MuRST: Multilingual Recursive Summarization Trees

Stanford CS224N Custom Project

Tarini Mutreja
Department of Computer Science
Stanford University
tarini@stanford.edu

Saron Samuel
Department of Computer Science
Stanford University
sdsam@stanford.edu

Humishka Zope
Department of Computer Science
Stanford University
zope@stanford.edu

Abstract

The vast majority of online information is available in English, leading to suboptimal performance in multilingual question answering (QA) systems, which are often trained and optimized primarily on English data. Consequently, these systems struggle to perform effectively in other languages, resulting in less accurate and efficient responses for non-English users. In this paper, we aim to improve multilingual QA using recursive summarization. To address this challenge, we employed recursive summarization trees for QA on short-form multilingual documents from the XQuAD dataset and long-form multilingual documents from the TyDi QA dataset. Additionally, we experimented with a new tree clustering algorithm involving hard clustering, in contrast to the soft clustering used in RAPTOR [13]. We evaluated the models based on F1 scores and BLEU scores between the ground truth and predicted answers. Our experiments demonstrate that recursive summarization and hard clustering can enhance multilingual QA models, improving their ability to overcome language barriers and democratize access to information. Despite varying degrees of improvement across different languages, our findings highlight areas for further research and optimization, including the exploration of language-specific clustering algorithms and approaches to tree building.

1 Key Information to include

- Mentor: Moussa Doumbouya
- Contributions: Humishka handled the implementation and experimentation of all baselines and vanilla RAPTOR, and all XQuAD experiments. Saron handled all data preprocessing, ran experiments for all MuRST models, and wrote the evaluation code. Tarini handled the implementation and experimentation for the hard clustering algorithm. All authors contributed to the paper equally.

2 Introduction

Effective multilingual question-answering (QA) models are essential in today's interconnected world, where a significant portion of the global population does not speak English. These models have the potential to democratize access to information by bridging language barriers and providing inclusive communication. Multilingual QA systems enable individuals from diverse linguistic backgrounds to access critical information in their native languages, which is crucial for sectors such as healthcare, education, and legal services, among countless others.

Despite the importance of multilingual QA systems, current approaches face several significant challenges. One major issue is the uneven performance across languages, largely due to the scarcity of high-quality training data for many low-resource languages [8]. Models often perform well on high-resource languages but poorly on others, exacerbating the digital divide. Machine translation-based methods, while leveraging robust English QA systems, frequently suffer from translation errors that degrade response accuracy [2]. Furthermore, cross-lingual transfer learning and multilingual pre-trained models, although promising, struggle to capture the cultural and linguistic nuances unique to each language, resulting in potential misinterpretations and inaccuracies [14].

The evaluation of multilingual QA systems presents another challenge, as existing benchmarks might not fully capture the complexity and diversity of real-world language use [2]. These limitations highlight the need for more comprehensive datasets, improved translation techniques, and more sophisticated models capable of understanding and processing a wide range of languages and their specific contexts effectively.

Recursive summarization trees have been shown to improve on QA in English [13]. The goal of this project is to investigate whether recursive summarization trees, combined with multilingual models and appropriate clustering techniques, can improve the performance of multilingual QA systems. This research aims to address the current challenges in multilingual QA and propose solutions that enhance the robustness and accuracy of these systems across diverse languages.

In this paper, we apply recursive summarization trees to multilingual QA. We present the Multilingual Recursive Summarization Trees (MuRST) model. In our experiments, MuRST shows improved QA performance for long length documents across multiple languages. Additionally, we experiment with different clustering algorithms, including hard clustering using k-means, to further refine the organization of multilingual data in our models.

3 Related Work

Multilingual question answering (QA) has garnered significant attention in recent research due to the increasing need for systems that can understand and generate responses in multiple languages.

One recent approach employs a knowledge injection strategy combined with transformers to enhance semantic understanding across languages, facilitating the linking of knowledge between various languages and thereby improving the performance of multilingual QA models [6]. Similar to this approach, our project also emphasizes the inclusion of underrepresented languages in the development of QA models. Zhao et. al's study focuses on training a multilingual mBERT QA system specifically for low-resource Indian languages. MuCoT employs data augmentation techniques such as translation and transliteration of training data to enhance performance, which are crucial for improving model accuracy in languages with limited annotated data [14].

Recently, RAPTOR (Recursive Abstractive Processing for Tree-Organized Retrieval) introduced a novel approach to handling long-form texts in QA [13]. RAPTOR employs recursive summarization techniques, which break down long documents into smaller, manageable segments and recursively summarize these segments to generate concise and coherent responses [13]. This method has demonstrated significant improvements in processing long contexts within English QA systems, addressing the limitations of traditional transformer models that struggle with lengthy documents [13]. However, RAPTOR's application has been limited to English, and its potential benefits have not yet been explored in multilingual settings.

One of the primary challenges in advancing multilingual question answering (QA) systems is the scarcity of comprehensive and high-quality multilingual datasets. While substantial datasets exist for widely spoken languages such as English, there is a notable deficiency in resources for many other languages, particularly low-resource languages [2]. This lack of data hampers the training and evaluation of robust multilingual QA models, leading to performance disparities across languages. Most existing QA datasets are either monolingual or predominantly focused on high-resource languages, thereby neglecting the diverse linguistic landscape that modern QA systems aim to serve [2]. This gap underscores the need for dedicated efforts to develop and curate multilingual datasets that encompass a broader range of languages, which is crucial for building more inclusive and effective QA systems.

4 Approach

Our main approach involves comparing existing multilingual QA models (both XLM-RoBERTa-Base-SQuAD2 [3] and GPT-3.5-turbo) with our model, MuRST (Multilingual Recursive Summarization Trees), a multilingual implementation of RAPTOR.

Our multilingual model utilizes RAPTOR's code which we adapted to utilize open source, multilingual models (described below) for summarization, QA, and embeddings. Further, we also tested two different clustering algorithms. First, we used the default Gaussian Mixture Model clustering algorithm that was used in Vanilla RAPTOR. In addition, we also adapted RAPTOR's soft clustering algorithm to instead follow a hierarchical clustering process for embeddings, focusing on both global and local levels using the k-means algorithm and silhouette scores for hard clustering. By organizing data into hierarchical clusters, the system can handle multilingual data effectively, identifying and summarizing relevant information across different languages while preserving the structure and context.

4.0.1 Baseline

We begin by establishing our baseline methods. We used two models for our baseline: a multilingual QA model XLM-RoBERTa-Base-SQuAD2 and GPT-3.5-turbo. XLM-RoBERTa-Base-SQuAD2 is based on XLM-RoBERTa, a multilingual version of Facebook's RoBERTa model released in 2019. We chose it due to its proven performance in handling multiple languages effectively, leveraging a transformer architecture pre-trained on a diverse multilingual corpus [5]. This makes it well-suited for our goal of building a model that includes underrepresented languages. XLM-RoBERTa-Base-SQuAD2 is optimized for completion tasks, focusing on predicting and filling in masked tokens within a text. It excels at understanding and modeling the context to accurately complete sentences or phrases, making it ideal for applications that require precise text completion or contextual predictions based on incomplete inputs [5].

We also selected GPT-3.5-turbo, which has demonstrated robust capabilities to support multiple languages [1]. GPT-3.5-turbo's advanced language understanding and generation abilities, combined with its versatility across different languages and contexts, make it a strong candidate for our baseline comparison. It also was designed to generate responses based on user prompts, effectively leveraging its vast training data to produce coherent and contextually appropriate replies. This makes it particularly suitable for applications where it can handle a wide range of question scenarios by responding directly to the input prompt [1].

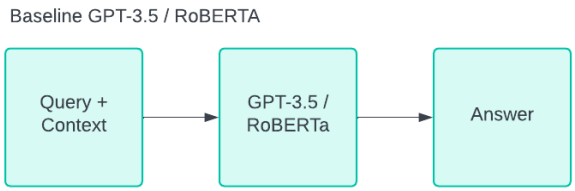

Figure 1: Model Flow for Baseline

4.0.2 Recursive Summarization with Soft Clustering

We ran experiments on our datasets using the following model configurations:

1. Vanilla RAPTOR
2. MuRST with GPT-3.5-turbo for QA
3. MuRST with XLM-RoBERTa-Base-SQuAD2 for QA

Both MuRST models listed above use mT5-multilingual-XLSum as the multilingual summarization model and multilingual multilingual-MiniLM-L12-v2 for embeddings. mT5-multilingual-XLSum is a large-scale summarization model based on the mT5 architecture and fine-tuned on the XL-Sum dataset,

which contains document summaries in 44 different languages. This model has displayed competitive ROUGE scores across nearly all languages for the task of document summarization, indicating its robust performance in generating coherent and relevant summaries across diverse languages [7]. For embeddings, we used multilingual-MiniLM-L12-v2, which has demonstrated competitive scores in semantic similarity tasks by effectively capturing the semantic nuances of multilingual text [12]. This combination ensures that our models benefit from state-of-the-art techniques in both summarization and semantic embedding.

Both the multilingual summarization model and embedding model were used in MuRST to build the recursive summarization tree. Afterwards, we tested the QA task with two different QA models: GPT-3.5-turbo and XLM-RoBERTa-Base-SQuAD2.

Note that the rest of our paper refers to MuRST with GPT-3.5-turbo for QA as 'MuRST-GPT-3.5' and MuRST with XLM-RoBERTa-Base-SQuAD2 for QA as 'MuRST-RoBERTa' for concision.

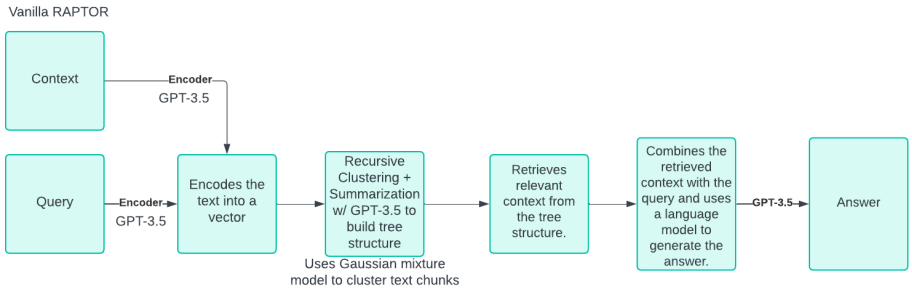

Figure 2: Model Flow for Vanilla RAPTOR

4.0.3 Recursive Summarization with Hard Clustering

We also ran experiments on MuRST with GPT-3.5-turbo for QA with hard clustering.

Vanilla RAPTOR uses GMM (Gaussian Mixture Models) for soft clustering, which groups similar word embeddings into clusters, allowing for flexible boundaries between clusters. In our approach, we adapted this algorithm to use hard clustering, which creates distinct and non-overlapping clusters of embeddings. Our hard clustering algorithm employs the following steps:

1. Dimensionality Reduction: We use UMAP (Uniform Manifold Approximation and Projection) to reduce the dimensionality of the multilingual embeddings. This step provides a global overview of the data, simplifying complex high-dimensional data into a lower-dimensional space while preserving the data's structure.
2. K-Means Clustering: After dimensionality reduction, we apply the K-Means algorithm to the reduced data. K-Means clustering partitions the data into K clusters, where each data point belongs to the cluster with the nearest mean. This method ensures that each cluster is distinct and cohesive.
3. Silhouette Scores: To determine the optimal number of clusters, we evaluate silhouette scores for different values of K. The silhouette score measures how similar an object is to its own cluster compared to other clusters. A higher silhouette score indicates better-defined clusters.
4. Recursive Refinement: If necessary, we recursively refine the clusters. Clusters that are too large or not well-defined are further subdivided using the same process until all clusters are of manageable size and high quality.

The motivation behind using hard clustering in the context of multilingual QA lies in its ability to create clear and distinct clusters [10]. This is particularly useful for handling multilingual data where languages might have diverse structures and semantics. Hard clustering can potentially improve the accuracy of retrieval and summarization by ensuring that each piece of information is clearly categorized, reducing overlap and ambiguity.

By organizing data into well-defined clusters, our system can better identify and summarize relevant information across different languages while preserving the structure and context. This approach can help in handling the variability in language and context, and we wish to test if it will improve the overall performance of the QA system for certain languages.

Note that the rest of our paper refers to MuRST with GPT-3.5-turbo for QA and hard clustering as 'MuRST-GPT3.5-Hard.'

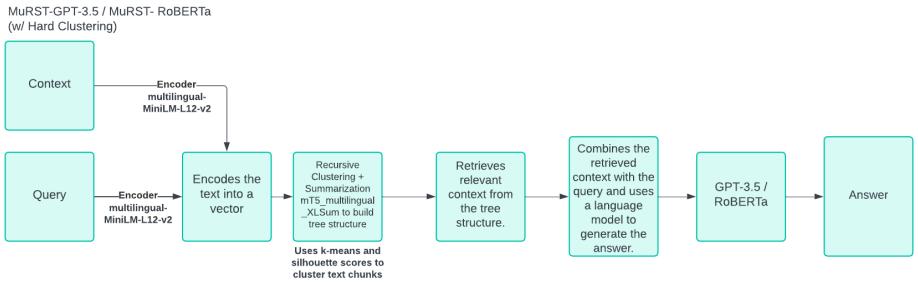

Figure 3: Model Flow for MuRST w/ Hard Clustering

5 Experiments

5.1 Datasets

We used two datasets, XQuAD for short documents and TyDi QA for long documents.

XQuAD (Cross-lingual Question Answering Dataset) [11] is a benchmark dataset used to evaluate the performance of multilingual question answering models. The dataset is designed to test the ability of models to understand and answer questions across multiple languages. XQuAD is a cross-lingual extension of the SQuAD (Stanford Question Answering Dataset), containing the same set of questions and answers translated into various languages. The average document length of XQuAD is 4-5 sentences, allowing us to test the impact of recursive summarization on short form documents.

TyDi QA is a question answering dataset that covers 11 different languages, where data was collected directly from each language without use of translation. The dataset consists of Wikipedia articles in each language as well as questions about the articles [4]. We used this dataset for multilingual QA with long form documents.

5.2 Evaluation methods

- **BLEU Scores**: BLEU (Bilingual Evaluation Understudy) [9] measures the similarity between the n-grams of a candidate text and reference texts, focusing on precision to evaluate the quality of machine-translated text.
- **F1 Scores**: The F1 score is the harmonic mean of precision and recall, providing a balanced measure of a model's accuracy by considering both false positives and false negatives.
- **Precision**: Precision is the ratio of correctly predicted positive instances to the total predicted positives, assessing the accuracy of the positive predictions.
- **Recall**: Recall is the ratio of correctly predicted positive instances to all actual positives, evaluating the model's ability to capture all relevant instances.

We chose BLEU and F1 because they are standard metrics used for QA that provide us a comprehensive understanding of the strengths and weaknesses of each model. We also chose to report on Precision and Recall for specific metrics that help us understand the model's ability to mitigate false positives and false negatives, and we used these metrics to gain a better understanding of the predicted versus true answers of our models.

5.3 Experimental details

We performed multilingual QA on the XQuAD dataset through the following 3 model configurations:

1. Baseline GPT-3.5
2. Baseline XLM-RoBERTa-Base-SQuAD2
3. MuRST-RoBERTa

After viewing our preliminary XQuAD results, we realized that XQuAD documents were too short to properly investigate recursive summarization, which is why we then focused on long form documents with TyDi QA.

We performed multilingual QA on the TyDi QA dataset through 4 model configurations with GPT-3.5 for QA and 3 configurations with XLM-RoBERTa-Base-SQuAD2 for QA:

GPT-3.5:

1. Baseline GPT-3.5
2. Vanilla RAPTOR
3. MuRST-GPT-3.5
4. MuRST-GPT-3.5-Hard Clustering

XLM-RoBERTa-Base-SQuAD2:

1. Baseline XLM-RoBERTa-Base-SQuAD2
2. Vanilla RAPTOR
3. MuRST-RoBERTa

To evaluate the baseline, we tested both our QA models (GPT-3.5-turbo and XLM-RoBERTa-Base-SQuAD2) on the XQuAD and TyDi QA datasets. We tested each model on 50 question-answer pairs for each language included in these datasets. This baseline evaluation did not involve recursive summarization: instead, the entire document was provided for context, allowing us to assess the models' performance on comprehensive multilingual QA tasks.

The rest of the model configurations all employ recursive summarization.

5.3.1 XQuAD Results

Language	XLM-RoBERTa-Base-SQuAD2 (baseline)	GPT-3.5 (baseline)	MuRST-RoBERTa
English	0.55	0.51	0.50
Chinese	0.45	0.10	0.45
Russian	0.40	0.53	0.38
Thai	0.27	0.51	0.26

Table 1: F1 Scores Across baselines and MuRST w/ XLM-RoBERTa-Base-SQuAD2 in XQuAD

Our results for XQuAD shown in Table 1 reveal that on short length multilingual documents, using recursive summarization for QA does not perform as well as our baseline. When analyzing the recursive tree formed from the documents, we saw that each tree consisted of only about 1 - 3 nodes, because the documents are so short that a longer-tree could not be constructed. Our results can be explained by the fact that a tree-based approach is not necessary to help traverse an already short document, and by splitting an already short text in chunks, the QA model loses long context that might be necessary to answer the question. This is not surprising, since the original RAPTOR paper was also not intended for short length documents, and why we chose to focus mostly on TyDi QA instead.

5.3.2 TyDi QA results

Our results in Table 2 show that recursive summarization models (both Vanilla RAPTOR and MuRST) outperformed the baselines in nearly all languages except for Japanese. The models showed

Language	Experiment	F1 Score	Precision	Recall	BLEU
English	GPT-3.5	0.0432	1	0.0222	0.3552
	MuRST-GPT-3.5	0.0821	1	0.045	0.4057
	MuRST-GPT3.5-Hard	0.0791	1	0.0434	0.391
	Vanilla RAPTOR	0.075	1	0.0406	0.3983
Japanese	GPT-3.5	0.1319	1	0.072	0.4371
	MuRST-GPT3.5	0.104	1	0.0567	0.4424
	MuRST-GPT3.5-Hard	0.1072	1	0.0584	0.4467
	Vanilla RAPTOR	0.0612	1	0.0318	0.3864
Thai	GPT-3.5	0.0865	1	0.0461	0.4027
	MuRST-GPT-3.5	0.1505	0.6	0.0876	0.3572
	MuRST-GPT3.5-Hard	0.1051	0.6	0.0596	0.3048
	Vanilla RAPTOR	0.1152	1	0.0631	0.442
Korean	GPT-3.5	0.1093	1	0.0584	0.4756
	MuRST-GPT-3.5	0.1648	1	0.0908	0.5348
	MuRST-GPT3.5-Hard	0.1773	1	0.0981	0.5517
	Vanilla RAPTOR	0.1794	1	0.1007	0.5432
Finnish	GPT-3.5	0.1672	1	0.0927	0.5322
	MuRST-GPT-3.5	0.2419	1	0.1386	0.607
	MuRST-GPT3.5-Hard	0.2419	1	0.1386	0.607
	Vanilla RAPTOR	0.2103	1	0.1193	0.572

Table 2: Performance on TyDi QA with GPT-3.5 based models across Different Languages

Language	Experiment	F1 Score	Precision	Recall	BLEU
English	XLM-RoBERTa-Base-SQuAD2	0.03428	0.8	0.0175	0.6079
	MuRST-RoBERTa	0.4	0.8	0.2667	0.6079
	Vanilla RAPTOR	0.075	1	0.0406	0.3983
Japanese	XLM-RoBERTa-Base-SQuAD2	0	0	0	0
	MuRST-RoBERTa	0.2	0.4	0.1333	0.3039
	Vanilla RAPTOR	0.0612	1	0.0318	0.3864
Thai	XLM-RoBERTa-Base-SQuAD2	0.02885	0.7	0.0147	0.3943
	MuRST-RoBERTa	0.22	0.6	0.1439	0.3943
	Vanilla RAPTOR	0.1152	1	0.0631	0.442
Korean	XLM-RoBERTa-Base-SQuAD2	0.0756	0.5	0.04	0.6079
	MuRST-RoBERTa	0.18	0.4	0.1167	0.2934
	Vanilla RAPTOR	0.1794	1	0.1007	0.5432
Finnish	XLM-RoBERTa-Base-SQuAD2	0.2106	0.9	0.1192	0.6079
	MuRST-RoBERTa	0.4	0.8	0.2667	0.6079
	Vanilla RAPTOR	0.2103	1	0.1193	0.572

Table 3: Performance on TyDi QA with XLM-RoBERTa-Base-SQuAD2 based models Across Different Languages

an improvement in F1 scores and BLEU scores by 3-8% and 4-7% respectively, depending on the language. This suggests that recursive summarization is generally effective in enhancing QA performance across multiple languages.

English and Finnish saw notable improvements when using MuRST, surpassing both Vanilla RAPTOR and the baseline models in F1 and BLEU scores. However, for the languages Japanese, Korean, and Thai this was not the case. This indicates that for certain languages, incorporating multilingual models can significantly enhance the recursive summarization process compared to using GPT-3.5 alone.

An interesting observation is that Japanese and Korean showed improvements with hard clustering when using MuRST, while the remaining languages did not. This could be attributed to the structural and linguistic characteristics of these languages. For instance, Korean and Japanese are relatively compact languages compared to English and Thai. Both languages also borrow extensively from Chinese, incorporating a significant number of Chinese characters and loanwords. The compact

nature of these languages, and the specific characters used in these languages, may be better suited for hard clustering.

With the XLM-RoBERTa-Base-SQuAD2 based models shown in Table 3, our results reveal that MuRST showed improvements in F1 scores across all languages. However, there was no clear trend observed with BLEU scores across different models. This implies that while recursive summarization improved precision and/or recall in answering questions (reflected by F1 scores), the quality of the generated text (measured by BLEU scores) did not show a consistent pattern.

6 Analysis

One of the significant challenges we encountered during our experiments was the discrepancy between the verbosity of the ground truth and the brevity of the predicted answers. This often led to low scores in standard evaluation metrics such as F1 and BLEU, despite the predicted answers being correct.

Example:

Question: 도널드 덕의 생일은 언제인가?
Ground Truth: 도널드 존 트럼프(, , 1946년 6월 14일)는 미국의 기업인 출신 제45대 대통령이다. 부동산 개발 등 다양한 사업을 하는 트럼프 기업의 대표이사 회장을 맡았으며, 트럼프 엔터테인먼트
Prediction: 6월 14일

Translated:
Question: *What is Donald Trump's Birthday?*
Ground Truth: *Donald John Trump (born June 14, 1946) is the 45th president of the United States. He served as CEO and Chairman of the Trump Enterprises, which engages in various businesses including real estate development, and Trump Entertainment.*
Prediction: *June 14th*

We found that predicted answers like this one, where the predicted answer is correct and brief, would receive very low F1 and BLEU scores when the ground truth was very verbose. While the predicted answer "6월 14일" (June 14th) is precise and correct, it lacks the detailed contextual information present in the ground truth, leading to an imbalance in evaluation metrics. This discrepancy highlights a fundamental issue with traditional evaluation metrics: they do not adequately capture the accuracy of concise answers when the ground truth is verbose.

This example illustrates a common issue encountered in our experiments: when the ground truth is verbose and the predicted answer is brief but correct, traditional evaluation metrics such as F1 and BLEU scores do not adequately capture the accuracy of the prediction.

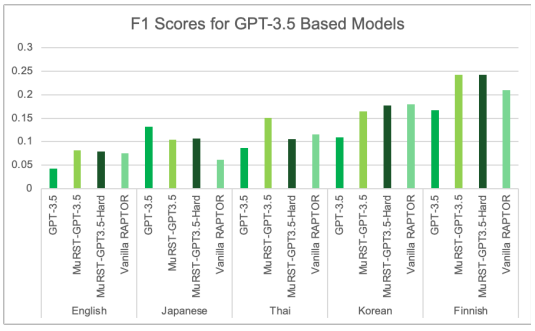

Figure 4: F1 Scores for GPT-3.5 Based Models

F1 score, for instance, is designed to balance precision and recall, but in cases where the ground truth is verbose, even a precise and fully correct answer like "June 14th" will have low recall because it matches only a small portion of the ground truth text. This discrepancy occurs because the recall component of the F1 score heavily penalizes the absence of the additional contextual information present in the ground truth, thus undervaluing the correctness and relevance of succinct, accurate answers.

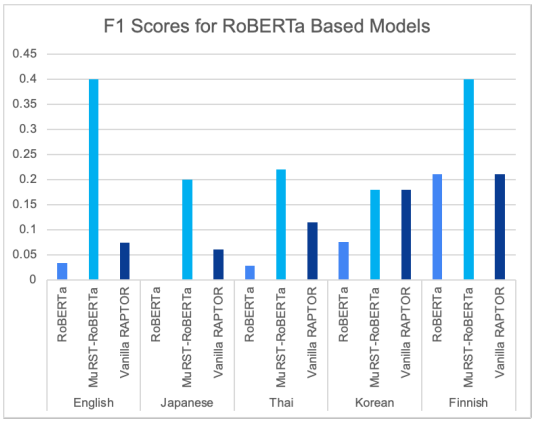

Figure 5: F1 Scores for XLM-RoBERTa-Base-SQuAD2 Based Models

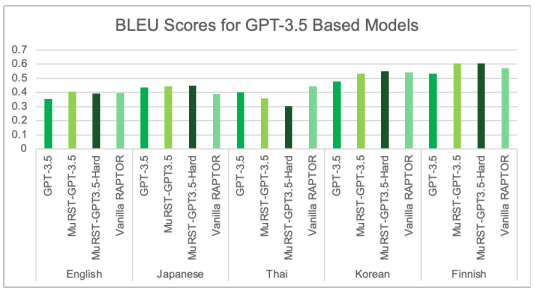

Figure 6: BLEU Scores for GPT-3.5 Based Models

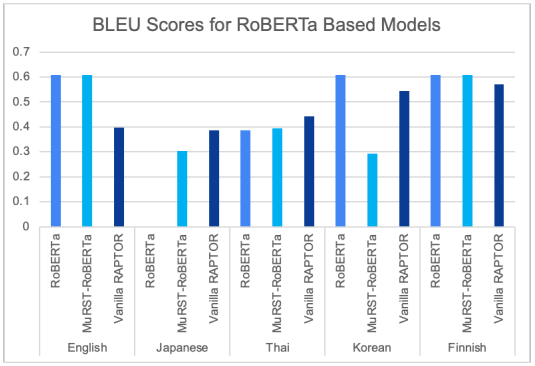

Figure 7: BLEU Scores for XLM-RoBERTa-Base-SQuAD2 Based Models

Similarly, BLEU scores, which are based on n-gram overlap, penalize the brevity by not rewarding the correct n-grams sufficiently unless they are embedded within a similarly verbose context. These metrics inherently favor answers that are longer and more detailed, aligning closely with the verbosity of the ground truth. As a result, they penalize concise answers that, while correct, do not include all the details present in the ground truth.

Qualitatively, the failure of traditional metrics to recognize the value of such brevity suggests a disconnect between model performance as measured by these metrics and real-world utility. This discrepancy is a critical area of concern for multilingual QA because accuracy is paramount in this case because it ensures that the information provided is reliable and precise, which is essential for maintaining integrity across diverse linguistic contexts.

This highlights the need for improved evaluation frameworks that can better balance the correctness and conciseness of answers.

7 Conclusion

The experiments demonstrated that recursive summarization trees, combined with multilingual embeddings and appropriate clustering techniques, can significantly improve the performance of multilingual QA systems. While MuRST models generally outperformed the baseline and vanilla RAPTOR across multiple languages, the degree of improvement varied, indicating areas for further research and optimization. Future work should focus on refining clustering algorithms, exploring more advanced multilingual models, and incorporating diverse datasets to enhance the robustness and accuracy of multilingual QA systems. Additionally, future work can explore language-specific clustering algorithms and language-specific approaches to tree building in general, which may further optimize performance for individual languages and address the unique challenges presented by different linguistic structures.

8 Ethics Statement

Multilingual question-answering (QA) systems pose several ethical issues and societal risks due to inherent challenges in handling diverse languages and cultures. One major concern is that the training data may not be equally representative of all languages, leading to a model that performs better for some languages than others. This discrepancy can perpetuate existing biases, privileging speakers of well-represented languages and marginalizing those who speak less common languages. Additionally, errors in question understanding or answer synthesis, especially across languages, can result in the dissemination of incorrect information. This is particularly problematic in critical contexts such as healthcare or legal advice, where misinformation can have serious consequences. One mitigation strategy for this issue would be to implement comprehensive data augmentation and balancing to ensure underrepresented languages are adequately included in the training data, thus promoting fairness and reducing biases in multilingual QA systems.

Furthermore, the challenge of considering cultural differences and sensitivities when retrieving and presenting information from diverse linguistic contexts can lead to misinterpretations, misunderstandings, or cultural insensitivity. Different words or phrases can have varying meanings and connotations across languages, and a failure to accurately capture these nuances may inadvertently reinforce stereotypes or perpetuate cultural hegemony. To mitigate these issues, one strategy is to actively curate and expand training datasets to include more diverse and underrepresented languages, ensuring more balanced performance. Additionally, integrating a layer of human review for high-stakes queries and implementing robust feedback mechanisms can help verify the accuracy and cultural appropriateness of responses, thereby fostering greater inclusivity and trust in the system.

References

[1] Kabir Ahuja, Harshita Diddee, Rishav Hada, Millicent Ochieng, Krithika Ramesh, Prachi Jain, Akshay Nambi, Tanuja Ganu, Sameer Segal, Maxamed Axmed, et al. Mega: Multilingual evaluation of generative ai. *arXiv preprint arXiv:2303.12528*, 2023.

[2] Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. On the cross-lingual transferability of monolingual representations. *arXiv preprint arXiv:1910.11856*, 2019.

[3] Branden Chan, Timo Möller, Malte Pietsch, and Tanay Soni. Multilingual xlm-roberta base for qa on various languages. 2024.

[4] Jonathan H Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. Tydi qa: A benchmark for information-seeking question answering in ty pologically di verse languages. *Transactions of the Association for Computational Linguistics*, 8:454–470, 2020.

[5] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*, 2019.

[6] Zhichao Duan, Xiuxing Li, Zhengyan Zhang, Zhenyu Li, Ning Liu, and Jianyong Wang. Bridging the language gap: Knowledge injected multilingual question answering. In *2021 IEEE International Conference on Big Knowledge (ICBK)*, pages 339–346. IEEE, 2021.

[7] Tahmid Hasan, Abhik Bhattacharjee, Md Saiful Islam, Kazi Samin, Yuan-Fang Li, Yong-Bin Kang, M. Sohel Rahman, and Rifat Shahriyar. Xl-sum: Large-scale multilingual abstractive summarization for 44 languages. 2021.

[8] Gokul Karthik Kumar, Abhishek Singh Gehlot, Sahal Shaji Mullappilly, and Karthik Nandakumar. Mucot: Multilingual contrastive training for question-answering in low-resource languages. *arXiv preprint arXiv:2204.05814*, 2022.

[9] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, 2002.

[10] Johannes Petrus et al. Soft and hard clustering for abstract scientific paper in indonesian. In *2019 International Conference on Informatics, Multimedia, Cyber and Information System (ICIMCIS)*, pages 131–136. IEEE, 2019.

[11] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In Jian Su, Kevin Duh, and Xavier Carreras, editors, *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas, November 2016. Association for Computational Linguistics.

[12] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019.

[13] Parth Sarthi, Salman Abdullah, Aditi Tuli, Shubh Khanna, Anna Goldie, and Christopher D Manning. Raptor: Recursive abstractive processing for tree-organized retrieval. *arXiv preprint arXiv:2401.18059*, 2024.

[14] Wen Zhang, Yang Feng, Fandong Meng, Di You, and Qun Liu. Bridging the gap between training and inference for neural machine translation. *arXiv preprint arXiv:1906.02448*, 2019.