In conclusion, though unrelated, these tasks helped us better understand the intricacies involved with using large pretrained transformer models in various contexts and shaped some of the choices made in our proposed systems across different chapters of this thesis.

Related publications

- Bhavyajeet Singh, Aditya Hari, Rahul Mehta, Manish Gupta, Vasudeva Varma **Cross-lingual Multi-Sentence Fact-to-Text Generation: Generating factually grounded Wikipedia Articles using Wikidata** In Proceedings of The Wiki Workshop 2023

- Bhavyajeet Singh, Pavan Kandru, Anubhav Sharma, Vasudeva Varma. **Massively Multilingual Language Models for Cross Lingual Fact Extraction from Low Resource Indian Languages** In Proceeding of ICON-2022 Main Conference.

- Bhavyajeet Singh, Aditya Hari, Rahul Mehta, Manish Gupta, Vasudeva Varma **XFLT : Exploring Techniques for Generating Cross Lingual Factually Grounded Long Text** Under review at ECAI 2023

- Shivprasad Sagare, Tushar Abhishek, Bhavyajeet Singh, Anubhav Sharma, Manish Gupta, Vasudeva Varma. **XF2T: Cross-lingual Fact-to-Text Generation for Low-Resource Languages** arXiv, Under review at INLG 2023

- Tushar Abhishek, Shivprasad Sagare, Bhavyajeet Singh, Anubhav Sharma, Manish Gupta, Vasudeva Varma. **XAlign: Cross-lingual Fact-to-Text Alignment and Generation for Low-Resource Languages.** In Companion Proceedings of the Web Conference 2022 (WWW 22 Companion).

Other Publications

- Bhavyajeet Singh, Ankita Maity, Siri Venkata Pavan kumar Kandru, Aditya Hari and Vasudeva Varma. **iREL at SemEval-2023 Task 9: Improving Understanding of Multilingual Tweets Using Translation-Based Augmentation and Domain Adapted Pre-Trained Models** in Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023), Association for Computational Linguistics

- Siri Venkata Pavan kumar Kandru, Bhavyajeet Singh, Ankita Maity, Kancharla Aditya Hari and Vasudeva Varma . **Tenzin-Gyatso at SemEval-2023 Task 4: Identifying Human Values behind Arguments Using DeBERTa** in Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023), Association for Computational Linguistics

- Ankita Maity, Siri Venkata Pavan kumar Kandru, Bhavyajeet Singh, Kancharla Aditya Hari and Vasudeva Varma. **IREL at SemEval-2023 Task 11: User Conditioned Modelling for Toxicity Detection in Subjective Tasks** in Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023), Association for Computational Linguistics

- Himanshu Maheshwari, Bhavyajeet Singh and Vasudeva Varma. **SciBERT Sentence Representation for Citation Context Classification** In Proceedings of the Second Workshop on Scholarly Document Processing. Association for Computational Linguistics

Bibliography

[1] T. Abhishek, S. Sagare, B. Singh, A. Sharma, M. Gupta, and V. Varma. Xalign: Cross-lingual fact-to-text alignment and generation for low-resource languages. In *The World Wide Web Conference*, pages 171–175, 2022.

[2] O. Agarwal, H. Ge, S. Shakeri, and R. Al-Rfou. Knowledge graph based synthetic corpus generation for knowledge-enhanced language model pre-training. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3554–3565, 2021.

[3] G. Angeli, M. J. Johnson Premkumar, and C. D. Manning. Leveraging linguistic structure for open domain information extraction. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 344–354, Beijing, China, July 2015. Association for Computational Linguistics.

[4] G. Attardi. Wikiextractor. `https://github.com/attardi/wikiextractor`, 2015.

[5] K. Bali, M. Choudhury, and P. Biswas. Indian language part-of-speech tagset: Bengali. *Linguistic Data Consortium, Philadelphia, LDC2010T16*, 2010.

[6] S. Banerjee and A. Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72, 2005.

[7] K. Bontcheva and Y. Wilks. Automatic report generation from ontologies: the miakt approach. In *International conference on application of natural language to information systems*, pages 324–335. Springer, 2004.

[8] J. A. Botha, Z. Shan, and D. Gillick. Entity linking in 100 languages. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7833–7845, 2020.

[9] S. R. Bowman, G. Angeli, C. Potts, and C. D. Manning. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal, Sept. 2015. Association for Computational Linguistics.

[10] D. L. Chen and R. J. Mooney. Learning to sportscast: a test of grounded language acquisition. In *Proceedings of the 25th international conference on Machine learning*, pages 128–135, 2008.

[11] M. Chen, S. Wiseman, and K. Gimpel. Wikitablet: A large-scale data-to-text dataset for generating wikipedia article sections. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 193–209, 2021.

[12] W. Chen, Y. Su, X. Yan, and W. Y. Wang. Kgpt: Knowledge-grounded pre-training for data-to-text generation. *arXiv preprint arXiv:2010.02307*, 2020.

[13] Z. Chi, L. Dong, S. Ma, S. H. X.-L. Mao, H. Huang, and F. Wei. Mt6: Multilingual pretrained text-to-text transformer with translation pairs, 2021.

[14] Z. Chi, L. Dong, F. Wei, W. Wang, X.-L. Mao, and H. Huang. Cross-lingual natural language generation via pre-training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7570–7577, 2020.

[15] P. Cimiano, J. Lüker, D. Nagel, and C. Unger. Exploiting ontology lexica for generating natural language texts from rdf data. In *Proceedings of the 14th European Workshop on Natural Language Generation*, pages 10–19, 2013.

[16] 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. *arXiv preprint arXiv:1911.02116*, 2019.

[17] A. Conneau, R. Rinott, G. Lample, A. Williams, S. R. Bowman, H. Schwenk, and V. Stoyanov. Xnli: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2018.

[18] 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.

[19] 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.

[20] B. Dhingra, M. Faruqui, A. Parikh, M.-W. Chang, D. Das, and W. Cohen. Handling divergent reference texts when evaluating table-to-text generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4884–4895, 2019.

[21] D. Duma and E. Klein. Generating natural language from linked data: Unsupervised template extraction. In *Proceedings of the 10th International Conference on Computational Semantics (IWCS 2013)–Long Papers*, pages 83–94, 2013.

[22] H. Elsahar, P. Vougiouklis, A. Remaci, C. Gravier, J. Hare, F. Laforest, and E. Simperl. T-REx: A large scale alignment of natural language with knowledge base triples. In *Proceedings of the*

Eleventh International Conference on Language Resources and Evaluation (LREC 2018), Miyazaki, Japan, May 2018. European Language Resources Association (ELRA).

[23] F. Feng, Y. Yang, D. Cer, N. Arivazhagan, and W. Wang. Language-agnostic bert sentence embedding. *arXiv preprint arXiv:2007.01852*, 2020.

[24] T. Ferreira, C. Gardent, N. Ilinykh, C. Van Der Lee, S. Mille, D. Moussallem, and A. Shimorina. The 2020 bilingual, bi-directional webnlg+ shared task overview and evaluation results (webnlg+ 2020). In *Proceedings of the 3rd International Workshop on Natural Language Generation from the Semantic Web (WebNLG+)*, 2020.

[25] T. C. Ferreira, D. Moussallem, E. Krahmer, and S. Wubben. Enriching the webnlg corpus. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 171–176, 2018.

[26] Z. Fu, B. Shi, W. Lam, L. Bing, and Z. Liu. Partially-aligned data-to-text generation with distant supervision. *arXiv preprint arXiv:2010.01268*, 2020.

[27] C. Gardent, A. Shimorina, S. Narayan, and L. Perez-Beltrachini. Creating training corpora for NLG micro-planners. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 179–188, Vancouver, Canada, July 2017. Association for Computational Linguistics.

[28] C. Gardent, A. Shimorina, S. Narayan, and L. Perez-Beltrachini. Creating training corpora for nlg micro-planning. In *55th annual meeting of the Association for Computational Linguistics (ACL)*, 2017.

[29] A. Gupte, S. Sapre, and S. Sonawane. Knowledge graph generation from text using neural machine translation techniques. In *2021 International Conference on Communication information and Computing Technology (ICCICT)*, pages 1–8, 2021.

[30] S. Gururangan, A. Marasović, S. Swayamdipta, K. Lo, I. Beltagy, D. Downey, and N. A. Smith. Dont stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, 2020.

[31] M. Honnibal and I. Montani. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. *To appear*, 2018.

[32] Z. Jin, Q. Guo, X. Qiu, and Z. Zhang. Genwiki: A dataset of 1.3 million content-sharing text and graphs for unsupervised graph-to-text generation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2398–2409, 2020.

[33] J. Kanerva, S. Rönnqvist, R. Kekki, T. Salakoski, and F. Ginter. Template-free data-to-text generation of Finnish sports news. In *Proceedings of the 22nd Nordic Conference on Computational Linguistics*, pages 242–252, Turku, Finland, Sept.–Oct. 2019. Linköping University Electronic Press.

[34] M. G. Kendall. A new measure of rank correlation. *Biometrika*, 30(1/2):81–93, 1938.

[35] 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.

[36] K. Kolluru, M. Rezk, P. Verga, W. W. Cohen, and P. Talukdar. Multilingual fact linking, 2021.

[37] K. Kolluru, M. Rezk, P. Verga, W. W. Cohen, and P. Talukdar. Multilingual fact linking, 2021.

[38] R. Koncel-Kedziorski, D. Bekal, Y. Luan, M. Lapata, and H. Hajishirzi. Text generation from knowledge graphs with graph transformers, 2019.

[39] H. W. Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

[40] A. Kunchukuttan. The IndicNLP Library. `https://github.com/anoopkunchukuttan/indic_nlp_library/blob/master/docs/indicnlp.pdf`, 2020.

[41] H. Lai, A. Toral, and M. Nissim. Thank you bart! rewarding pre-trained models improves formality style transfer. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 484–494, 2021.

[42] R. Lebret, D. Grangier, and M. Auli. Neural text generation from structured data with application to the biography domain. *arXiv preprint arXiv:1603.07771*, 2016.

[43] J. Lehmann, R. Isele, M. Jakob, A. Jentzsch, D. Kontokostas, P. N. Mendes, S. Hellmann, M. Morsey, P. van Kleef, S. Auer, and C. Bizer. Dbpedia - a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, 6(2):167–195, 2015.

[44] 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. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, 2020.

[45] R. Li, D. Li, J. Yang, F. Xiang, H. Ren, S. Jiang, and L. Zhang. Joint extraction of entities and relations via an entity correlated attention neural model. *Information Sciences*, 581:179–193, 2021.

[46] S. Li, K. K. Wong, D. Zhu, and C. C. Fung. Improving question answering over knowledge graphs using graph summarization, 2022.

[47] Y. Liang, N. Duan, Y. Gong, N. Wu, F. Guo, W. Qi, M. Gong, L. Shou, D. Jiang, G. Cao, X. Fan, R. Zhang, R. Agrawal, E. Cui, S. Wei, T. Bharti, Y. Qiao, J.-H. Chen, W. Wu, S. Liu, F. Yang, D. Campos, R. Majumder, and M. Zhou. Xglue: A new benchmark dataset for cross-lingual pretraining, understanding and generation, 2020.

[48] 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.

[49] Y. Lu, H. Lu, G. Fu, and Q. Liu. Kelm: Knowledge enhanced pre-trained language representations with message passing on hierarchical relational graphs, 2022.

[50] Mausam, M. Schmitz, S. Soderland, R. Bart, and O. Etzioni. Open language learning for information extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 523–534, Jeju Island, Korea, July 2012. Association for Computational Linguistics.

[51] H. Mei, T. UChicago, M. Bansal, and M. R. Walter. What to talk about and how? selective generation using lstms with coarse-to-fine alignment. In *Proceedings of NAACL-HLT*, pages 720–730, 2016.

[52] D. Moussallem, D. Gnaneshwar, T. Castro Ferreira, and A.-C. Ngonga Ngomo. Nabu–multilingual graph-based neural rdf verbalizer. In *The Semantic Web–ISWC 2020: 19th International Semantic Web Conference, Athens, Greece, November 2–6, 2020, Proceedings, Part I 19*, pages 420–437. Springer, 2020.

[53] P. Nema, S. Shetty, P. Jain, A. Laha, K. Sankaranarayanan, and M. M. Khapra. Generating descriptions from structured data using a bifocal attention mechanism and gated orthogonalization. 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 1539–1550, 2018.

[54] J. Novikova, O. Dušek, and V. Rieser. The e2e dataset: New challenges for end-to-end generation. *arXiv preprint arXiv:1706.09254*, 2017.

[55] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, 2002.

[56] C. Patel and K. Gali. Part-of-speech tagging for gujarati using conditional random fields. In *Proceedings of the IJCNLP-08 Workshop on NLP for Less Privileged Languages*, 2008.

[57] M. Popović. chrf++: words helping character n-grams. In *Proceedings of the second conference on machine translation*, pages 612–618, 2017.

[58] R. Puduppully, L. Dong, and M. Lapata. Data-to-text generation with entity modeling. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2023–2035, 2019.

[59] R. Qader, K. Jneid, F. Portet, and C. Labbé. Generation of company descriptions using concept-to-text and text-to-text deep models: dataset collection and systems evaluation. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 254–263. Association for Computational Linguistics, 2018.

[60] P. Qi, Y. Zhang, Y. Zhang, J. Bolton, and C. D. Manning. Stanza: A Python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 2020.

[61] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu, et al. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(140):1–67, 2020.

[62] G. Ramesh, S. Doddapaneni, A. Bheemaraj, M. Jobanputra, R. AK, A. Sharma, S. Sahoo, H. Diddee, M. J, D. Kakwani, N. Kumar, A. Pradeep, S. Nagaraj, K. Deepak, V. Raghavan, A. Kunchukuttan, P. Kumar, and M. S. Khapra. Samanantar: The largest publicly available parallel corpora collection for 11 indic languages, 2021.

[63] G. Ramesh, S. Doddapaneni, A. Bheemaraj, M. Jobanputra, R. AK, A. Sharma, S. Sahoo, H. Diddee, D. Kakwani, N. Kumar, et al. Samanantar: The largest publicly available parallel corpora collection for 11 indic languages. *arXiv preprint arXiv:2104.05596*, 2021.

[64] E. Reiter and R. Dale. Building applied natural language generation systems. *Natural Language Engineering*, 3(1):57–87, 1997.

[65] L. F. R. Ribeiro, M. Schmitt, H. Schütze, and I. Gurevych. Investigating pretrained language models for graph-to-text generation, 2021.

[66] S. Sagare, T. Abhishek, B. Singh, A. Sharma, M. Gupta, and V. Varma. Xf2t: Cross-lingual fact-to-text generation for low-resource languages. *arXiv preprint arXiv:2209.11252*, 2022.

[67] A. B. Sai, A. K. Mohankumar, and M. M. Khapra. A survey of evaluation metrics used for nlg systems. *ACM Computing Surveys (CSUR)*, 55(2):1–39, 2022.

[68] V. Sanh, L. Debut, J. Chaumond, and T. Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108, 2019.

[69] T. Sellam, D. Das, and A. Parikh. Bleurt: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, 2020.

[70] H. Shahidi, M. Li, and J. Lin. Two birds, one stone: A simple, unified model for text generation from structured and unstructured data. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3864–3870, 2020.

[71] S. Srivastava, M. Patidar, S. Chowdhury, P. Agarwal, I. Bhattacharya, and G. Shroff. Complex question answering on knowledge graphs using machine translation and multi-task learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3428–3439, Online, Apr. 2021. Association for Computational Linguistics.

[72] D. Sui, Y. Chen, K. Liu, J. Zhao, X. Zeng, and S. Liu. Joint entity and relation extraction with set prediction networks, 2020.

[73] R. Tian, S. Narayan, T. Sellam, and A. P. Parikh. Sticking to the facts: Confident decoding for faithful data-to-text generation. *arXiv preprint arXiv:1910.08684*, 2019.

[74] R. Tian, S. Narayan, T. Sellam, and A. P. Parikh. Sticking to the facts: Confident decoding for faithful data-to-text generation, 2020.

[75] P. Vougiouklis, H. Elsahar, L.-A. Kaffee, C. Gravier, F. Laforest, J. Hare, and E. Simperl. Neural wikipedian: Generating textual summaries from knowledge base triples. *Journal of Web Semantics*, 52:1–15, 2018.

[76] D. Vrandečić and M. Krötzsch. Wikidata: A free collaborative knowledgebase. *Commun. ACM*, 57(10):7885, sep 2014.

[77] Q. Wang, S. Yavuz, X. V. Lin, H. Ji, and N. Rajani. Stage-wise fine-tuning for graph-to-text generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: Student Research Workshop*, pages 16–22, 2021.

[78] T.-H. Wen, M. Gasic, N. Mrksic, L. M. Rojas-Barahona, P.-H. Su, D. Vandyke, and S. Young. Multi-domain neural network language generation for spoken dialogue systems. *arXiv preprint arXiv:1603.01232*, 2016.

[79] R. J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Reinforcement learning*, pages 5–32, 1992.

[80] S. Wiseman, S. M. Shieber, and A. M. Rush. Challenges in data-to-document generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2253–2263, 2017.

[81] 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, 2020.

[82] 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.

[83] X. Yao and B. Van Durme. Information extraction over structured data: Question answering with Freebase. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 956–966, Baltimore, Maryland, June 2014. Association for Computational Linguistics.

[84] S. Zhang, K. Duh, and B. Van Durme. MT/IE: Cross-lingual open information extraction with neural sequence-to-sequence models. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 64–70, Valencia, Spain, Apr. 2017. Association for Computational Linguistics.

[85] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*, 2019.

[86] C. Zhao, M. Walker, and S. Chaturvedi. Bridging the structural gap between encoding and decoding for data-to-text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2481–2491, 2020.

[87] Z. Zhong and D. Chen. A frustratingly easy approach for entity and relation extraction, 2020.

[88] J. Zhu, Q. Wang, Y. Wang, Y. Zhou, J. Zhang, S. Wang, and C. Zong. Ncls: Neural cross-lingual summarization. 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 3054–3064, 2019.

[89] X. Zou. A survey on application of knowledge graph. *Journal of Physics: Conference Series*, 1487(1):012016, mar 2020.

That's all Folks !