# Learning from Noisy Data for Cross Lingual Text Generation in Low-Resource Languages

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

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

by

Kancharla Aditya Hari
2020121010
aditya.hari@research.iiit.ac.in

International Institute of Information Technology
Hyderabad - 500 032, INDIA
June 2024

International Institute of Information Technology
Hyderabad, India

# CERTIFICATE

It is certified that the work contained in this thesis, titled **"Learning from Noisy Data for Cross Lingual Text Generation in Low-Resource Languages"** by **Kancharla Aditya Hari**, has been carried out under my supervision and is not submitted elsewhere for a degree.

Date

Advisor: Prof. Vasudeva Varma

Don't Panic

# Acknowledgments

The path that has led me to this stage has been very long and circuitous, but here I am finally. It would be an understatement to say that it wouldn't be possible without the support and presence of many people. In these few pages, I can thank half of them half as well as I should like and thank half of them less than half as well as they deserve, but it's all I am afforded, and I hope to make the best of it.

First and foremost, I'd like to thank my advisor, Prof. Vasudeva Varma, who presented me with the opportunity to join iREL. Here, I could explore my interests and passions in a supportive environment, the credit for which also goes to you. Your encouragement to explore problems to a deeper and more meaningful degree and your insights and feedback to support such an endeavour is why I can look back at my work and, to a larger extent, my time at IIIT-H with a sense of accomplishment. The culture you fostered at the lab, and your exhortations for our enthusiastic involvement with it is the reason why I came to think of it as *our* lab, why it's the place I love the most in this college, and why I will always miss the weekly meetings. Your belief in collaborative excellence is why I could meet and work with many amazing people. I cannot thank you enough for your support and guidance and for the platform you've provided.

In the same vein, I'd also like to thank Prof. Manish Gupta. The experience of working with you has been incredibly transformative. Your work ethic and dedication towards research are a source of constant inspiration and something I will always strive to emulate. That you can accomplish this whilst never compromising on your family is, to me, a miracle.

I'd also like to thank Prof. Karthik Vaidyanathan and Prof. Nazia Akthar - the professors responsible for the most enjoyable courses I've taken at IIIT-H and whose courses I TAed. More importantly than that, people who inspire me to continue pursuing my ambitions in academia. Every conversation with you has been insightful and delightful, and I hope they continue for a long time to come.

The vibrant spirit at iREL was integral to the experience. To the seniors - Bhavyajeet, Sagar, Pavan, Ankita, Dhaval, Tathagatha, Sumanth, Anshul, Anubhav, the peers - Shivansh, Nirmal, Gokul, and to the juniors - Harshit, Manav, your friendship and collaboration has played a vital role in making my time here as nourishing as it was. There was always someone to turn to for advice and help when I was stuck. The lively discussions in the lab, which were rarely ever restricted to just research or work, made it the home it was. I would be remiss if I didn't

mention the countless hours spent staring at dots playing football with Shivansh, which was often a much-needed break from the drudgery of work. A special thanks to Bhavyajeet, whose serendipitous invitation to join him in his project shaped the bulk of my work, for showing me the ropes, and is still the first person I turn to when I need help with my research.

Beyond the lab, the friends I've made on the way prevented this long journey from ever becoming tedious with the countless rounds around the campus, the hours of conferences in our labs and rooms, and the excursions to DLF and beyond. Prince, VJS, Snehal, Sree - thank you for being there whenever I needed it; most of all, Prince, who has been a patient listener to interminable rants and rambles. I am lucky to have found such an incredible group of people to call friends.

An important shout-out to the Bhilai gang - Aditi, DR, DM, Ketki, Sahay, Sanjana (honorary member). Thank you for being the constants in my life that I cherish greatly.

And most importantly, I want to thank the strongest pillars of moral support and stability - my family. Papa, Ma, Chikku, Nani and Pednaina - thank you for your unconditional love, unending support and guidance, and belief in my abilities. Thank you for being so patient while keeping up with this lengthy journey, where I've often made the wrong turns. That home was only ever a quick phone call away was always a source of comfort. The weekends to wind down in Bangalore, the home-cooked meals at Chaitanyapuri, and the weeks of respite in Bhilai - these were my essential sojourns between the stresses of college. I don't thank you enough for the role you've played.

Reflecting on the years I've spent working towards this thesis is an emotional and overwhelming feeling. At several stages I've doubted if I've made the right decisions or if I'll ever reach the destination I desired. It's because of these people that I can put these doubts aside and be proud of who I've become and where I've reached, and cast an optimistic glance towards the future. There are countless others whom I wish I could name but cannot for the want of space. I thank all of you for your help because I needed all of it.

# Abstract

With Large Language Models (LLMs) and Language models in general becoming a more significant part of our daily content consumption, it is paramount to ensure that languages with fewer resources do not get excluded. As most language models are trained using online data, their performance is usually significantly worse for low-resource languages than languages such as English. This gap in performance leaves speakers of low-resource languages with a handicapped experience of consuming information and participating in online discourse. In recent years, methods have come up that seek to address this resource gap by generating large datasets to enable the training of models for low-resource languages across various tasks. One such task is fact-to-text generation, where cross-lingual generation has gained prominence due to its ability to leverage high-resource languages to augment generation for low-resource languages. However, these works rarely address the noisy nature of synthetically created datasets, which can cause models to hallucinate, reducing their usefulness for factually grounded tasks.

This work investigates various methods and ideas that revolve around carefully using noisy datasets. Methods that account for the noisy nature of data can improve the quality of generation of texts without requiring significant modelling or architectural changes. We leverage techniques such as curriculum learning and, in the process, describe various metrics that can be used to quantify data quality. Our work focuses on cross-lingual fact-to-text generation, and thus, we extend our work to generating factually-grounded text.

We begin our study by using the XAlign dataset. We investigate how curriculum learning can be used to improve the performance of models for the task mentioned above. We experiment with different curriculum schedules and data-ordering metrics using a sharded curriculum learning framework and delineate how different metrics perform under different schedules. We show that curriculum learning outperforms plain, non-curriculum learning-based training using commonly used metrics. We also introduce a novel metric for ordering data - coverage score, which captures the semantic alignment between the input text and reference text. We show that training with data ordered according to coverage score under a gradually refining schedule results in the best-performing model.

Next, we apply these findings to a more challenging setting - long-text generation. To this end, we create a new synthetic dataset using the XAlign dataset and show that previous findings do not apply to this problem setting. We identify the cause of this discrepancy and show that

more than a simple curriculum learning framework is needed here. We denoise the training set using different trusted data sources and show that ordering data based on this noise score and a probabilistic sampling-based curriculum improves performance.

Finally, we conclude our studies by explicitly focusing on reducing hallucinations in long-text generation. We introduce a modular pipeline-based approach with multiple steps to mitigate hallucination during various stages of training. We show that this approach results in sizeable improvements compared to end-to-end training. We also introduce a new evaluation metric for evaluating texts with divergent references, where accounting for the source is also essential.

In summary, this work covers various facets of learning with noisy data for the problem of cross-lingual fact-to-text generation. Synthetically created datasets can bridge the gap between languages, but training models using such datasets is challenging. Through extensive experimentation, we demonstrate several ways to tackle this problem.

# Contents

# List of Figures

# List of Tables

*Chapter 1*

# Introduction

This thesis focuses on the potential of using synthetic data for training natural language generation systems, particularly for low-resource languages. We use cross-lingual fact-to-text generation to evaluate our proposed methods and ideas. This chapter provides an overview of the problem statement. We begin by motivating the problem's importance, then describe the different ideas and avenues tackled. We conclude with a summary of the contributions of this work and give an account of the organization of the chapters.

## 1.1 Motivation

Generative AI has made significant inroads across various problems and domains. While improvements are being made across all modalities - videos, images, speech, text, etc.- it is the last relevant to this thesis. Arguably spearheaded by Generative Pre-trained Transformer (GPT), generative methods represent the cutting edge in various problems within Natural language processing (NLP). In particular, natural language generation (NLG) has captured the zeitgeist. Chatbots and AI companions such as ChatGPT have weaved their way into the workflows of every domain. The backbone of such technologies are Pretrained Language Models (PLMs) - language models trained on vast amounts of data and fine-tuned on diverse tasks. Models trained using this approach exhibit strong performance even on unseen tasks. However, their performance is bounded by the quality of the data they are trained on, which naturally points to their fatal flaw - the skewed availability of data across languages.

### 1.1.1 Scarcity of resources and associated challenges for low-resource languages

Training language models to perform different tasks is usually a two-step process - first, a language model is "pretrained" on vast amounts of data using self-supervised tasks such as masked language modelling, next sentence prediction etc. The data for such tasks is sourced

from the internet with datasets such as Common Crawl. The next step involves "fine-tuning" the models on specific tasks of interest using smaller but higher quality annotated datasets. By definition, such resources for low-resource languages are scant. The data available for even pretraining language models is inadequate for most languages. For instance, consider Wikipedia - the largest free encyclopedia on the internet. Figure 1.1 compares the number of articles hosted on the website in various Indian languages and English, as well as the number of global speakers for each language. It is easily observable that Indian languages are disproportionately underrepresented. For fine-tuning, datasets are often manually curated. However, this process's expensive and time-consuming nature means they are again rarer for low-resource languages. These pitfalls mean that the performance of NLG systems for such low-resource languages continues to lag behind that of English. As access to the internet expands and the internet share of developing countries subsequently increases, it is paramount that this gap is addressed to ensure fair and equitable access to the internet. Bridging this gap would enable equal opportunity to contribute to the internet and participate within its various frameworks.

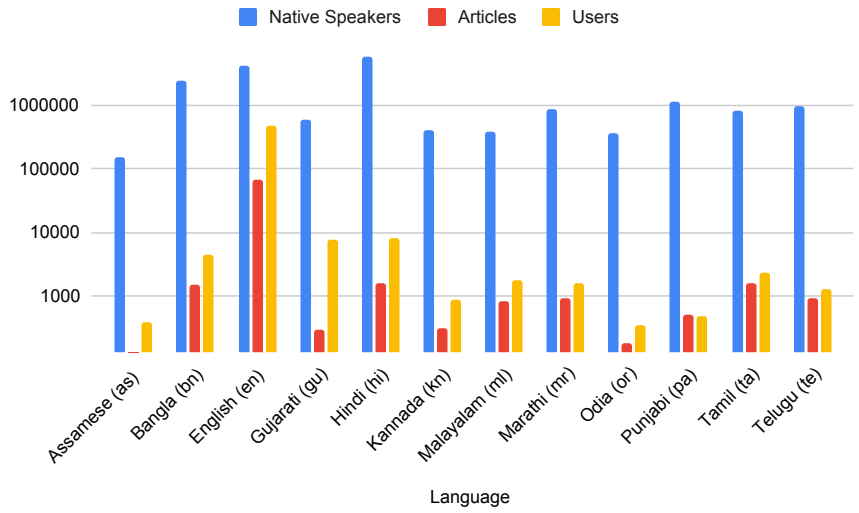



Figure 1.1: Comparison of native L1 speakers of various languages, along with the number of articles and users on Wikipedia

### 1.1.2 Leveraging existing resources to bridge the resource gap

#### 1.1.2.1 Synthetic Data

The expensive nature of curating high-quality datasets necessitates developing alternative methods for generating datasets to fine-tune language models. A prominent avenue of exploration has been automatically generating high-quality datasets. There are two prominent

methods for the generation of such datasets. The first involves using LLMs to generate data. Second, algorithms can be devised to construct the dataset by re-purposing existing resources. Utilization of such data has allowed open-source LLMs to obtain highly competitive results compared to closed-source LLMs like GPT-4. Synthetic data is particularly relevant for low-resource languages, where existing resources are low in quantity and quality, as discussed above. One drawback this approach suffers from, however, is the noisy nature of the constructed dataset. Naive training over such data can exacerbate the tendency of models to "hallucinate" - generating outputs that, while coherent, are not factually grounded or aligned with the input. Thus, measured and careful use is vital to train functional models.

#### 1.1.2.2 Cross-lingual Generation

Cross-lingual generation involves generating text where the output and input text are in different languages. Many tasks can have cross-lingual variants, such as question answering, summarization, named entity recognition, etc. Cross-lingual generation presents an exciting avenue to generate content in low-resource languages by leveraging existing resources in high-resource languages as input data. They also benefit from cross-lingual and multilingual learning, as knowledge gained from performing the task in one language can improve performance in other languages. A natural application of this notion is to utilize high-resource languages to improve the performance of systems for low-resource languages.

## 1.2 Cross-Lingual Fact-to-Text Generation

Data-to-text generation is an essential problem in the domain of NLP. It involves transforming structured data into natural text. Structured data here can be in the form of charts, tables, graphs etc. Utilizing structured data allows for grounded generation of text and can help make complex data easily interpretable. It also makes the generated text more reliable and trustworthy.

Fact-to-text generation is a subproblem of data-to-text generation. Here, knowledge graphs are used as structured data input. Knowledge graphs are representations of some knowledge base where different entities are linked together using various relations. Large knowledge bases such as Wikidata and DBpedia contain information about millions of entities and are important resources for sourcing factually grounded information. F2T is thus naturally motivated - it can allow for the generation of informative and encyclopedic content, thus enriching resources like Wikipedia or generating content for business use cases. It also allows for integrating such knowledge bases into applications like ChatBots and agents.

F2T systems use knowledge graphs as inputs, with the entities and relations verbalized in some language, typically English. Cross-lingual F2T (XF2T) is a variant of this problem

where the generated text language differs from the verbalization language. This problem is the topic of focus for this thesis. As previously discussed, online textual content is predominantly present in English. This is also reflected in the sources of knowledge, which are extensive in English. Resources such as Wikidata and DBpedia encompass information about various topics and domains. More than translation of content from English to low-resource languages is required, as such content is often endemic in nature. Thus, cross-lingual generation here is of vital importance. Resources from high-resource languages such as English can be leveraged to generate content in desired low-resource languages.

## 1.3 Learning With Noisy Data

A major challenge with using automatically generated data, and synthetic data in general, is that they are often noisy. This is a natural trade-off - while human-annotated data is cleaner and of a higher quality, it is difficult and expensive to source. On the other hand, synthetic data can be rapidly generated for a variety of tasks and use cases, but is difficult to control the quality of. Thus, it is important to devise methods that employ careful use of this data. Language models trained on such data are prone to hallucinations - generating factually incorrect content or content that is not grounded in the inputs. This is a significant bottleneck towards adopting systems like LLMs for critical tasks, as it makes them unreliable and untrustworthy. A major component of this thesis revolves around finding methods and algorithms that allow for learning from the informative parts of data while mitigating the effects of noise.

## 1.4 Contributions

The primary contributions of the thesis are summarized below

1. We investigate different methods for generating text for cross-lingual fact-to-text generation that explicitly focus on using noisy data. We experiment with different methods and show that with careful use of data, the performance of models can be improved even with noisy data

2. We experiment with curriculum learning for cross-lingual fact-to-text generation, using various ordering metrics and schedules. We propose novel metrics for ordering data that jointly model both the input and reference text and show that they result in the best performance for this task. We also explore the idea of "denoising" noisy data using trusted data. We also investigate the nature of the chosen metrics and show that quality-based metrics demonstrate promising performance when dealing with noisy data.

3. We construct the XLAlign dataset and propose the cross-lingual fact-to-long text generation problem, a more challenging variant of the XF2T task that is closer to the real-world use case for generating factual and informative content than single-sentence generation.

## 1.5 Thesis Organisation

- In **Chapter 1** (current chapter), we introduce the core premise of this thesis and motivate its importance and utility. We briefly summarize the key problems tackled and the key contributions
- **Chapter 2** presents an overview of current literature for the tasks in focus in this thesis. We highlight the limitations and gaps in literature and contextualize our contributions in light of this.
- In **Chapter 3**, we visit the problem for XF2T, and show that careful use of data using curriculum learning can substantially improve the performance of text generation models. We investigate various ordering metrics and schedules and characterize their properties and performance.
- **Chapter 4** extends the findings of the previous chapter to a more challenging setting by constructing the XLAlign dataset, where the focus is on generating lengthier texts. We revisit curriculum learning methods to train models using this dataset. We explore denoising noisy data using various trusted data sources, including synthetically generated data.
- **Chapter 5** steps away from curriculum learning and instead explores explicit methods for grounding generated text and reducing hallucinations. We experiment with different reward models and propose a novel evaluation metric to study the performance of models when learning with divergent references in a data-to-text setting.
- We conclude with **Chapter 6**, where we summarize our studies' key findings and results. We finally highlight avenues for future exploration that can build on the foundations of this thesis.

*Chapter 2*

# Related work

This chapter examines the current literature on the various problems tackled throughout this thesis. We provide a comprehensive overview of the different methods, datasets, and challenges investigated and studied, allowing us to identify gaps in the literature where improvements are necessary. We focus on an important class of problems that revolve around enabling training models for low-resource languages where the available resources are noisy. Specifically, we focus on data-to-text generation, cross-lingual generation of text, and methods for dealing with noisy data, including an in-depth analysis of curriculum learning.

## 2.1 Fact-to-Text Generation

Data-to-text generation is a problem in NLP, which aims at transforming structured, non-linguistic data, such as tables, knowledge graphs, time series, etc., into a user-consumable form such as natural text [37]. This transformation allows users to ingest and understand complex data easily. The nuance that separates this task from tasks such as machine translation and summarization is the necessity of the data to be not exclusively linguistic [20, 56]. This also excludes modalities such as speech and images from the scope of the problem. The problem finds applications in various domains such as healthcare [54, 52], sports [4], and finance [47, 3], among many others.

In this thesis, we focus on a specific variant of the data-to-text generation problem: fact-to-text generation. It involves transforming structured data in the form of fact triples into natural language [56]. It can also be thought of as verbalizing knowledge graphs [46].

Several datasets have been proposed for this task. A prominent one is WebNLG [19], which contains pairs of RDF triples and human-annotated verbalizations of the text. The RDF triples span across multiple domains such as people, cities, architecture etc. Initially created for English texts, it has since seen multilingual variants with Russian, German, and low-resource European languages like Welsh, Irish, and Breton released [17, 10]. Due to the time-consuming and expensive nature of creating datasets, several automatically created datasets have also

been put forward. One common approach is the alignment of structured entities to natural language text. WikiBio [34] does so for infoboxes, with the approach expanded to different domains [55] and languages [49] in later works. Aligning sentences to RDF triples extracted from knowledge graphs is also a common approach [18, 5]. Other approaches include devising extensive pipelines [14, 28]. However, most of these approaches rely on the assumption that the text and the structured data are in the same language i.e. monolingual, using techniques such as direct string matching to align the entities Cross-lingual fact-to-text generation (XF2T) is a subtask where the input data and the generated natural text are in different languages [1]. This is of particular relevance for low-resource languages, which typically have lower availability of data as it allows input data from high-resource languages such as English to be leveraged to generate text in low-resource languages [10, 62, 61]. Table 2.1 summarizes various data-to-text generation tables.

Recent systems for this task have relied on neural methods. Seq2seq methods have been particularly effective, with many employing models like LSTMs, GRUs, GATs [12, 60, 58] as the encoders and decoders. Methods sometimes involve several preprocessing steps. One such method shown to improve performance is that of delexicalization [13, 42]. Here, slot-value pairs for entities are replaced by placeholders during training and then replaced with the entity names during inference. This contrasts copy-based methods, which use copy mechanisms to directly copy entities from the input. End-to-end methods are also common, such as those used by Dusek et al. [12]. Recently, pretrained language models have also been shown to perform strongly. Chen et al. [8] propose a three-layered approach called TASD which uses pre-trained language models along with explicit steps for table structure understanding and text deliberation, while Ribeiro et al. investigate their efficacy for graph-to-text generation [57]. Various neural approaches have been investigated for cross-lingual fact-to-text generation. Abhishek et al. [1] put forward the XAlign dataset and establish baselines using seq2seq models. Sagare et al. [61] expanded the dataset and investigated multilingual pretraining and fact-aware embeddings. Moussallem et al. [46] used a graph attention network-based encoder and a transformer decoder to verbalize RDF triples in English, German and Russian using the enriched version of the WebNLG dataset [6]. Acknowledging the lack of resources and efforts in low-resource languages, which represent an important use-case of NLG systems, the WebNLG 2023 challenge [10] invited systems for fact-to-text generation for low-resource European languages. Monolingual solutions were proposed for Russian [29], where the knowledge graphs were first translated to Russian and Irish [43], which involved hand-crafting rules. Multilingual solutions ranged from using a combination of NLG+MT [2, 33] or using LLMs to directly generate the outputs without training [40]. Another work in generation for low-resource languages is that of Wang et al. [70], where they use cyclical training to simultaneously train both G2T and T2G models, improving the performance of both.

| Name | Construction | Languages | Format | Size |
|---|---|---|---|---|
| E2E [50] | Manual | en | MR | 50k |
| WebNLG 2020 [16] | Manual | en, ru | RDF | 40k, 17k |
| WebNLG 2023 [10] | Manual | 4 | RDF | 1.6k |
| XAlign [1] | Automatic | 12 | RDF | 550k |
| DART [48] | Automatic | en | Record | 33k |
| WikiBio en [34] | Automatic | en | Record | 728k |
| WikiBio fr-de [49] | Automatic | fr, de | Record | 170k, 50k |
| RotoWire [74] | Automatic | en | Record | 11k |
| ToTTo [51] | Automatic | en | Record | 136k |

Table 2.1: Overview of some popular data-to-text datasets. Here, construction refers to the manner of construction, while format refers to the data representation format. MR stands for meaning representation, RDF stands for Resource Description Framework, and Record implies table records.

## 2.2 Learning from Noisy Data

Due to the expenses and time involved in manually annotating data, automatically generated datasets can easily increase the availability of data for different domains and languages. However, this kind of data is prone to being noisy, and thus, special care is needed to ensure that systems trained using this data do not hallucinate or over-generate. Dusek et al. [12] investigated the impact of semantic noise in training data and found that cleaning the data can decrease the prevalence of semantic noise by 97%. Extensive efforts have thus been dedicated to reducing the impact of noise in training data in various domains. For data-to-text generation, a work closely related to ours is that by Fu et al. [18], who propose a distant supervision framework to enable learning from "partially-aligned" data, a task which they call Partially-Aligned Data-to-Text Generation (PADTG). For this, they estimate the input data's supportiveness for each target word and then apply a supportiveness adapter and rebalanced beam search to control over-generation. Rebalanced beam search has also been explored by Tian et al. [65], where they include information from the source during decoding.

An important line of work for dealing with noisy data involves "denoising" data. Here, the effect of noise during training is mitigated by explicitly modelling methods to quantify or account for the noise in the data. In neural Neural Machine Translation (NMT), partially aligned data can be in the form of automatically aligned texts in different language pairs. Data

selection based methods are a prominent choice, with methods generally using cross-entropy difference (CED) to identify data that closely aligns with a domain with higher availability of data [45, 67]. Inspired by this, Wang et al. [68] extended this approach to select high-quality data from a noisy dataset, with a "clean" dataset used to select data. Denoising the pre-training procedure with quality estimation when reference texts are unavailable has also been investigated with positive results [21]. Relation extraction is another domain where an extensive body of work exists for denoising data. Distant supervision, where a labelled corpus is used to extract relations from a large unlabelled corpus, is an effective method for extracting a large number of resources [44]. However, a dataset constructed with distant supervision is noisy [26], necessitating special care using methods such as denoising. A simple method is that of multi-instance learning, where instead of single candidates, multiple candidates are considered together as a set during the alignment process [59]. Another method for accounting for the noise in the data is ranking sentences that best represent a relationship, with attention mechanisms used to improve the performance [41, 38, 79]. Metrics for evaluating models trained on noisy data are also important. Dhingra et al. [11] note that evaluating divergent references requires source-dependent metrics, propose the PARENT metric and show that it has a higher correlation with human judgement.

## 2.3 Curriculum Learning

Curriculum learning (CL) is based on the assumption that the order of training samples matters while training data. It uses the intuition that going in a specific order, such as from easier to more difficult samples, is useful. This is inspired by how human education curricula are structured - learning a concept like advanced calculus requires knowledge of basic mathematical concepts. For machine learning, this is an alternative to the traditional training routine where training begins with the entire dataset. Bengio et al. [5] showed empirically that curriculum learning has an effect on both the convergence speed and, in some cases, the quality of local minima obtained, and it has since found utility in a variety of problems. It has been shown to be effective across multiple domains, such as computer vision and NLP, in various recent studies [23, 24, 15]. They have been used for problems such as segmentation learning [35, 64], reranking for retrieval [27], clustering [76], multilabel and multiclass classification [22, 36], clustering etc. The main challenges that need to be addressed while designing curriculum learning schemes are deciding the criterion to order the data and choosing the right pacing or sampling function for selecting data while training. For this, various classes of curriculum learning strategies have been identified [69].

The first is that of predefined CL, where the difficulty metric and training schedule is manually defined. These metrics are specific to the task, with measures such as sequence length [53], number of conjunctions [31] etc. used for NLP, number of objects for semantic segmentation

in CV [72], and number of nesting functions for code-related tasks [78]. Training schedules are either discrete or continuous. For discrete schedules, the available samples are readjusted after a fixed number of training epochs, such as the baby step algorithm [5, 63]. Continuous schedules, on the other hand, map the epoch to a scalar value using a competence function [53], which determines the proportion of data available in that epoch. The drawback of this CL strategy is that determining the difficulty metric is challenging and requires exhaustive experimentation. It also often requires domain expertise based on the class of problem. The schedulers and orderings are also often inflexible [69]. Automatic CL methods seek to address these limitations of predefined CL. One such method is self-paced CL, where the training loss is used to determine the proportion of data available for training [32]. This uses the model's performance to determine the difficulty of samples, which results in high uncertainty at the start of training. An alternative approach is a transfer-teacher framework, where a teacher model is used to determine the order of samples [24, 75, 73]. Automatic CL methods do not require human-defined difficulty measures, and are thus domain agnostic. They also allow the scheduler to utilize model feedback, making scheduling dynamic.

For text-based tasks, n-gram frequency, token rarity, and sentence length are some metrics used which are based only on the input or output text [31, 53, 39, 7]. Kocmi and Bojar [31] also use linguistic features such as number of coordinating conjunctions. Metrics such as data uncertainty [81], Damerau-Levenshtein Distance and a soft-edit distance have been used to jointly consider both the input and output [7]. In the context of denoising, metrics that quantify the noise in samples have also been used for learning from noisy data for image captioning [25] and denoising machine translation data [68].

*Chapter 3*

# Curriculum Learning for Cross-Lingual Fact-to-Text Generation

Curriculum learning has been used to improve the quality of text generation systems for various tasks, particularly when learning with noisy data. Their application for cross-lingual fact-to-text generation has yet to be studied. In this chapter, we explore different metrics that can be used to improve the performance of generation systems for cross-lingual fact-to-text generation with noisy data using different curriculum schedules. We propose using a novel metric - coverage score, for ordering samples. We show that using a gradually refining schedule for training results in strong improvements compared to non-curriculum based methods across multiple languages.

## 3.1 Overview

Curriculum learning has been shown to improve the performance of monolingual data-to-text generation systems [7]. While metrics such as sequence length and word rarity are effective, the most significant performance improvement is obtained using metrics that jointly model the input and reference text. The applicability of these approaches to the XF2T problem, however, has not been studied. This problem presents unique challenges for current metrics. Defined for monolingual data-to-text generation, these metrics cannot be generalized to the cross-lingual setting. Furthermore, they also do not account for the noisy nature of data, where more difficult examples can simply be of poorer quality and are thus not a fair representation for the model's performance. We propose novel metrics that consider both factors - the noisy nature of data and the cross-lingual nature of input and output.

Existing works only study schedules based on the notion of increasing difficulty, which does not account for potential noise in the data. In this work, we experiment with "annealing" schedules. Curriculum learning schedules that progressively remove lower-quality examples have previously been used for learning from noisy data in other tasks [68, 25]. The work on the intuition that as the training progresses, the model is refined on examples of higher quality. The utility of this approach bears further investigation for our task.

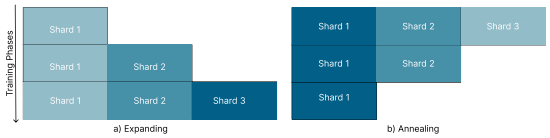



Figure 3.1: Two curriculum schedules used. Each row represents a training phase, with the shade of the block representing a different shard (also indicated with labels on the blocks): a) Expanding schedule with new shards introduced as training progresses and b) annealing schedule with shards removed as training progresses.

Concretely, we make the following contributions:

1. We empirically study the behaviour of different metrics for curriculum learning with two different schedules - an expanding schedule and an annealing schedule.
2. We propose a new quality-based cross-lingual metric - coverage score and show that with an annealing schedule, it results in the best performance compared to other metrics.

## 3.2 Methodology

### 3.2.1 Curriculum Learning Strategy

In this work, we use a predefined curriculum learning strategy. This means that both the ordering metric and the training schedule are manually defined rather than relying on automatic measures. Previous studies use a similar approach [7]. The drawbacks of this approach have been elucidated before - finding effective ordering metrics is complex, and the training schedule can often be inflexible. Therefore, we perform extensive experiments with different ordering metrics and learning schedules. Both source-dependent and source-independent metrics are investigated, and learning schedules that are both gradually refining and gradually expanding, as well as which order the data from easy-to-difficult and easy-to-hard are considered.

### 3.2.2 Shard-based Scheduling

We use a probabilistic curriculum learning strategy similar to the one that Zhang et al [80] for neural machine translation. Here, the training samples are first distributed into distinct *shards* based on the value of the chosen metric. The training process is segmented into different phases, with samples selected from only a subset of shards in a phase. We experiment with two approaches for selecting the shards (Figure 3.1). The first is to begin the training with only the shard with the lowest scores, with more shards added in the subsequent phases in ascending order of scores. We term this the *expanding* approach. The *annealing* approach is based on

works related to learning from noisy data for other NLG tasks. Here, training begins with every shard available in the first phase, and shards are removed in subsequent phases, starting with the shard with the lowest scores. Note that both the data within a shard and the shards themselves are shuffled during a specific phase; the data is not presented in a deterministic order.

We choose this strategy over the competence-based curriculum learning strategies used in previous studies due to its flexible nature. Since it only requires modifying the sampling strategy, it enables plug-and-play experimentation with the expanding and annealing schedules.

### 3.2.3 Curriculum Schedule Metrics

We experiment with two novel metrics designed to jointly model the input and reference text for cross-lingual settings - **n-gram semantic match** and **coverage score**.

#### 3.2.3.1 N-gram Semantic Match (NSM)

Direct comparisons between the input and target text are not possible for cross-lingual data. Thus, we use a cosine similarity-based metric to consider the two jointly. Consider an n-gram $g$ from the set of n-grams $G$ for the target sequence $s$ and the set of lexical tokens in the input facts $F = \{v_1, v_2 \dots v_k\}$. For a lexical token $t$, let $\hat{t}$ represent its embedding. Then, we define the token similarity for a token $g_j$ from $g$ as

$$f(g_j, F) = \max_{v_i \in F} s(\hat{g_j}, \hat{v_i})$$

Then, the similarity of the n-gram $g$ is given as the geometric average of token similarity of each of each of its tokens

$$w(g, F) = \left( \prod_{g_i \in g} f(g_i, F) \right)^{\frac{1}{|g|}}$$

Finally, the NSM score of $s$ is given as the average of the similarity scores over all its n-grams.

$$d_{NSM}(s, F) = \frac{\sum_{g \in G} w(g, F)}{|G|}$$

#### 3.2.3.2 Coverage Score

Here, we introduce the concept of coverage and present a way to quantify it. Dhingra et al. [11] observe that automatically generated datasets are prone to being noisy. A particular problem observed is divergent references, where the reference texts diverge from the input data. This divergence can be in the form of including information that cannot be inferred from the input data or failing to mention information present in the input data. XAlign dataset suffers

| Input Facts | Language | Reference |
|---|---|---|
| <H> sally steele <R> editor <T> vegas rocks! magazine <R> place of birth <T> indianapolis | English | Sally Steele *(née Craig)* was born in Indianapolis and is the *publisher, founder, CEO* and Editor-In-Chief of Vegas Rocks! |
| <H> chanida sutthiruang <R> date of birth <T> 16 july 1993 <R> occupation <T> cricketer | Hindi | चनिदा सुथिरुंग (जन्म 16 जुलाई 1993) एक *थाई* क्रिकेटर है। (Translated: Chanida Sutthiruang (born 16 July 1993) is a *Thai* cricketer.) |

Figure 3.2: Example of divergent references in the XAlign dataset. The text highlighted in blue is not present in the input set of facts

from the same problem. Most instances in the dataset contain information in the text that diverges from the input set of facts. Some examples are shown in Figure 3.2

Coverage score is used to quantify the degree of divergence between the input and reference text. First, we manually annotate 4400 examples from the XAlign dataset using binary labels of partial coverage or complete coverage. Partial coverage means that the reference text diverges from the set of input facts - some information within the sentence cannot be inferred from the sentence. Complete coverage means that the set of input facts and reference text do not diverge. A pre-trained MURIL model [30] is used to train the classifier. Then, confidence scores from the classifier are used as coverage score value for a given pair of input facts and text.

## 3.3 Experimental Setup

### 3.3.1 Baselines

We consider two metrics used in previous literature for monolingual data-to-text generation - sequence length and token rarity [7]. We consider only the target text for scoring samples using these metrics for our experiments. The soft edit distance metric proposed in their work does not generalize to the cross-lingual setting as it relies on exact sequence matching between the input and reference text.

**Length** Length is a natural metric for ordering data based on difficulty. This is because generating longer sentences is more challenging, as errors made early in the decoding process propagate further. For a sequence $s = \{w_1, w_2 \dots w_N\}$,

$$d_{\text{length}}(s) = N$$

**Rarity** This is the product of unigram probabilities of the tokens in a sequence. Based on the intuition that rarer words are more challenging to generate, this implicitly encodes information

about the sequence length and frequency of words in the sequence.

$$d_{\text{rarity}}(s) = -\sum_{k=1}^{N} \log p(w_k)$$

where the unigram probability of $w_i$ is given as $p(w_i)$

### 3.3.2 Data.

We use the XAlign dataset released by Tushar et al. [1] [1]. It contains 0.45M cross-lingual fact-text pairs in 8 languages - English and 7 Indian languages. The dataset was automatically generated by aligning facts represented as RDF triples from Wikidata to sentences from Wikipedia using transfer learning. This results in partially aligned data containing noisy samples, which is suitable for our experiments. The test set contains 5042 manually annotated examples is thus largely devoid of noise.

### 3.3.3 Configurations.

We use the mT5 model [77] for Indian languages and the flan-T5 model [9] for English. The small variants of both models are used [2], containing 300M and 60M parameters, respectively. Flan-T5 was used for English as it is a stronger model specifically for English, and would also help demonstrate the generalizability of the method across different models. Adafactor optimizer with an initial learning rate of 0.001 is used, with a linear decaying schedule. For the curriculum learning framework, the data was divided into 8 shards for all languages. For all experiments, the model with the lowest validation loss was picked

Two sets of experiments were performed for every curriculum metric - training with an expanding schedule and an annealing schedule. We also train a baseline model without curriculum learning.

## 3.4 Results and Analysis

BLEU and chrF++ scores computed using sacrebleu[3] for every experiment are reported in Table 3.1 and in Table 3.2 respectively.

BLEU score measures the similarity of a given text to a reference text by comparing the n-grams in both texts in a position-independent manner.

chrF++ compares a given text to a reference text based on character-level n-gram precision. This results in a tokenization-independent evaluation.

[1] `https://github.com/tushar117/XAlign/`

[2] `https://huggingface.co/google/mt5-small` and `https://huggingface.co/google/flan-t5-small`

[3] `https://github.com/mjpost/sacrebleu`

### 3.4.1 Curriculum Metrics

Figure 3.4 shows the distribution of the different metrics experimented with.

It can be observed that length and rarity display similar distributions, and NSM and coverage score display similar distributions. We characterize length and rarity as difficulty-based metrics and NSM and coverage score as quality-based metrics. We argue that difficulty-based metrics are unsuited for noisy data as they conflate noise with difficulty. Longer sequences will tend to have more noise, meaning that latter iterations will train on largely noisy data, reducing the performance of the models. In line with this rationale, training models using difficulty-based metrics result in better performance in general.

### 3.4.2 Annealing and Expanding schedule

**(1)** For all languages, it can be observed that training with a curriculum learning strategy outperforms non-curriculum based training. Considering the best curriculum learning approach for each language, curriculum learning based approaches result in an average 3.57 BLEU improvement. Ordering data based on coverage score with an annealing schedule yields the best-performing model across all languages, outperforming non-curriculum training by 2.12 BLEU and the next best curriculum training strategy - the sequence length metric with an expanding schedule by 0.98 BLEU. However, the former performs significantly worse for Bengali, with an 11.16 BLEU deficit. In fact, non-curriculum training outperforms this approach for the language by 9.91 BLEU.

**(2)** We also observe another interesting trend in the performance of the metrics under different schedules - length and rarity perform significantly better with an expanding schedule, while coverage and NSM perform better with an annealing schedule. This points to the difference in nature of the two metrics. Sequence length and token rarity are based only on the difficulty and do not consider the noisy nature of the data. As the training phase progresses, samples with lower scores are removed in an annealing schedule. For the metrics above, this means that only the longer sequences and sequences with rarer words are available, which are likelier to be noisier. Hence, training with an annealing schedule results in a catastrophic degradation in performance. On the other hand, NSM score and coverage quantifies the degree of alignment of the reference text to the input facts. As the training progresses, the noisier examples are removed and the model is refined on only the cleanest examples. To ensure the robustness of the implications, we also evaluate the performance of the systems with chrf++ scores. As with BLEU, the best-performing model is obtained using coverage score to order the samples using an annealing schedule. For 6 of the eight languages, the best model is obtained using coverage score to order the data and an annealing schedule. The best model for every language is obtained using curriculum learning. Bengali once again results in anomalous performance,

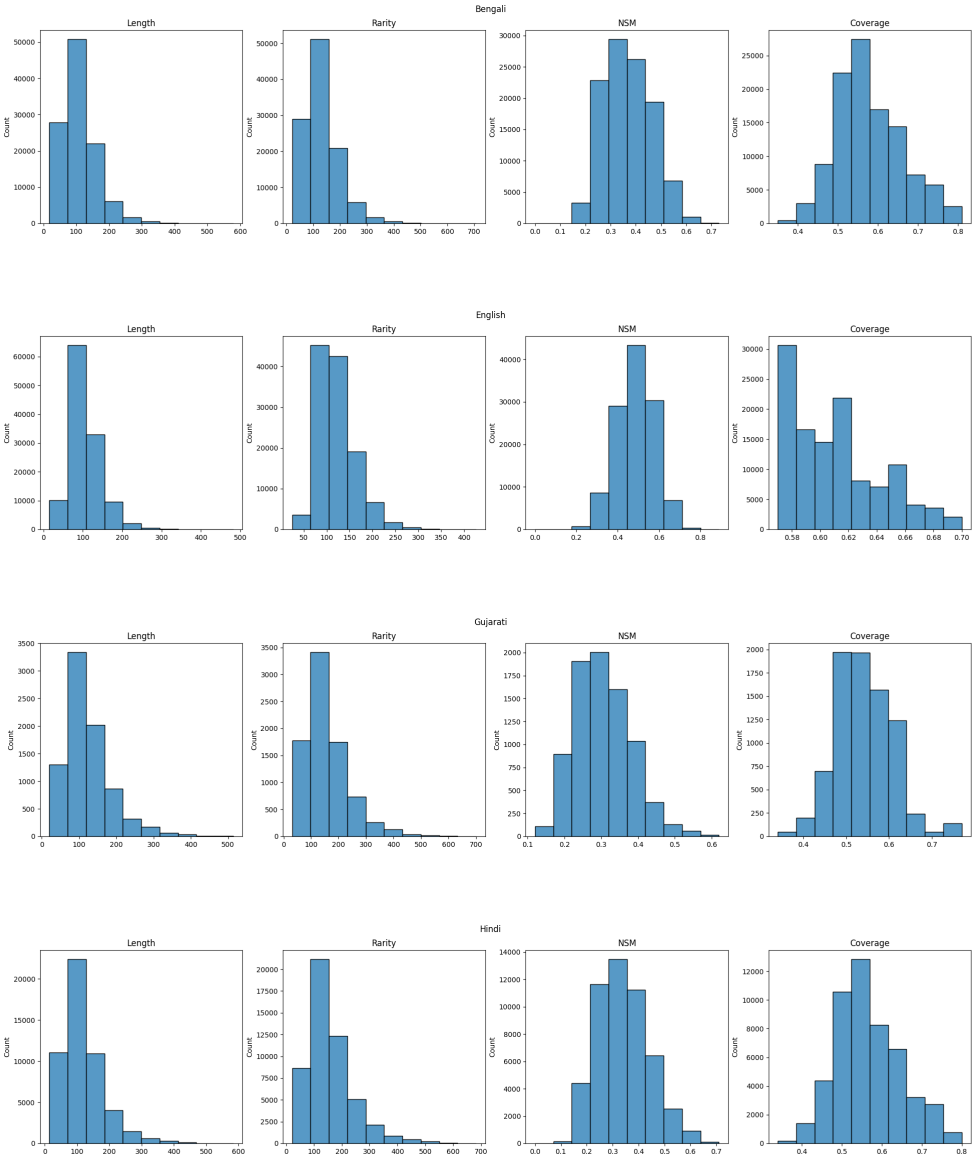



Figure 3.3: Distribution of scores for sequence length, word rarity, NSM, and coverage score for every language for train set (Bengali, English, Gujarati, Hindi).

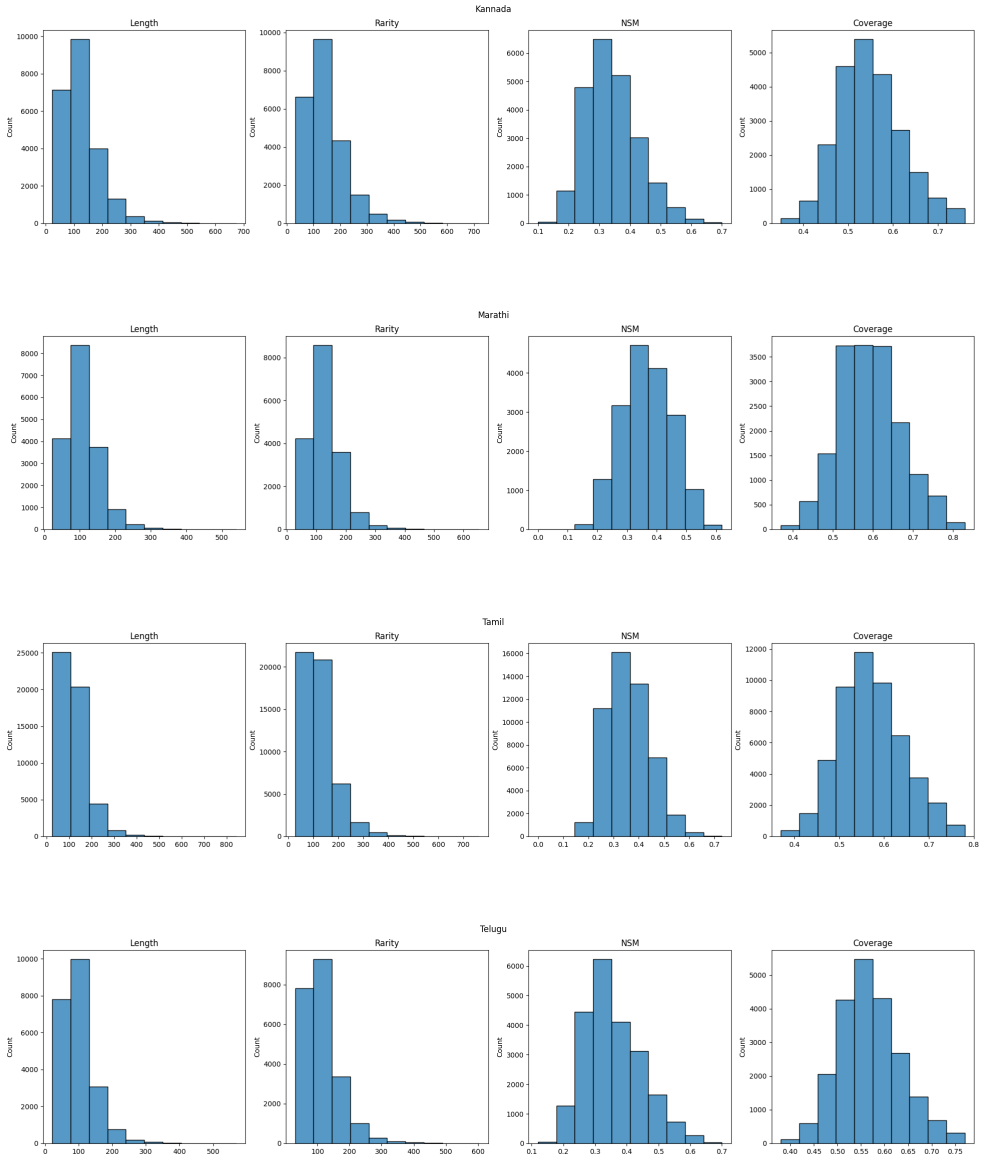



Figure 3.4: Distribution of scores for sequence length, word rarity, NSM, and coverage score for every language for train set (Kannada, Marathi, Tamil, Telugu).

| **Method** | | **bn** | **en** | **gu** | **hi** | **kn** | **mr** | **ta** | **te** | **Average** |
|---|---|---|---|---|---|---|---|---|---|---|
| Non-CL | | 58.71 | 46.31 | 14.63 | 39.04 | 5.55 | 22.61 | 20.75 | 10.78 | 27.30 |
| **Metric** | **S** | | | | | | | | | |
| Length | E | **61.96** | 47.29 | **15.6** | 41.79 | 5.07 | 20.08 | 18.49 | 9.25 | 27.44 |
| | A | 27.36 | 6.94 | 3.55 | 5.82 | 1.6 | 6.45 | 4.99 | 2.63 | 7.42 |
| Rarity | E | 61.51 | 49.63 | 10.96 | 38.09 | 7.23 | 22.9 | 22.54 | 6.9 | 27.47 |
| | A | 24.64 | 6.83 | 4.58 | 5.08 | 1.48 | 5.94 | 5.08 | 2.24 | 6.98 |
| EP | E | 28.42 | 47.19 | 8.4 | 41 | 3.35 | 16.77 | 20.1 | 6.14 | 21.42 |
| | A | 49.7 | 39.88 | 11.79 | 40.85 | 5.21 | 27.77 | 17.56 | 11.34 | 25.51 |
| Coverage | E | 35.48 | 46.81 | 4.92 | 12.63 | 1.71 | 14.62 | 20.16 | 5.79 | 17.77 |
| | A | 50.8 | **47.37** | 15.06 | **43.78** | **10.32** | **31.26** | **24.55** | **12.23** | **29.42** |

Table 3.1: Results for different metrics with different schedules. Non-CL represents non-curriculum based training. The best score for every language is highlighted in bold. All scores are computed on the XAlign test set using the model with lowest validation loss. Column **S** stands for schedule, where E is expanding and A annealing.

with the coverage score based model performing worse than the non-curriculum learning based model.

## 3.5 Conclusion

We study the performance of curriculum learning in the context of cross-lingual fact-to-text generation with noisy data. We experiment with metrics used in previous studies and propose a new metric suitable for the cross-lingual problem that results in a substantial improvement in performance. We also show that different schedules are suited for different metrics, with metrics like sequence length and word rarity performing better with an expanding curriculum, while metrics like NSM and coverage score performing better with an annealing schedule. Our experiments show that a curriculum learning based approach with an annealing schedule based on coverage score leads to the best-performing system.

| **Method** | | **bn** | **en** | **gu** | **hi** | **kn** | **mr** | **tm** | **te** | **Average** |
|---|---|---|---|---|---|---|---|---|---|---|
| Non-CL | | 74.48 | 62.36 | 37.89 | 64.49 | 35.52 | 50.84 | 54.49 | 43.32 | 52.92 |
| **Metric** | **S** | | | | | | | | | |
| Length | E | **79.26** | **64.85** | 35.84 | 63.93 | 32.24 | 46.24 | 53.09 | 39.47 | 51.87 |
| | A | 43.32 | 58.3 | 23.3 | 31.81 | 22.8 | 32.16 | 37.9 | 26.2 | 34.47 |
| Rarity | E | 74.54 | 64.12 | 34.2 | 62.12 | 33.92 | 48.83 | 54.24 | 37.64 | 51.20 |
| | A | 42.92 | 56.43 | 26.65 | 30.01 | 21.64 | 31.31 | 38.57 | 25.72 | 34.16 |
| EP | E | 53.54 | 63.69 | 31.73 | 64.33 | 27.95 | 44.16 | 51.61 | 34.28 | 46.41 |
| | A | 65.68 | 64.31 | 35.16 | 63.62 | 33.72 | 52.23 | 53.26 | 42.22 | 51.28 |
| Coverage | E | 63.79 | 63.55 | 25.78 | 42.57 | 25.64 | 39.49 | 52.23 | 33.04 | 43.26 |
| | A | 71.33 | 63.24 | **38.24** | **65.73** | **39.45** | **52.84** | **58.85** | **43.45** | **54.14** |

Table 3.2: Results for different metrics with different schedules measured using chrF++. Non-CL represents non-curriculum based training. The best score for every language is highlighted in bold. All scores are computed on the XAlign test set using the model with lowest validation loss. Column **S** stands for schedule, where E is expanding and A annealing.

*Chapter 4*

# Denoising Long Text Generation with Curriculum Learning

In the previous chapter, we demonstrated methods for learning from noisy data for cross-lingual fact-to-text generation using the XAlign dataset. This dataset, however, contains only single-sentence texts, which is not representative of the difficulties faced in generating encyclopedic texts, which are generally longer and more involving. Thus, in this chapter, we focus on the problem of cross-lingual fact to *long text* generation. We describe the construction of a new dataset for this task, as well as establish baselines using curriculum learning.

## 4.1 Overview

Verbalizing knowledge graphs is an important problem with wide-ranging applications, given the diverse variety of data that can be represented using knowledge graphs. One such use case is the generation of informative or encyclopedic content, which allows for the rapid enrichment of online ecosystems. As discussed previously, cross-lingual generation represents an important avenue due to its potential to bridge the information gap between high-resource and low-resource languages. However, works for cross-lingual fact-to-text generation for longer texts is limited. Existing datasets use machine translation systems to translate content from high-resource languages. This approach has the drawback of not covering knowledge endemic to low-resource languages, necessitating native generation. Works which address this are limited to the generation of short, single-sentence texts. While this represents an important step towards enriching resources for these languages, it is still far from the use cases of generating complete paragraphs and even articles

Applying curriculum learning to generate longer sequences requires careful consideration of the characteristics of the data. In the previous chapter, we empirically observed that a gradually refining training schedule with coverage score as the ordering metrics performs the strongest compared to alternatives. The annealing-based approach relies on the intuition that as the training progresses, the model is only fine-tuned on the highest quality examples. However, it is also important to ensure that the model learns from diverse sequence lengths at all stages in

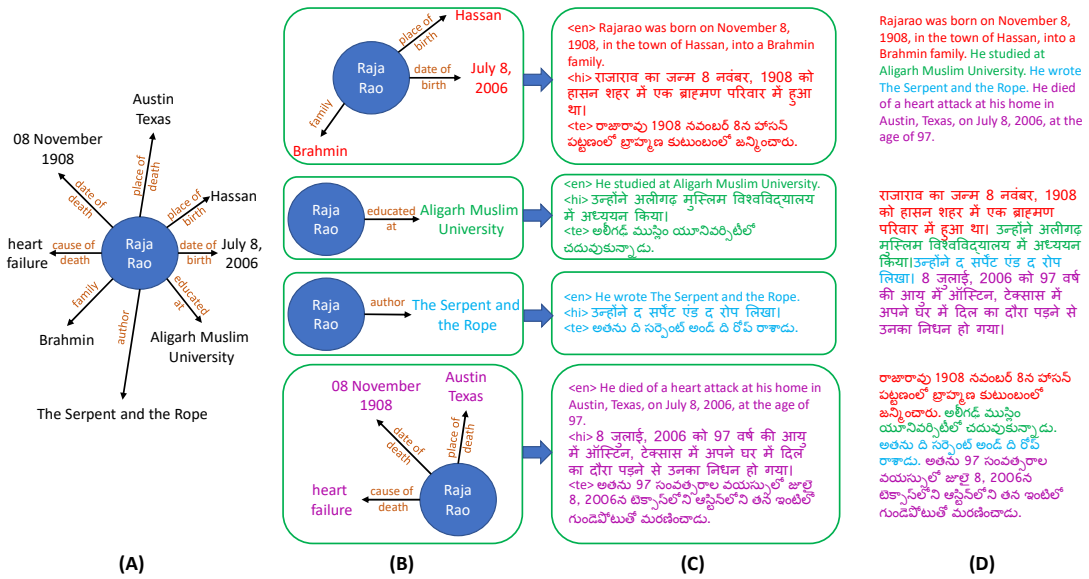



Figure 4.1: An example of cross-lingual fact-to-long text generation for Hindi, English and Telugu. A) Knowledge graph about an entity with various relations B) Various relations in the knowledge graph that convey disparate facts C) Verbalization of each set of facts into different languages D) Concatenation of the different facts into a cohesive passage

the case of longer sequences. At the same time, coverage score as a metric does not cohesively model the longer sequences; instead, it relies on sentence-level deconstruction of the data to quantify their quality.

To address this, we instead apply findings from problems such as machine translation with neural data to "denoise" the data. Even noisy samples contain useful information; if the extent of this information is quantified, this measure can be used to order the samples. Previous studies have used small trusted datasets to denoise large synthetically generated and noisy datasets. In summary, we make the following contributions through this chapter -

- We describe the construction of a new dataset to aid the generation of longer texts from knowledge graphs.

- We experiment with different curriculum learning strategies to generate longer texts. We use a probabilistic sampling strategy to ensure diverse training samples during every stage of the training and "denoise" data using trusted data sources to quantify the quality of a data point.

| Language | Train | | | Val | | | Test | | |
|---|---|---|---|---|---|---|---|---|---|
| | Count | coh | cov | Count | coh | cov | Count | coh | cov |
| as | 799 | 0.57 | 0.55 | 159 | 0.64 | 0.60 | 111 | 0.69 | 0.63 |
| gu | 901 | 0.60 | 0.55 | 179 | 0.73 | 0.60 | 121 | 0.72 | 0.65 |
| or | 1,742 | 0.54 | 0.57 | 348 | 0.61 | 0.64 | 237 | 0.62 | 0.67 |
| kn | 2,026 | 0.59 | 0.56 | 404 | 0.68 | 0.62 | 273 | 0.71 | 0.66 |
| te | 2,820 | 0.57 | 0.56 | 563 | 0.64 | 0.62 | 379 | 0.69 | 0.66 |
| mr | 5,394 | 0.86 | 0.58 | 1,077 | 0.92 | 0.67 | 722 | 0.93 | 0.72 |
| pa | 5,454 | 0.63 | 0.54 | 1,085 | 0.72 | 0.61 | 731 | 0.76 | 0.65 |
| ml | 8,363 | 0.61 | 0.57 | 1,671 | 0.70 | 0.64 | 1,117 | 0.76 | 0.67 |
| hi | 9,266 | 0.75 | 0.57 | 1,850 | 0.85 | 0.66 | 1,239 | 0.86 | 0.70 |
| ta | 10,026 | 0.67 | 0.56 | 2,004 | 0.75 | 0.63 | 1,340 | 0.80 | 0.68 |
| bn | 14,858 | 0.56 | 0.58 | 2,968 | 0.66 | 0.65 | 1,984 | 0.74 | 0.69 |
| en | 32,176 | 0.63 | 0.60 | 6,427 | 0.69 | 0.64 | 4,292 | 0.73 | 0.66 |

Table 4.1: Counts, average coherence (coh) scores, and average coverage (cov) scores for all languages across the training, validation and test splits in the proposed dataset

## 4.2 Dataset Construction

We utilize the XAlign dataset [1], a dataset with pairs of facts and reference texts, to construct the XLAlign dataset where the focus is on generating longer texts. To accomplish this, we concatenate sentences with the same entity as the subject to obtain paragraph-level texts. The union of the set of facts for every sentence serves as the new input. The order of sentences is determined by their relative position with the original article. Statistics regarding the dataset are shown in Table 4.1, while the number of sentences is shown in Figure 4.2

The testing and validation splits for the dataset are also generated automatically. A naive, randomized split of the complete dataset would result in a low-quality test and validation set. To address this, coherence and coverage scores were obtained for every data point to create a high-quality dataset. A combination of these scores and the number of sentences was used to partition the dataset into training, validation and test splits in the ratio of $7.5 : 1.5 : 1$. The relative ratios of texts of different lengths were consistent across all splits.

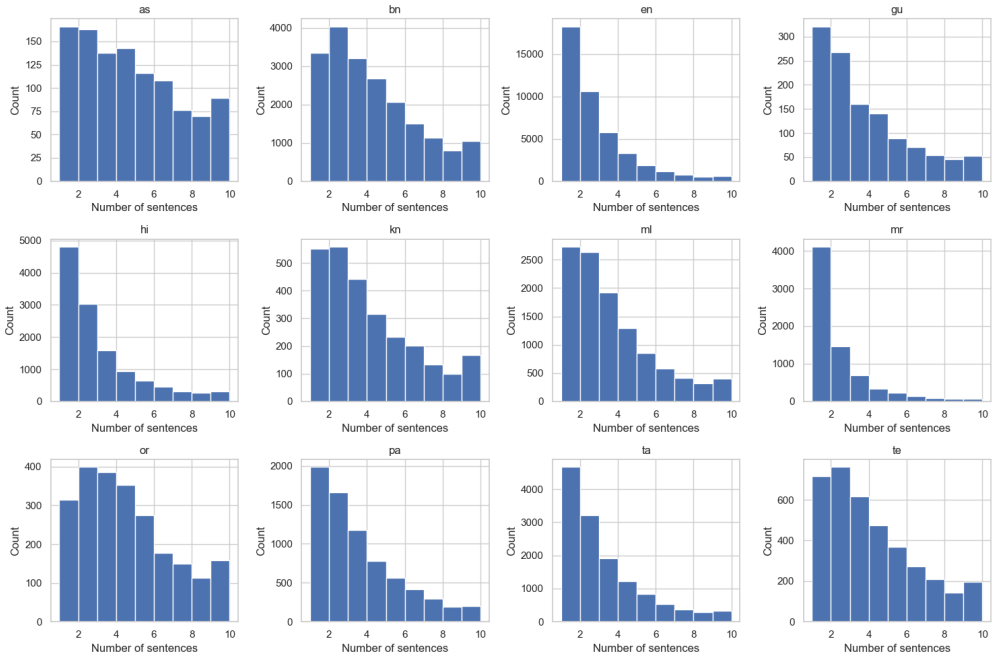



Figure 4.2: Distribution of the number of sentences across all languages in the dataset

### 4.2.1 Coherence

Coherence represents the fluency of a piece of text; it is the quality of being logical and consistent. Since two consecutive sentences in our constructed dataset may not necessarily appear consecutively in the original text, coherence allows us to quantify how well the text is structured and identify the high-quality examples.

We train a classifier to help quantify coherence for our use-case. For this, we create a synthetic dataset of sentence pairs from high-quality Wikipedia articles in the languages under consideration. Positive samples are pairs that appear in-order in the original text. In contrast, negative samples are created by shuffling sentence pairs from within the same section to avoid the problem of trivial negatives. A MURIL model is used to train the classifier. The performance of the classifier is shown in Table 4.2

For a given pair of sentences, the confidence score of this trained classifier is treated as the coherence score for the example. For a paragraph, the average of the coherence scores of every pair of consecutive sentences is used as the coherence score. Note that a coherence score of 1 is ascribed to single sentences. The distribution of coherence scores is shown in Figure

### 4.2.2 Coverage

We introduced the concept of coverage score in Chapter 3. We train a classifier on manually annotated pairs of facts and sentences with partial or complete coverage labels. Then, this

| Language | Recall | Precision | F1 |
|---|---|---|---|
| Assamese (as) | 0.83 | 0.70 | 0.76 |
| Bangla (bn) | 0.84 | 0.66 | 0.74 |
| English (en) | 0.65 | 0.73 | 0.69 |
| Gujarati (gu) | 0.82 | 0.70 | 0.75 |
| Hindi (hi) | 0.86 | 0.54 | 0.67 |
| Kannada (kn) | 0.87 | 0.50 | 0.63 |
| Malayalam (ml) | 0.85 | 0.59 | 0.70 |
| Marathi (mr) | 0.89 | 0.55 | 0.68 |
| Odia (or) | 0.79 | 0.72 | 0.75 |
| Punjabi (pa) | 0.86 | 0.53 | 0.66 |
| Tamil (ta) | 0.81 | 0.65 | 0.72 |
| Telugu (te) | 0.82 | 0.76 | 0.79 |

Table 4.2: Performance of the coherence classifier trained using featured Wikipedia articles

classifier's confidence scores are used to measure the degree of divergence between the reference text and the input set of facts. This measure was used to ensure that the validation and test splits contain cleaner examples with minimal divergence.

## 4.3 Methodology

### 4.3.1 Curriculum Learning Strategy

We make significant changes to the curriculum learning framework to adapt it to the context of longer texts with multiple sentences. In the previous chapter, we used a predefined CL strategy with a simple shard-based scheduler. While extensive experimentation was performed to address the typical pitfalls of this approach, it is also important to explore automatic CL methods. Similar to the approach used by Wang et al. [68] and Moore et al. [45], we use a transfer teacher based method. Two language models are used to determine the quality of samples. This approach does not require crafting metrics specific to tasks, such as the coverage score introduced in the previous chapter. For scheduling data, the simple shard-based strategy makes way for a weighted sampling strategy. This results in a more flexible approach with greater diversity of samples during training.

### 4.3.2 Quantifying Noise in Noisy Data

Noise refers to irrelevant or meaningless information in the training data that results in models generalizing to incorrect patterns or hindering their ability to identify patterns. In our dataset, this can be in the form of candidates where the alignment is incorrect or where the sentence contains extra information. Some examples are shown in Figure 3.2. A key observation to make is that the noisy samples still contain useful information that the model can leverage to identify how the facts can be verbalized. We build on this intuition to motivate the denoising problem. Given a small trusted dataset and a larger noisy dataset, we devise a method to identify samples from the noisy dataset that are closer in distribution to the trusted and presumably cleaner dataset.

Our denoising algorithm is similar to the ones used in previous works for denoising synthetic data [25, 68]. Assume that we have a model $M_\theta$ parameterized by $\theta$ that given an RDF triple $x$ and sentence $y$ outputs probability $p(y|x,\theta)$ that the sentence corresponds to the RDF triple. Then, noisy log probability of the pair $(x, y)$ can be computed as

$$L_{p(y|x,\theta)} = log(p(y|x,\theta))$$

Consider two models - a noisy model $M_{\hat{\theta}}$, and a denoised model $M_\theta$ which outputs a more accurate probability distribution. Then, the quality of a given pair $(x, y)$ can be quantified as

$$u = quality(x, y, \theta, \hat{\theta}) = L_{p(y|x,\theta)} - L_{p(y|x,\hat{\theta})}$$

A positive score means that the pair is more likely according to the denoised model than the noisy model, indicating that it is of higher quality. To estimate $p(y|x,\theta)$, we use seq2seq language models, specifically the t5-small model. Given two datasets - $D$, a small, trusted dataset and $\hat{D}$, a larger, noisy dataset, a model is first trained on the noisy data $\hat{D}$ to obtain the noisy model. This model is then further trained on the smaller, trusted dataset $D$ to obtain the denoised model.

### 4.3.3 Trusted Data Sources

The proposed method for denoising data requires a trusted data source. We experiment with two possibilities for sourcing this data.

#### 4.3.3.1 LLM-based generation

LLMs have been shown to perform on par with human annotators for various tasks [71, 66]. We first experimented with the feasibility of using this to generate a small, trusted dataset. Generating high-quality data with LLMs requires careful prompting. We experimented with two prompting techniques.

**1. Few-shot Prompting.** Few-short prompting leverages in-context learning to perform novel tasks with LLMs. Here, the model is provided with a few examples of the task to be performed. The template used for this technique is given below. We prompt the model with five examples.

```
You are WikiGen, an agent that can automatically generate content about
↪ prominent people for Wikipedia, given a set of facts about them.
Person: <>, Facts: <>
Sentence: <>
Person: <>, Facts: <>
Sentence:
...
```

**2. Chain-of-Thought Prompting.** Chain-of-thought prompting involves making the LLMs perform a series of reasoning steps before generating the final output. This allows LLMs to perform more complex tasks. The reasoning steps are demonstrated through exemplars which enable reasoning abilities. For our use case, we break down the task into a series of single-sentence construction steps. We first collect facts into "clusters" and then generate a sentence for each cluster.

```
You are WikiGen, an agent that can automatically generate content about
↪ prominent people for Wikipedia given a set of facts about them. Generate
↪ the content following these instructions -
1. Identify facts from the complete set of facts that belong together. Call
↪ this a fact cluster
2. Generate a sentence for the identified fact cluster.
3. Repeat till all facts are part of a fact cluster.

Example:
Person: <>, Facts: <>

Response:
Number of fact clusters: <>
```

```
Fact cluster 1: <>
Sentence: <>
Fact cluster 2: <>
Sentence: <>

Final text:
```

#### 4.3.3.2 Leveraging related datasets

An alternative formulation is to consider the trusted data source as an in-domain dataset and the noisy dataset as an out-of-domain dataset. Therefore, we can use an existing dataset proposed for a similar task as the trusted data source. We use the human-annotated split of the XAlign dataset as the trusted dataset for our task. We ensure that the test set of our constructed dataset does not share any entities with the selected dataset. This results in a dataset with 6880 samples across the 12 languages.

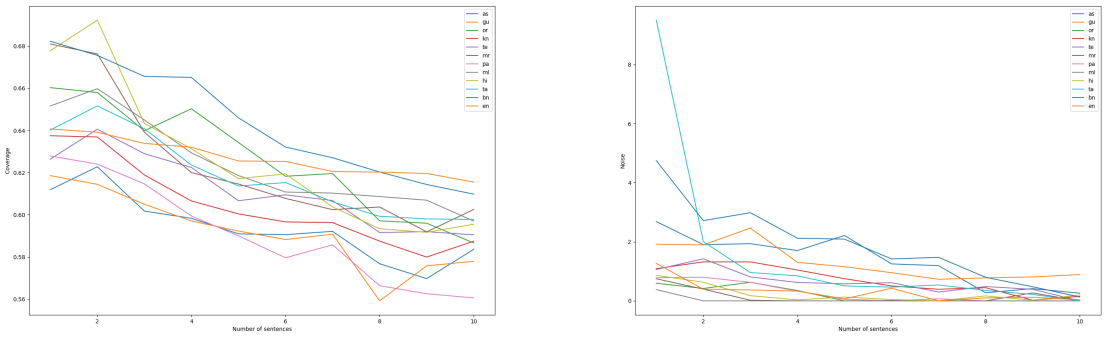
### 4.3.4 Probabilistic Sampling for Sharded Training

In the previous chapter, we used a sharded training regime where instances are assigned to shards based on the value of the chosen ordering metric. With an annealing schedule, which yielded the best performing model, training begins with every shard available, with lower quality shards removed as it proceeds. However, in the case of long text generation, where generation of longer texts is the priority, it is also important to ensure that later training iterations contain diverse examples in terms of the length of the sequences. Figure 4.3 plots the highest score for all samples with a given sentence length. Clearly, as the number of sentences increases, the scores trend downward. Since longer sequences will tend to have lower quality scores, the training process can become prone to catastrophic forgetting if the longer sequences are eliminated early on.

Thus, similar to the approach used in [25], we sample instances for every iteration using a soft-step function to introduce randomness. Without this, the sampling can be treated as a step-function based on the threshold for every iteration. Concretely, given the threshold $T_i$ for the $i^{th}$ iteration, a sample $j$ with quality score $u_j$, the sampling probability is given as -

$$f(u_j; T_i; s) = \frac{1}{2}\left[1 + tanh\left(\frac{u_j - T_i}{s}\right)\right]$$

s is a hyperparameter that controls the smoothness of the step function. This is a smooth step function centred around $T_i$. As $s \to 0$, the step function becomes an ideal step function. The



(a) Coverage score (b) Noise score

Figure 4.3: Highest value of A) coverage score and B) noise score by sentence length across all languages. As the number of sentences increases, the highest value decreases

behaviour of the sampling function for different values of $s$ when $T_i = 0.5$ is shown in Figure 4.4

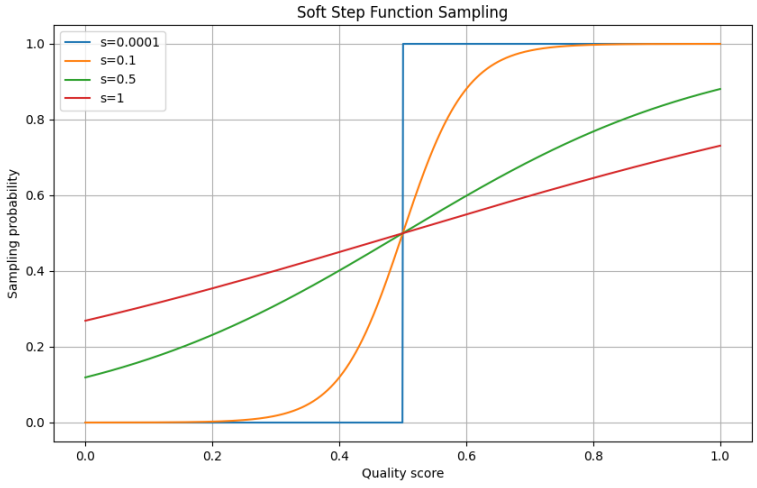



Figure 4.4: Soft step function with $T_i = 0.5$ for different values of $s$. As can be seen, the function approaches an ideal step function for smaller values of $s$. To allow for the inclusion of lower-quality examples in latter iterations, $s$ needs to be larger.

## 4.4 Experimental Setup

### 4.4.1 Baselines

We compare the noise metric against two metrics used in the previous chapter - **sentence length** and **coverage score**. Both metrics resulted in the best-performing models for some languages in previous experiments.

To validate the probabilistic data selection strategy for the curriculum learning based approaches, the sharded strategy as described in the previous chapter was also experimented with. To reiterate, the sharded strategy involves dividing the data into equally sized shards based on the value of the chosen metric. Additionally, we also train a model without curriculum learning as a baseline.

### 4.4.2 Data

The experiments are performed on the XLAlign dataset, the construction of which is described above. For the denoising based curriculum learning, the human annotated split of the XAlign dataset is used as the trusted dataset, with entities which are present in the XLAlign test set removed to prevent data leak.

### 4.4.3 Configurations

The mT5 model [77] was used for training models for all languages and experiments. The small variant with 300M parameters was used. Adafactor optimizer with an initial learning rate of 0.001 is used, with a linear decaying schedule.
For curriculum learning, similar to the previous chapter, 8 training iterations were performed based on the sampling strategy described above.

For all experiments, the model with the lowest validation loss was picked to evaluate on the test set.

## 4.5 Results and Analysis

We use BLEU score and chrF++ to evaluate the performance of the different approaches. BLEU score measures the similarity of a given text to a reference text by comparing the n-grams in both texts in a position-independent manner. chrF++ compares a given text to a reference text based on character-level n-gram precision. This results in a tokenization-independent evaluation.

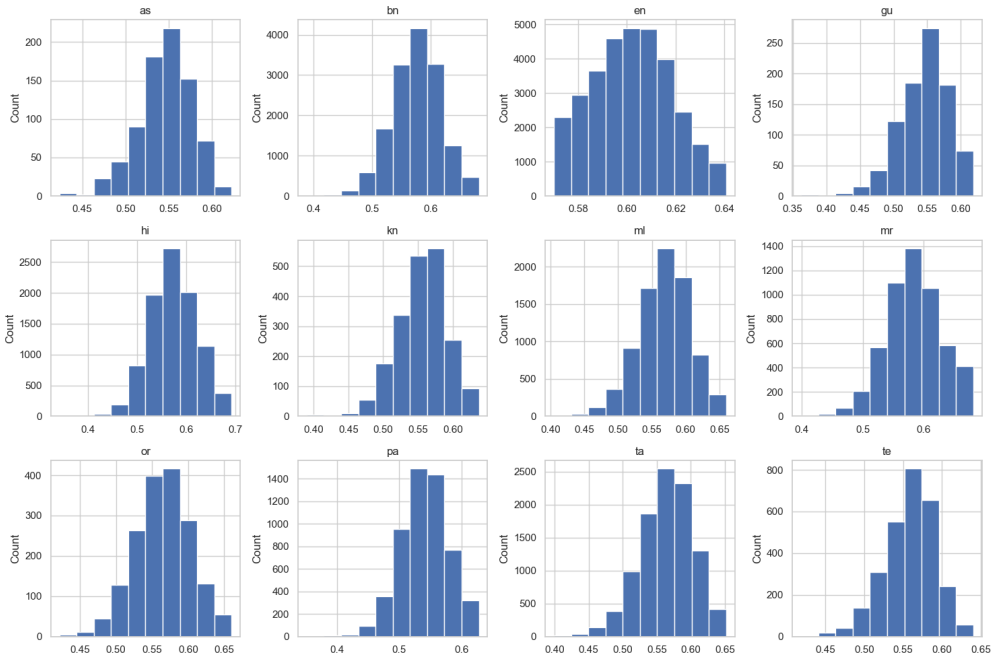



Figure 4.5: Distribution of coverage scores for all languages

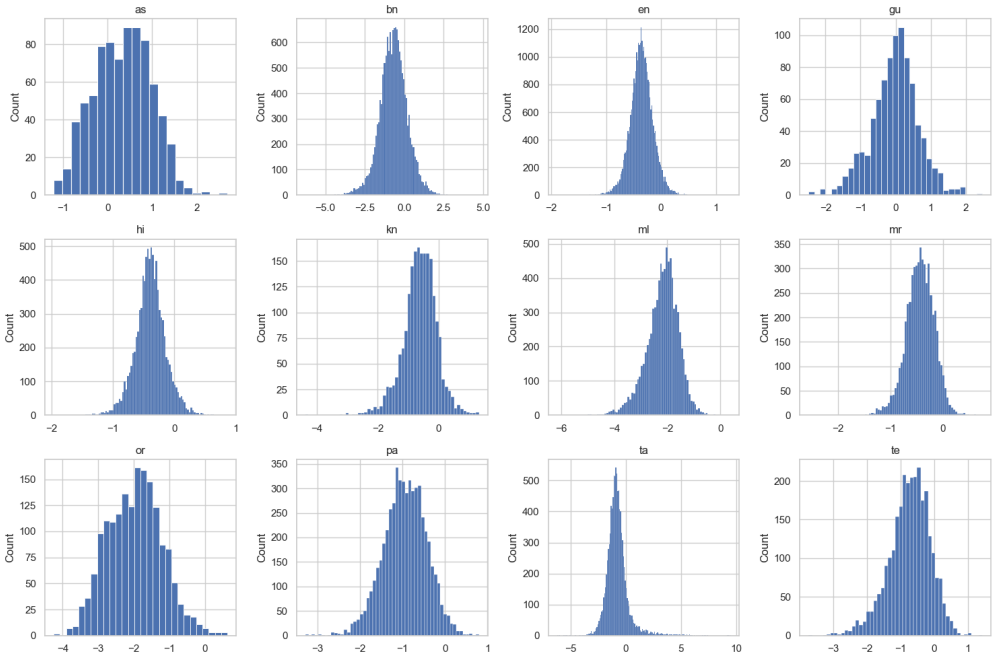
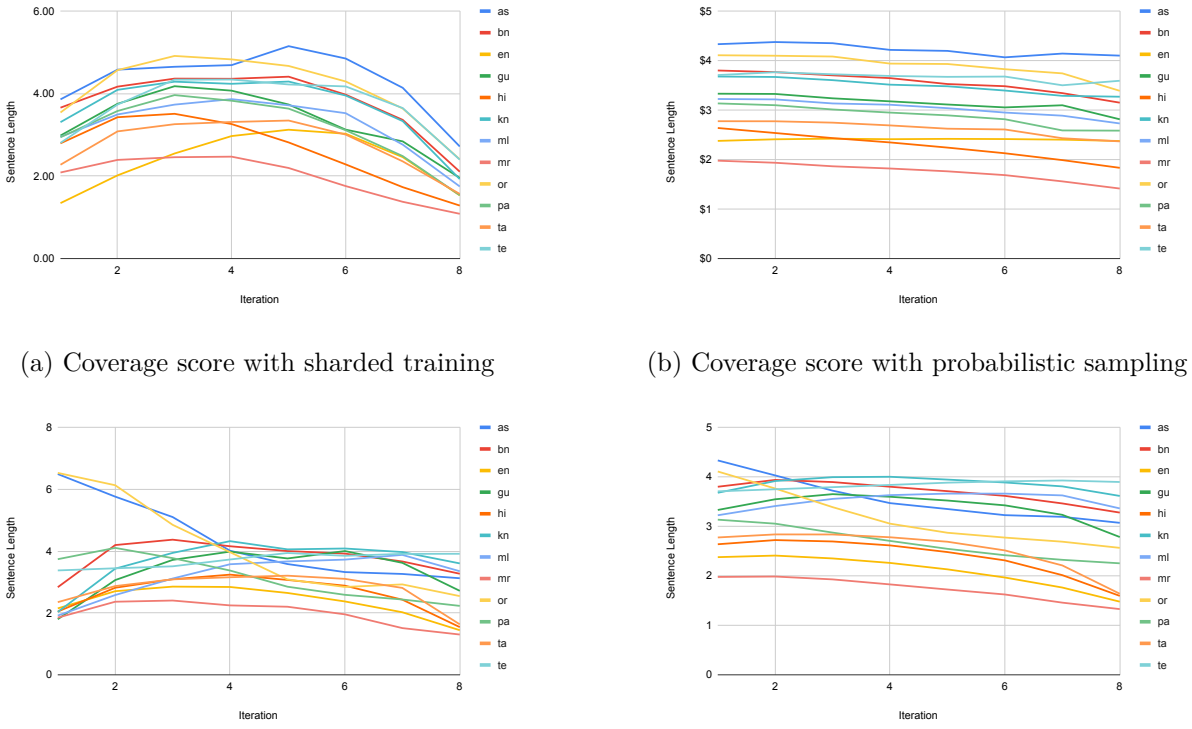



Figure 4.6: Distribution of noise scores for all languages

### 4.5.1 Curriculum Learning Metrics

The distribution of the coverage scores and noise scores is shown in Figure 4.5 and 4.6, respectively. The average Kendall-Tau rank correlation for the two orderings across all languages is 0.12, indicating a low degree of correlation. This means that both metrics capture different aspects while ordering the data.

### 4.5.2 Probabilistic Sampling




(a) Coverage score with sharded training

(b) Coverage score with probabilistic sampling

(c) Noise score with sharded training

(d) Noise score with probabilistic sampling

Figure 4.7: Average sentence length by iteration for coverage score using a) sharded training and b) probabilistic sampling and noise score using c) sharded training and d) probabilistic sampling

To compare the impact of probabilistic sampling, we train models using the previous sharded schedule and compare them against this approach. The BLEU and chrF++ scores for both can be seen in Table 4.3 and Table 4.4. Clearly, probabilistic sampling results in a significant improvement in performance. To further emphasize its impact, we compare the average sentence length in every training iteration for both schemes, which can be seen in Figure 4.7. With a simple sharded training routine, the latter iterations contain, on average, significantly shorter

| | | Sharded | | Probabilistic | |
|---|---|---|---|---|---|
| Language | Base | Coverage | Noise | Coverage | Noise |
| as | **6.69** | 5.56 | 4.56 | 5.36 | 6.42 |
| bn | 26.04 | 21.35 | 22.23 | 21.54 | **26.70** |
| en | **27.06** | 26.51 | 23.72 | 25.37 | 26.49 |
| gu | 6.75 | 6.77 | 6.78 | 10.42 | **7.97** |
| hi | **19.10** | 18.75 | 16.01 | 16.95 | 17.76 |
| kn | 3.93 | 4.46 | 3.69 | **4.89** | 4.83 |
| ml | 6.26 | 6.34 | 6.64 | 6.19 | **6.67** |
| mr | 17.69 | 21.51 | 19.15 | **25.00** | 22.87 |
| or | 21.80 | 20.08 | 22.23 | 22.66 | **24.48** |
| pa | **11.90** | 11.46 | 11.36 | 11.33 | 11.48 |
| ta | 6.88 | 6.71 | 6.60 | 6.99 | **7.54** |
| te | 3.59 | 3.34 | 3.64 | 3.44 | **4.24** |
| Average | 13.14 | 12.74 | 12.22 | 13.34 | **13.95** |

Table 4.3: BLEU scores for the sharded and probabilistic curriculum for all languages. Best results are highlighted in bold

examples than the initial iterations for both coverage score and noise score, even if the noise score is more robust to it. The probabilistic sampling scheme rectifies this skewed distribution of data by ensuring that longer sequences are present in later iterations as well. This improves the quality of generation as catastrophic forgetting is avoided during training. This is reflected in the results, with the simple sharded curriculum resulting in a decrease in performance compared to non-curriculum learning for both metrics. With the probabilistic curriculum, both coverage score and noise score show an improvement over non-curriculum learning, with noise score presenting the strongest performing system.

### 4.5.3 LLM-based generation

We used two prompting techniques with GPT-3.5 to explore the potential of using LLMs to generate a trusted data source. We experimented with only English, with the few-shot prompt resulting in a BLEU score of 13.53 and the CoT prompt resulting in a BLEU score of 17.09. The chain-of-thought approach performs better than the few-shot approach, as expected.

| | | Sharded | | Probabilistic | |
|---|---|---|---|---|---|
| Language | Base | Coverage | Noise | Coverage | Noise |
| as | 23.90 | 23.74 | 19.46 | 23.11 | **26.58** |
| bn | 47.15 | 42.76 | 42.29 | 42.05 | **48.36** |
| en | 45.08 | 44.70 | 41.15 | 43.32 | **45.17** |
| gu | 25.88 | 23.04 | 23.87 | **26.19** | 25.90 |
| hi | 36.11 | **36.45** | 33.19 | 34.68 | 35.98 |
| kn | 23.16 | 23.08 | 21.70 | 23.26 | **26.00** |
| ml | 27.69 | 26.51 | 24.98 | 25.20 | **27.89** |
| mr | 36.10 | 38.14 | 34.46 | 36.60 | **37.86** |
| or | 44.77 | 42.00 | 40.95 | 42.05 | **44.93** |
| pa | **28.41** | 24.99 | 26.29 | 25.83 | 26.92 |
| ta | **35.59** | 32.08 | 31.82 | 32.21 | 34.17 |
| te | 23.79 | 23.22 | 22.92 | 23.67 | **25.85** |
| Average | 33.14 | 31.73 | 30.26 | 31.51 | **33.80** |

Table 4.4: chrF++ scores for sharded and probabilistic curriculum for all languages. Best results are highlighted in bold

Breaking down the generation into multiple smaller steps of generating one sentence at a time helps the model generate higher-quality texts. However, compared to the other methods, both result in poor performance. We hypothesize that this is due to the unique style of Wikipedia, which is difficult to grasp using only a few exemplars. Considering the fact that generating content natively in low-resource languages or using translation would both result in even worse performance for the low-resource languages, we did not proceed with this approach for generating the trusted dataset, especially given the high cost. While not the intended direction, this demonstrates that this problem is challenging even with the advent of powerful LLMs.

### 4.5.4 Analyzing Discrepencies

Compared to the baseline method, the proposed method does not result in improvement in performance for 4 languages - English, Hindi, Assamese and Punjabi. This can be attributed to two causes -

- **Sampling invariant of data distribution** - The same sampling function is used for all languages. However, the distribution of coverage scores and sentence length is not consistent across all languages. For instance, the average length of sentences in the test set for English and Hindi is smaller than that of other languages (2.4 and 2.6 vs 3.5). However, the average sentence length per iteration is higher than that of other languages during training. Thus, while the sampling scheme is designed to ensure that samples of all sentence lengths are represented throughout the training process, this results in a deterioration of performance in some languages where this results in disproportionate presence of longer sentences in the latter iterations.
- **Poor quality test set** - Unlike XAlign, the test set of XLAlign is also automatically constructed based on thresholds of coverage and coherence scores, resulting in a noisy dataset. Assamese and Punjabi have the lowest quality test sets as measured by coverage score. This can result in inaccurate estimation of quality by reference based metrics, as the reference themselves are noisy in nature.

## 4.6 Conclusion

In summary, this chapter expands on the problem of fact-to-text generation by focusing on longer texts. We introduce a new dataset for this problem and provide training, validation and test splits based on the quality of the data using coherence - which quantifies the consistency of the texts, and coverage - which quantifies the alignment of the data. The challenges that noisy data entail, such as hallucination, are exacerbated by this shift in focus, and we show that previous methods that work well with shorter texts result in subpar performance here. We subsequently experiment with denoising data using a trusted data source. We train two

language models - one of the out-of-domain noisy dataset and one on the in-domain trusted dataset and quantify the difference in probability of a fact-text pair using the two models. This difference serves as an approximation for the noise in the data. We show that using this measure as the metric for ordering data, along with using a probabilistic sampling scheme to ensure diversity of data throughout the training, results in performance improvements.

*Chapter 5*

# Reducing Hallucinations for Cross-Lingual Fact-to-Long Text Generation

In the previous chapters, we have focused on using curriculum learning to improve the quality of generation for the cross-lingual fact-to-text generation problem. While these methods show promising improvements in performance compared to training without curriculum learning, they are oriented more towards the considered of data than addressing the problem of hallucination. In this chapter, we step away from these methods and work towards devising explicit methods for minimizing hallucination by focusing on different stages of training language models, from the input organization to the evaluation metric.

## 5.1 Overview

A major challenge in using language models and generative models for sensitive use cases is the phenomenon of hallucinations - when models generate factually incorrect content or content not grounded in the provided input. This is particularly critical in scenarios like generating informative content in domains such as education, law, medicine, finance etc., where accurate and consistent generation is paramount. Thus, the investigation of methods for mitigating hallucinations represents a critical problem.

In this chapter, we investigate methods for reducing hallucinations for the cross-lingual fact-to-text generation problem. We have motivated the significance of this problem in the previous chapters. Given the low-resource nature of the languages, the lack of high-quality training resources means that existing solutions perform poorly for it. Additionally, the problem finds uses in several critical applications, further accentuating the need for reliable and trustworthy solutions. We also address the gap in evaluating such a problem - existing evaluation metrics are source-independent or only account for the monolingual setting source-dependent. Source-independent metrics fail to properly evaluate problems with divergent references, where the

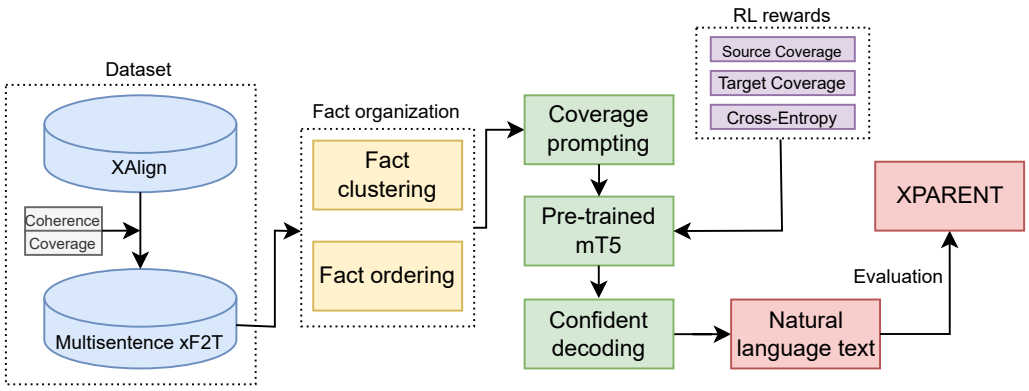



Figure 5.1: The complete pipeline proposed for XFLT with explicit means for hallucination reduction.

source and reference do not align. We perform our studies on the previously proposed XLAlign dataset. In summary, we make the following contributions with this work -

- We propose a flexible system for the generation of long texts from facts with explicit means for controlling hallucinations at various stages such as including using coverage prompts in the inputs and performing confident decoding, which takes into account the input
- We extend the PARENT metric, a source-dependent metric shown to have a greater correlation with human evaluation, to the cross-lingual setting, referred to as the xPARENT metric.

## 5.2 Methodology

Figure 5.1 illustrates the overall proposed pipeline encompassing the different steps. The XLAlign dataset contains a set of facts from knowledge graphs as input and corresponding text from Wikipedia as the reference text. Our approach involves breaking the problem down into two steps - 1) grouping similar facts and 2) generating sentences for each group of facts. This approach is based on the intuition that generating single sentences is easier than generating multiple sentences in one go. The first step allows us to collect facts that frequently occur together, such as cause of death and date of death. The second step allows us to generate the text one sentence at a time, avoiding problems with the generation of longer texts, such as propagation of errors, loss of information, repetition of facts, etc. In the subsequent sections, we describe each step in detail.

### 5.2.1 Input Organization

As stated above, the first step in the pipeline involves collecting similar facts into distinct logical groups. Here, a group represents facts used to construct a single sentence. This is to leverage the co-occurrence of facts in the same sentence. For instance, a common formulation for representing date of birth and date of death for Wikipedia articles is "<date-of-birth>-<date-of-death>". For this, two methods were experimented with -

- MURIL classifier - For this, a classifier was trained using a pretrained MURIL model [30] to predict the number of fact groupings given the complete set of facts. The ground truth number of groups was determined using the number of sentences in the reference text
- mT5 text-to-text generator - For this, a mT5 model [77] was trained to generate the fact groups as a text-to-text generation problem. Note that this performs both the task of grouping the facts and ordering the groups.

### 5.2.2 Input Coverage Prompts

Owing to its synthetic construction, the XLAlign dataset contains significant noise. To account for this, we re-use the coverage score introduced in previous chapters to inform the model of the quality of input during training. For this, the coverage score, which is the confidence score of a binary classifier trained on manually annotated data, is used to assign one of three labels to each input - high, medium or low coverage based on predetermined thresholds. During training, this label is provided along with the input to the model. Only the high label is provided during inference, guiding the model to generate text with a high degree of alignment with the input data.

### 5.2.3 Training with Policy Gradient Optimization

Reinforcement learning allows us to guide models to align closely with desired metrics. We leverage this to improve the generation quality on the following scales - 1) alignment with reference text and 2) alignment with input facts. Due to cross-lingual data, the former is syntactic in nature, while the latter is semantic in nature. Optimization is performed using policy gradient optimization to maximize the expected reward. This is accomplished by sampling as follows for a sequence $s$ and model parameters $\theta$

$$\Delta_\theta J(\theta) = E[R.\Delta_\theta log(P(y_k|x;\theta)]$$

Here, $R$ is the reward(s), $y_k$ is sampled from the distribution of the model outputs at each decoding step, $x$ is the input. A combination of the base model and policy gradient for the different rewards is used to calculate the overall loss.

We experiment with the following rewards -

**1. Target Similarity.** BLEU score is used to reward the model for syntactic alignment with the reference texts. Concretely, given the generated output $\hat{y}$ and reference text $y$, the reward is given as $R_t = \lambda_T.BLEU(\hat{y}, y)$. Here, $\lambda_T$ is a hyperparameter to control the importance of the reward

**2. Source Similarity.** To capture the alignment of the generated text with the input set of facts, we first begin with the n-gram semantic score introduced in Chapter 1. Given the input set of facts, it captures the probability that an n-gram in the reference text is correct. Consider an n-gram $g$ from the set of n-grams $G$ for the target sequence $s$ and the set of lexical tokens in the input facts $F = \{v_1, v_2 \dots v_k\}$. For a lexical token $t$, let $\hat{t}$ represent its embedding. Then, we define the token similarity for a token $g_j$ from $g$ as

$$f(g_j, F) = \max_{v_i \in F} s(\hat{g_j}, \hat{v_i})$$

Then, the similarity of the n-gram $g$ is given as the geometric average of token similarity of each of each of its tokens. We call this the entailment probability.

$$w(g, F) = \left( \prod_{g_i \in g} f(g_i, F) \right)^{\frac{1}{|g|}}$$

The entailment score of $s$ for all n-grams of order n is given as the average of the similarity scores over all its n-grams.

$$ES_n(s, F) = \frac{\sum_{g \in G} w(g, F)}{|G|}$$

Finally, the entailment score of $s$ with respect to $F$ is computed as the geometric mean of $ES_n(s, F)$ across n-grams of order 1-3. Similar to source similarity, the final reward is weighted by a hyperparameter to control its importance

$$R_S = \lambda_S.ES(s, F)$$

### 5.2.4 Confident Decoding

Generally, decoding of text during inference is based on the language model probabilities of the computed logits. This, however, disregards the input entirely, which is undesirable when generating text with diverging references which may not necessarily align with the input. To address this, we employ confident decoding during inference as proposed by Tian et al. [65]. For the top $k$ tokens based on the language modelling probabilities, we compute the entailment probability as described above. A combination of these two measures is used during beam search to promote the generation of text entailed by the input set of facts.

### 5.2.5 Evaluation for Noisy References

Evaluation of models trained on divergent references is challenging and requires source-dependent metrics. We extend the PARENT score metric for the cross-lingual problem, thereby creating the xPARENT score. It is defined as the mean of the entailed precision and entailed recall, whose formulation is described next for each instance $(F^i, R^i, G^i)$ of input facts, reference text, and generated text.

**1. Entailed Precision** is based on the fraction of n-grams in $G_i$ present in $G_i$. First, we define the entailed precision $E_p^n$ for n-grams of order n. This is defined as

$$E_p^n = \frac{\sum_{g \in G_n^i} [Pr(g \in R_n^i) + Pr(g \notin R_n^i) w(g)] \#_{G_n^i}(g)}{\sum_{g \in G_n^i} \#_{G^n}^i(g)}$$

In other words, it assigns a score of 1 to n-grams that appear in the reference. If not, the score is weighted by the entailment probability with respect to the input facts. Both the numerator and the denominator are weighted by the count of the n-gram. Finally, to compute the entailed precision of an instance, the geometric mean of $E_n^p$ is computed for n-grams of the order 1-4.

**2. Entailed Recall** is computed against both the reference text and the set of input facts to ensure proper structure and reward generations mentioning more information from the input, respectively.

Entailed Recall against reference $E_r(R^i)$ is computed as follows

$$E_r^p(R^i) = \frac{\sum_{g \in G^i} \#_{G_n^i, R_n^i}(g) w(g)}{sum_{g \in R^i} \#_{R_n^i}(g)}$$

Here, the n-grams are weighted by their entailment probability to penalize divergent references that do not align with the source. Entailed Recall against source $E_s(F^i)$ is computed as follows

$$E_s(F^i) = \frac{\sum_{g \in F^i} (I(w(g) > T) . \#_{F^i}(g)}{\sum_{g \in F^i} \#_{F^i}(g)}$$

Here, $I$ is an indicator function, and $T$ is a threshold determined based on manual inspection. Finally, entailed recall for an instance is computed as the weighted geometric average of $E_s$ and $E_r$ with the hyperparameter $\lambda$, allowing varying the relative importance of the two measures.

$$ER = E_s^{\lambda_R} . E_r^{1-\lambda_R}$$

**xPARENT** for an instance is computed as the harmonic mean of its entailed precision and entailed recall. For a corpus, the xPARENT score is computed as the mean of the xPARENT score of the instances.

## 5.3 Experimental Setup

### 5.3.1 Baselines

We compare the performance of our pipeline against various baselines, with several ablations to determine the relative importance of different steps.

- Vanilla - This represents a model trained on the XLAlign dataset end-to-end
- Fact Organizer with single sentence generation (FS) - This represents identifying distinct fact groups and using a model trained for single-sentence generation to generate text for each fact group.
- FS + Coverage Prompt (CP) - This involves providing the coverage prompts in the input to the single sentence of the fact organizer pipeline
- FS + CP + Policy Gradient Optimization (GO) - Here, the policy gradients for the described rewards and the model loss function are used during optimization.
- FS + CP + GO + Confident Decoding (CD) - Confident decoding and the previous steps are used at inference time.

### 5.3.2 Configurations

For all experiments, mT5-base models were trained using V100 GPUs. A training rate of 0.001 was used for all experiments except the policy gradient optimization-based experiments, where a learning rate of 0.00002 was used. The models were trained for a maximum of 30 epochs, with the best model selected based on the validation loss.

## 5.4 Results and Analysis

We use BLEU and chrF++ to evaluate the performance of the different approaches. Additionally, we also xPARENT score to evaluate the performance. The averaged results of the ablations can be seen in Table 5.2, while the best-performing model's performance compared to the basic approach across different languages for both single-sentence instances and multi-sentence instances can be seen in Table 5.4.

### 5.4.1 Fact Organization

The results of the different approaches used to organize facts is shown in Table 5.1. Additionally, the number of predicted groups against the ground truth number of groups is illustrated via a heatmap for both approaches in Figure 5.2. The mT5 text-to-text generator-based approach

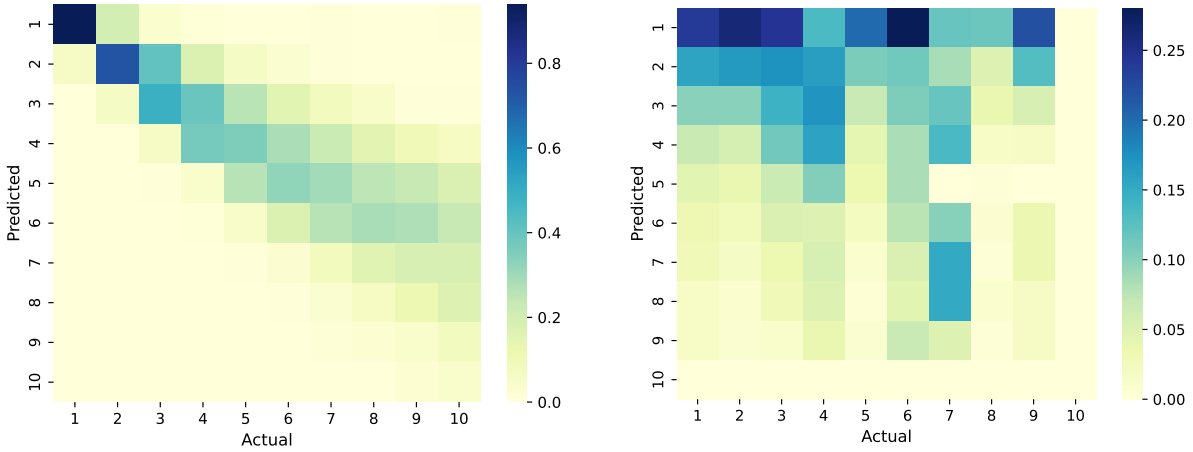



Figure 5.2: Heatmap comparing actual versus predicted number of logical groups using the proposed fact organizer (left) and Muril-base classifier(right).

| | F1 | MSE |
|---|---|---|
| MURIL | 0.25 | 4.67 |
| mT5 | 0.60 | 1.28 |

Table 5.1: Results of different fact-organization methods. F1 is micro-average F1

is superior to the Muril classifier. This organizer determines the facts in each group and the order of the groups. It is important to quantify the model's performance on both these tasks. We quantify the former by modelling it as a minimum weight matching problem to identify 1:1 matches between the identified groups and ground truth groups, representing a bipartite graph. Then, the correctness of the assignment is computed as the percentage of correctly grouped facts. This reveals that over 81% of the facts are correctly clustered for instances with more than or equal to 2 sentences in the ground truth. For the latter, we compute the Kendall rank correlation coefficient, which for instances with more than or equal to 2 sentences is 0.70. Both numbers are high and indicate that the model is successful at both identifying the groups and ordering the groups.

### 5.4.2 Text Generation

From the ablations, it is clear that the proposed pipeline significantly improves the quality of generation. There is a sizeable improvement in performance when switching from vanilla training to training with fact organization and single sentence generation. Each progressive step introduced results in better performance, with the final model that makes use of fact or-

| | All | | | >1 sentence | | |
|---|---|---|---|---|---|---|
| | chrF++ | xPARENT | BLEU | chrF++ | xPARENT | BLEU |
| Vanilla | 38.97 | 49.35 | 18.99 | 36.83 | 46.40 | 16.96 |
| FO | 44.14 | 52.68 | 20.40 | 43.37 | 52.63 | 18.23 |
| FO+CP | 48.82 | 55.27 | 22.06 | 48.12 | 55.07 | 18.44 |
| FO+CP+GO | 49.53 | 55.33 | 22.66 | 48.72 | 54.97 | 18.76 |
| FO+CP+GO+CD | 50.14 | 56.56 | 23.01 | 49.32 | 56.13 | 19.04 |

Table 5.2: Results averaged across all languages with various ablations; FS- Fact Organizer followed by single sentence generation, CP - Coverage prompts in input, GO - Policy based gradient optimization with rewards, CD - Confident decoding

ganization, coverage prompts, policy gradient optimization and confident decoding, resulting in the best model. Each step results in improvement across almost all metrics. However, we note that the xPARENT score decreases with the introduction of policy gradient optimization, which bears further investigation. The breakdown of performance across the languages reveals interesting insights about the nature of the problem and the metrics chosen. The significant difference in performance between the two methods makes it clear that generating longer contexts requires special methods. Further, while BLEU observes the biggest improvement for the complete dataset, xPARENT observes the biggest improvement for the setting where only multi-sentence instances are evaluated. This indicates that xPARENT is better suited for this context and presents a more accurate summary of the performance of the methods, particularly with respect to reducing hallucinations.

### 5.4.3 Human Evaluation

Partial human evaluation was conducted to establish the validity of the proposed metric. For this, outputs from the vanilla baseline model (A) were compared to the outputs from the model trained using the proposed pipeline (B). Several outputs were chosen such that BLEU scores and xPARENT scores were in opposition i.e the output A was preferred by A but output B was preferred by xPARENT. The responses were evaluated along 3 axes - a) fidelity, to evaluate the presence of hallucinations, b) recall, to evaluate the coverage of input facts, and c) coherence, to evalute the legibility and construction of the outputs. In a majority of cases, the outputs from the proposed pipeline were preferred.

| | Punjabi | | | English | | | Hindi | | | Marathi | | | Telugu | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Measure | F | R | C | F | R | C | F | R | C | F | R | C | F | R | C |
| Proposed | 53 | 65 | 64 | 42 | 33 | 31 | 46 | 45 | 52 | 42 | 55 | 59 | 21 | 54 | 68 |
| Vanilla | 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 5.3: Percentage of time the response from each model was preferred across fidelity (F), Recall (R), and Coherence (C) by human evaluators

| Lang | Vanilla | | | | | | FS+CP+GO+CD | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | All | | | >1 sentence | | | All | | | >1 sentence | | |
| | chrF++ | xPARENT | BLEU | chrF++ | xPARENT | BLEU | chrF++ | xPARENT | BLEU | chrF++ | xPARENT | BLEU |
| as | 23.9 | 31.34 | 6.69 | 23.84 | 29.81 | 5.04 | 43.36 | 40.31 | 8.12 | 43.54 | 41.36 | 7.23 |
| bn | 47.15 | 48.72 | 26.04 | 46.65 | 49.31 | 16.23 | 58.77 | 62.99 | 25.22 | 58.71 | 62.50 | 22.65 |
| en | 45.08 | 67.29 | 27.06 | 42 | 62.92 | 19.58 | 53.92 | 68.67 | 30.65 | 52.77 | 67.57 | 25.70 |
| gu | 25.88 | 33.58 | 6.75 | 25.45 | 32.47 | 6.11 | 40.64 | 43.82 | 13.60 | 39.95 | 45.50 | 10.58 |
| hi | 36.11 | 50.68 | 19.1 | 33.64 | 46.23 | 16.50 | 48.26 | 59.00 | 25.95 | 47.21 | 58.46 | 20.97 |
| kn | 23.16 | 24.96 | 3.93 | 22.44 | 25.29 | 4.20 | 36.22 | 39.05 | 7.55 | 36.14 | 40.65 | 6.43 |
| ml | 27.69 | 28.95 | 6.26 | 27.24 | 27.02 | 6.72 | 41.39 | 37.13 | 10.51 | 41.28 | 39.08 | 9.11 |
| mr | 36.1 | 48.38 | 17.69 | 27.8 | 38.27 | 12.06 | 51.13 | 56.45 | 29.86 | 45.95 | 51.95 | 18.50 |
| or | 44.77 | 48.95 | 21.8 | 44.73 | 48.58 | 18.11 | 60.01 | 50.53 | 26.60 | 60.35 | 52.33 | 26.85 |
| pa | 28.41 | 42.41 | 11.9 | 27.34 | 40.07 | 10.06 | 39.78 | 52.49 | 15.84 | 39.28 | 50.60 | 12.22 |
| ta | 35.59 | 26.62 | 6.88 | 34.16 | 25.05 | 5.85 | 44.94 | 36.69 | 11.91 | 45.14 | 37.93 | 9.12 |
| te | 23.79 | 28.85 | 3.59 | 23.35 | 27.95 | 4.12 | 39.59 | 38.41 | 8.49 | 39.47 | 40.10 | 7.11 |

Table 5.4: Comparison of vanilla training vs the best-performing system for every language

## 5.5 Conclusion

In this chapter, we perform a comprehensive study of generating longer texts for cross-lingual fact-to-text generation, focusing on methods for reducing hallucinations. We describe our pipeline, which empirically results in the best performance. It involves breaking the task down to identifying logical groups of facts and then generating sentences for each group. With the help of methods like coverage prompts to identify the quality of a training instance, policy gradient based optimization to align the model to both the reference and the source, and confident decoding during inference to mitigate the effects of divergent references, we ground the text on the input facts, minimizing hallucinations. Finally, we also propose a novel metric for evaluating the performance of models when dealing with divergent references for the cross-lingual setting.

*Chapter 6*

# Conclusion

In this thesis, we explored ways to deal with noisy data. Our experiments and results consistently show that considered use of data is important in this setting and can result in sizeable improvement in quality of generations. The testing grounds for our studies was the problem of cross-lingual fact-to-text generation of low-resource Indian languages. A critical problem, progress towards devising reliable solutions for it holds promise for a more equitable and fair internet, where people can participate freely, unconstrained by language barriers. Our contributions include new training methods, datasets, and a novel evaluation metric.

In **Chapter 1**, we established the importance of this problem. Given the lack of high-quality resources, we motivated why methods suited for training with noisy data are especially applicable for cross-lingual generation. We highlight the information gap between English and Indian languages despite the vast number of native speakers of these languages. We introduce the idea of using synthetic datasets and generative techniques to bridge this gap whilst also acknowledging the pitfalls of these methods.

**Chapter 2** represented a study of the current status quo. We studied the current literature to understand the progress made towards data-to-text generation and cross-lingual text generation. We observed that datasets for generating informative content in Indian languages leverage high-resource languages like English to generate synthetic datasets whilst also learning about the performance of current systems for this problem. We also investigated how curriculum learning has been used for various tasks and realized its potential to enable careful use of data, subsequently improving performance. This study allowed us to understand the drawbacks and pitfalls of current methods and identify gaps in the literature that need to be addressed.

In **Chapter 3**, we began our journey towards addressing the gaps in literature. We employed a predefined curriculum learning strategy with novel ordering metrics that jointly model both input and reference text for cross-lingual data. We introduced the notion of coverage score and showed it to be an effective metric for ordering data. We also experimented with different schedules, which allowed us to understand the nature of different metrics.

Building on the insights and results from **Chapter 4**, we introduced a more challenging variant of the cross-lingual fact-to-text generation problem by focusing on generating longer texts. For this, we constructed the XLAlign dataset by reusing existing datasets. We partitioned the dataset based on essential metrics like coherence and coverage to create high-quality test and validation sets. We then expanded the methods that proved effective in the previous chapter to become more pliable to this problem statement. Specifically, we utilized an automatic CL strategy with a weighted sampling-based scheduler to ensure diversity of samples during training. We experimented with the notion of denoising data and using a probabilistic sampling scheme, both of which resulted in sizeable performance improvements.

Finally, we concluded our explorations in **Chapter 5** where we focused on generating more grounded texts, mitigating the effects of hallucinations. We devised a pipeline for generating longer texts with several mechanisms in place, such as sequential generation of sentences based on the clustering of facts, coverage prompts in the input, alignment-based rewards, and confident decoding during inference to mitigate hallucinations. We also introduced a better way of evaluating the performance of models in this setting with divergent references.

Overall, this thesis represents a step towards improving the generation quality for low-resource languages by leveraging resources from high-resource languages. It does so by honing in on leveraging automatically created synthetic datasets, which can bridge the gap between high-resource and low-resource languages. However, the importance of considered use of such data is always at the forefront, with methods dedicated to ensuring this being the focus. This goes hand in hand with thinking about mitigating hallucinations, a common problem this domain suffers from.

## 6.1 Future Work

We close this thesis by discussing potential avenues for building on the work presented here.

1. **Generalization to different problems:** The bulk of the methods discussed in this thesis can be generalized to other data-to-text generation tasks and on a larger scope to text generation problems in general. Noisy data is a common problem in various areas such as summarization, machine translation, etc. Thus, the generalization of the methods investigated here to other problems needs further study.

2. **Better human evaluation:** Due to the expensive nature of human evaluation, the evaluations performed in the thesis were based on automatic metrics. However, such metrics have pitfalls and can have a low correlation with human judgement. The validity of the approaches can be further reinforced with comprehensive human evaluations.

3. **Leveraging LLMs:** This work's investigation of LLMs was limited in scope. Given their rising importance and ever-improving performance, future work could seek to involve them in the pipeline. They can be used to generate synthetic datasets on their own.

4. **Expanding to different languages:** This thesis's works were constrained to exploring Indian languages. However, studying them in the context of other low-resource languages from across the globe is equally important. This could potentially necessitate creating new datasets and opening up several new threads for research.

5. **Incorporating non-generative models:** All methods in this thesis are based on the text-to-text generation problem using the T5 model, which is a text-to-text model. A combination of other types of models and techniques could enrich future studies. This can be in the form of techniques like prompt tuning, alternative architectures etc.

## Related publications

- Bhavyajeet Singh*, Kancharla Aditya Hari*, Rahul Mehta, Manish Gupta, Vasudeva Varma **XFLT : Exploring Techniques for Generating Cross Lingual Factually Grounded Long Text** In Proceedings of ECAI 2023
- Kancharla Aditya Hari, Vasudeva Verma. **Curriculum Learning for Cross-Lingual Fact-to-Text Generation With Noisy Data** Under review.
- Bhavyajeet Singh*, Kancharla 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

## Other Publications

- Kancharla Aditya Hari, Bhavyajeet Singh, Anubhav Sharma and Vasudeva Varma. **WebNLG Challenge 2023: Domain Adaptive Machine Translation for Low-Resource Multilingual RDF-to-Text Generation** In Proceedings of the First Workshop on Muiltimodal, Multilingual NLG
- Bhavyajeet Singh, Ankita Maity, Siri Venkata Pavan kumar Kandru, Kancharla 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

- Sagar Joshi, Sumanth Balaji, Kancharla Aditya Hari, Abhijeet Singham and Vasudeva Varma. **Efficacy of Pretrained Architectures for Code Comment Usefulness Prediction** in Proceedings of the Forum for Information Retrieval Evaluation, 2022

- Sumanth Balaji, Sagar Joshi, Bhavyajeet Singh, Kancharla Aditya Hari, Abhijeet Singham and Vasudeva Varma. **COVID-19 vaccine stance classification from tweets** in Proceedings of the Forum for Information Retrieval Evaluation, 2022

# 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 *Companion Proceedings of the Web Conference 2022*, WWW '22, page 171175, New York, NY, USA, 2022. Association for Computing Machinery.

[2] K. Aditya Hari, B. Singh, A. Sharma, and V. Varma. WebNLG challenge 2023: Domain adaptive machine translation for low-resource multilingual RDF-to-text generation (WebNLG 2023). In A. Gatt, C. Gardent, L. Cripwell, A. Belz, C. Borg, A. Erdem, and E. Erdem, editors, *Proceedings of the Workshop on Multimodal, Multilingual Natural Language Generation and Multilingual WebNLG Challenge (MM-NLG 2023)*, pages 93–94, Prague, Czech Republic, Sept. 2023. Association for Computational Linguistics.

[3] T. Aoki, A. Miyazawa, T. Ishigaki, K. Goshima, K. Aoki, I. Kobayashi, H. Takamura, and Y. Miyao. Generating market comments referring to external resources. In E. Krahmer, A. Gatt, and M. Goudbeek, editors, *Proceedings of the 11th International Conference on Natural Language Generation*, pages 135–139, Tilburg University, The Netherlands, Nov. 2018. Association for Computational Linguistics.

[4] R. Barzilay and M. Lapata. Collective content selection for concept-to-text generation. In R. Mooney, C. Brew, L.-F. Chien, and K. Kirchhoff, editors, *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 331–338, Vancouver, British Columbia, Canada, Oct. 2005. Association for Computational Linguistics.

[5] Y. Bengio, J. Louradour, R. Collobert, and J. Weston. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, page 4148, New York, NY, USA, 2009. Association for Computing Machinery.

[6] T. Castro Ferreira, D. Moussallem, E. Krahmer, and S. Wubben. Enriching the WebNLG corpus. In E. Krahmer, A. Gatt, and M. Goudbeek, editors, *Proceedings of the 11th International Conference on Natural Language Generation*, pages 171–176, Tilburg University, The Netherlands, Nov. 2018. Association for Computational Linguistics.

[7] E. Chang, H.-S. Yeh, and V. Demberg. Does the order of training samples matter? improving neural data-to-text generation with curriculum learning. In P. Merlo, J. Tiedemann, and R. Tsarfaty, editors, *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 727–733, Online, Apr. 2021. Association for Computational Linguistics.

[8] M. Chen, X. Lu, T. Xu, Y. Li, Z. Jingbo, D. Dou, and H. Xiong. Towards table-to-text generation with pretrained language model: A table structure understanding and text deliberating approach. In Y. Goldberg, Z. Kozareva, and Y. Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8199–8210, Abu Dhabi, United Arab Emirates, Dec. 2022. Association for Computational Linguistics.

[9] H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, Y. Li, X. Wang, M. Dehghani, S. Brahma, A. Webson, S. S. Gu, Z. Dai, M. Suzgun, X. Chen, A. Chowdhery, A. Castro-Ros, M. Pellat, K. Robinson, D. Valter, S. Narang, G. Mishra, A. Yu, V. Zhao, Y. Huang, A. Dai, H. Yu, S. Petrov, E. H. Chi, J. Dean, J. Devlin, A. Roberts, D. Zhou, Q. V. Le, and J. Wei. Scaling instruction-finetuned language models, 2022.

[10] L. Cripwell, A. Belz, C. Gardent, A. Gatt, C. Borg, M. Borg, J. Judge, M. Lorandi, A. Nikiforovskaya, and W. Soto Martinez. The 2023 WebNLG shared task on low resource languages. overview and evaluation results (WebNLG 2023). In A. Gatt, C. Gardent, L. Cripwell, A. Belz, C. Borg, A. Erdem, and E. Erdem, editors, *Proceedings of the Workshop on Multimodal, Multilingual Natural Language Generation and Multilingual WebNLG Challenge (MM-NLG 2023)*, pages 55–66, Prague, Czech Republic, Sept. 2023. Association for Computational Linguistics.

[11] 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.

[12] O. Dušek, D. M. Howcroft, and V. Rieser. Semantic noise matters for neural natural language generation. In K. van Deemter, C. Lin, and H. Takamura, editors, *Proceedings of the 12th International Conference on Natural Language Generation*, pages 421–426, Tokyo, Japan, Oct.–Nov. 2019. Association for Computational Linguistics.

[13] O. Dušek and F. Jurčíček. Sequence-to-sequence generation for spoken dialogue via deep syntax trees and strings. In K. Erk and N. A. Smith, editors, *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 45–51, Berlin, Germany, Aug. 2016. Association for Computational Linguistics.

[14] 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).

[15] Y. Fan, F. Tian, T. Qin, X.-Y. Li, and T.-Y. Liu. Learning to teach. In *International Conference on Learning Representations*, 2018.

[16] 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.

[17] 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.

[18] Z. Fu, B. Shi, W. Lam, L. Bing, and Z. Liu. Partially-aligned data-to-text generation with distant supervision. In B. Webber, T. Cohn, Y. He, and Y. Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9183–9193, Online, Nov. 2020. Association for Computational Linguistics.

[19] 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.

[20] A. Gatt and E. Krahmer. Survey of the state of the art in natural language generation: core tasks, applications and evaluation. *J. Artif. Int. Res.*, 61(1):65170, jan 2018.

[21] X. Geng, Y. Zhang, J. Li, S. Huang, H. Yang, S. Tao, Y. Chen, N. Xie, and J. Chen. Denoising pre-training for machine translation quality estimation with curriculum learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(11):12827–12835, Jun. 2023.

[22] C. Gong, D. Tao, S. J. Maybank, W. Liu, G. Kang, and J. Yang. Multi-modal curriculum learning for semi-supervised image classification. *IEEE Transactions on Image Processing*, 25(7):3249–3260, 2016.

[23] A. Graves, M. G. Bellemare, J. Menick, R. Munos, and K. Kavukcuoglu. Automated curriculum learning for neural networks. In D. Precup and Y. W. Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1311–1320. PMLR, 06–11 Aug 2017.

[24] G. Hacohen and D. Weinshall. On the power of curriculum learning in training deep networks. In *International Conference on Machine Learning*, 2019.

[25] E. Hirsch and A. Tal. Clid: Controlled-length image descriptions with limited data. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 5531–5541, January 2024.

[26] W. Hogan. An overview of distant supervision for relation extraction with a focus on denoising and pre-training methods, 2022.

[27] L. Jiang, D. Meng, T. Mitamura, and A. G. Hauptmann. Easy samples first: Self-paced reranking for zero-example multimedia search. In *Proceedings of the 22nd ACM International Conference on Multimedia*, MM '14, page 547556, New York, NY, USA, 2014. Association for Computing Machinery.

[28] 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 D. Scott, N. Bel, and C. Zong, editors, *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2398–2409, Barcelona, Spain (Online), Dec. 2020. International Committee on Computational Linguistics.

[29] M. Kazakov, J. Preobrazhenskaya, I. Bulychev, and A. Shain. WebNLG-interno: Utilizing FRED-t5 to address the RDF-to-text problem (WebNLG 2023). In A. Gatt, C. Gardent, L. Cripwell, A. Belz, C. Borg, A. Erdem, and E. Erdem, editors, *Proceedings of the Workshop on Multimodal, Multilingual Natural Language Generation and Multilingual WebNLG Challenge (MM-NLG 2023)*, pages 67–72, Prague, Czech Republic, Sept. 2023. Association for Computational Linguistics.

[30] S. Khanuja, D. Bansal, S. Mehtani, S. Khosla, A. Dey, B. Gopalan, D. K. Margam, P. Aggarwal, R. T. Nagipogu, S. Dave, et al. Muril: Multilingual representations for indian languages. *arXiv preprint arXiv:2103.10730*, 2021.

[31] T. Kocmi and O. Bojar. Curriculum learning and minibatch bucketing in neural machine translation. In R. Mitkov and G. Angelova, editors, *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, pages 379–386, Varna, Bulgaria, Sept. 2017. INCOMA Ltd.

[32] M. Kumar, B. Packer, and D. Koller. Self-paced learning for latent variable models. In J. Lafferty, C. Williams, J. Shawe-Taylor, R. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems*, volume 23. Curran Associates, Inc., 2010.

[33] N. Kumar, S. Obaid Ul Islam, and O. Dusek. Better translation + split and generate for multilingual RDF-to-text (WebNLG 2023). In A. Gatt, C. Gardent, L. Cripwell, A. Belz, C. Borg, A. Erdem, and E. Erdem, editors, *Proceedings of the Workshop on Multimodal, Multilingual Natural Language Generation and Multilingual WebNLG Challenge (MM-NLG 2023)*, pages 73–79, Prague, Czech Republic, Sept. 2023. Association for Computational Linguistics.

[34] 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.

[35] Y. J. Lee and K. Grauman. Learning the easy things first: Self-paced visual category discovery. In *CVPR 2011*, pages 1721–1728, 2011.

[36] C. Li, F. Wei, J. Yan, X. Zhang, Q. Liu, and H. Zha. A self-paced regularization framework for multilabel learning. *IEEE Transactions on Neural Networks and Learning Systems*, 29(6):2660–2666, 2018.

[37] Y. Lin, T. Ruan, J. Liu, and H. Wang. A survey on neural data-to-text generation. *IEEE Transactions on Knowledge and Data Engineering*, 36(4):1431–1449, 2024.

[38] Y. Lin, S. Shen, Z. Liu, H. Luan, and M. Sun. Neural relation extraction with selective attention over instances. In K. Erk and N. A. Smith, editors, *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2124–2133, Berlin, Germany, Aug. 2016. Association for Computational Linguistics.

[39] X. Liu, H. Lai, D. F. Wong, and L. S. Chao. Norm-based curriculum learning for neural machine translation. In D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 427–436, Online, July 2020. Association for Computational Linguistics.

[40] M. Lorandi and A. Belz. Data-to-text generation for severely under-resourced languages with GPT-3.5: A bit of help needed from Google Translate (WebNLG 2023). In A. Gatt, C. Gardent, L. Cripwell, A. Belz, C. Borg, A. Erdem, and E. Erdem, editors, *Proceedings of the Workshop on Multimodal, Multilingual Natural Language Generation and Multilingual WebNLG Challenge (MM-NLG 2023)*, pages 80–86, Prague, Czech Republic, Sept. 2023. Association for Computational Linguistics.

[41] B. Luo, Y. Feng, Z. Wang, Z. Zhu, S. Huang, R. Yan, and D. Zhao. Learning with noise: Enhance distantly supervised relation extraction with dynamic transition matrix. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, 2017.

[42] F. Mairesse, M. Gašić, F. Jurčíček, S. Keizer, B. Thomson, K. Yu, and S. Young. Phrase-based statistical language generation using graphical models and active learning. In J. Hajič, S. Carberry, S. Clark, and J. Nivre, editors, *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1552–1561, Uppsala, Sweden, July 2010. Association for Computational Linguistics.

[43] S. Mille, E. Uí Dhonnchadha, S. Dasiopoulou, L. Cassidy, B. Davis, and A. Belz. DCU/TCD-FORGe at WebNLG'23: Irish rules! (WegNLG 2023). In A. Gatt, C. Gardent, L. Cripwell, A. Belz, C. Borg, A. Erdem, and E. Erdem, editors, *Proceedings of the Workshop on Multimodal, Multilingual Natural Language Generation and Multilingual WebNLG Challenge (MM-NLG 2023)*, pages 87–92, Prague, Czech Republic, Sept. 2023. Association for Computational Linguistics.

[44] M. Mintz, S. Bills, R. Snow, and D. Jurafsky. Distant supervision for relation extraction without labeled data. In K.-Y. Su, J. Su, J. Wiebe, and H. Li, editors, *Proceedings of the Joint Conference*

*of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1003–1011, Suntec, Singapore, Aug. 2009. Association for Computational Linguistics.

[45] R. C. Moore and W. Lewis. Intelligent selection of language model training data. In J. Hajič, S. Carberry, S. Clark, and J. Nivre, editors, *Proceedings of the ACL 2010 Conference Short Papers*, pages 220–224, Uppsala, Sweden, July 2010. Association for Computational Linguistics.

[46] 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.

[47] S. Murakami, A. Watanabe, A. Miyazawa, K. Goshima, T. Yanase, H. Takamura, and Y. Miyao. Learning to generate market comments from stock prices. In R. Barzilay and M.-Y. Kan, editors, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1374–1384, Vancouver, Canada, July 2017. Association for Computational Linguistics.

[48] L. Nan, D. Radev, R. Zhang, A. Rau, A. Sivaprasad, C. Hsieh, X. Tang, A. Vyas, N. Verma, P. Krishna, Y. Liu, N. Irwanto, J. Pan, F. Rahman, A. Zaidi, M. Mutuma, Y. Tarabar, A. Gupta, T. Yu, Y. C. Tan, X. V. Lin, C. Xiong, R. Socher, and N. F. Rajani. DART: Open-domain structured data record to text generation. In K. Toutanova, A. Rumshisky, L. Zettlemoyer, D. Hakkani-Tur, I. Beltagy, S. Bethard, R. Cotterell, T. Chakraborty, and Y. Zhou, editors, *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 432–447, Online, June 2021. Association for Computational Linguistics.

[49] 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.

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

[51] A. P. Parikh, X. Wang, S. Gehrmann, M. Faruqui, B. Dhingra, D. Yang, and D. Das. Totto: A controlled table-to-text generation dataset. *arXiv preprint arXiv:2004.14373*, 2020.

[52] S. Pauws, A. Gatt, E. Krahmer, and E. Reiter. *Making Effective Use of Healthcare Data Using Data-to-Text Technology*, pages 119–145. Springer International Publishing, Cham, 2019.

[53] E. A. Platanios, O. Stretcu, G. Neubig, B. Poczos, and T. Mitchell. Competence-based curriculum learning for neural machine translation. In J. Burstein, C. Doran, and T. Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1162–1172, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[54] F. Portet, E. Reiter, A. Gatt, J. Hunter, S. Sripada, Y. Freer, and C. Sykes. Automatic generation of textual summaries from neonatal intensive care data. *Artificial Intelligence*, 173(7):789–816, 2009.

[55] 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.

[56] E. Reiter and R. Dale. Building applied natural language generation systems. *Natural Language Engineering*, 3(1):5787, 1997.

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

[58] L. F. R. Ribeiro, Y. Zhang, C. Gardent, and I. Gurevych. Modeling global and local node contexts for text generation from knowledge graphs. *Transactions of the Association for Computational Linguistics*, 8:589–604, 2020.

[59] S. Riedel, L. Yao, and A. McCallum. Modeling relations and their mentions without labeled text. In J. L. Balcázar, F. Bonchi, A. Gionis, and M. Sebag, editors, *Machine Learning and Knowledge Discovery in Databases*, pages 148–163, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

[60] M. Roberti, G. Bonetta, R. Cancelliere, and P. Gallinari. Copy mechanism and tailored training for character-based data-to-text generation. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2019, Würzburg, Germany, September 1620, 2019, Proceedings, Part II*, page 648664, Berlin, Heidelberg, 2019. Springer-Verlag.

[61] S. Sagare, T. Abhishek, B. Singh, A. Sharma, M. Gupta, and V. Varma. Xf2t: Cross-lingual fact-to-text generation for low-resource languages. pages 15–27, 01 2023.

[62] B. Singh, A. Hari, R. Mehta, T. Abhishek, M. Gupta, and V. Varma. *XFLT: Exploring Techniques for Generating Cross Lingual Factually Grounded Long Text.* 09 2023.

[63] V. I. Spitkovsky, H. Alshawi, and D. Jurafsky. From baby steps to leapfrog: How "less is more" in unsupervised dependency parsing. In R. Kaplan, J. Burstein, M. Harper, and G. Penn, editors, *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 751–759, Los Angeles, California, June 2010. Association for Computational Linguistics.

[64] K. Tang, V. Ramanathan, L. Fei-fei, and D. Koller. Shifting weights: Adapting object detectors from image to video. In F. Pereira, C. Burges, L. Bottou, and K. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.

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

[66] P. Törnberg. Chatgpt-4 outperforms experts and crowd workers in annotating political twitter messages with zero-shot learning, 2023.

[67] M. van der Wees, A. Bisazza, and C. Monz. Dynamic data selection for neural machine translation. In M. Palmer, R. Hwa, and S. Riedel, editors, *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1400–1410, Copenhagen, Denmark, Sept. 2017. Association for Computational Linguistics.

[68] W. Wang, T. Watanabe, M. Hughes, T. Nakagawa, and C. Chelba. Denoising neural machine translation training with trusted data and online data selection. In O. Bojar, R. Chatterjee, C. Federmann, M. Fishel, Y. Graham, B. Haddow, M. Huck, A. J. Yepes, P. Koehn, C. Monz, M. Negri, A. Névéol, M. Neves, M. Post, L. Specia, M. Turchi, and K. Verspoor, editors, *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 133–143, Brussels, Belgium, Oct. 2018. Association for Computational Linguistics.

[69] X. Wang, Y. Chen, and W. Zhu. A survey on curriculum learning, 2021.

[70] Z. Wang, M. Collins, N. Vedula, S. Filice, S. Malmasi, and O. Rokhlenko. Faithful low-resource data-to-text generation through cycle training. In A. Rogers, J. Boyd-Graber, and N. Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2847–2867, Toronto, Canada, July 2023. Association for Computational Linguistics.

[71] J. Wei, C. Yang, X. Song, Y. Lu, N. Hu, J. Huang, D. Tran, D. Peng, R. Liu, D. Huang, C. Du, and Q. V. Le. Long-form factuality in large language models, 2024.

[72] Y. Wei, X. Liang, Y. Chen, X. Shen, M.-M. Cheng, J. Feng, Y. Zhao, and S. Yan. Stc: A simple to complex framework for weakly-supervised semantic segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(11):23142320, nov 2017.

[73] D. Weinshall, G. Cohen, and D. Amir. Curriculum learning by transfer learning: Theory and experiments with deep networks, 2018.

[74] S. Wiseman, S. Shieber, and A. Rush. Challenges in data-to-document generation. In M. Palmer, R. Hwa, and S. Riedel, editors, *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2253–2263, Copenhagen, Denmark, Sept. 2017. Association for Computational Linguistics.

[75] B. Xu, L. Zhang, Z. Mao, Q. Wang, H. Xie, and Y. Zhang. Curriculum learning for natural language understanding. In D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, editors, *Proceedings of the*

*58th Annual Meeting of the Association for Computational Linguistics*, pages 6095–6104, Online, July 2020. Association for Computational Linguistics.

[76] C. Xu, D. Tao, and C. Xu. Multi-view self-paced learning for clustering. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, page 39743980. AAAI Press, 2015.

[77] 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 K. Toutanova, A. Rumshisky, L. Zettlemoyer, D. Hakkani-Tur, I. Beltagy, S. Bethard, R. Cotterell, T. Chakraborty, and Y. Zhou, editors, *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online, June 2021. Association for Computational Linguistics.

[78] W. Zaremba and I. Sutskever. Learning to execute, 2015.

[79] D. Zeng, K. Liu, Y. Chen, and J. Zhao. Distant supervision for relation extraction via piecewise convolutional neural networks. In L. Màrquez, C. Callison-Burch, and J. Su, editors, *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1753–1762, Lisbon, Portugal, Sept. 2015. Association for Computational Linguistics.

[80] X. Zhang, G. Kumar, H. Khayrallah, K. Murray, J. Gwinnup, M. J. Martindale, P. McNamee, K. Duh, and M. Carpuat. An empirical exploration of curriculum learning for neural machine translation, 2018.

[81] Y. Zhou, B. Yang, D. F. Wong, Y. Wan, and L. S. Chao. Uncertainty-aware curriculum learning for neural machine translation. In D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6934–6944, Online, July 2020. Association for Computational Linguistics.

Going for a walk in the park