ISSN 2277-2685
IJESR/June. 2025/ Vol-15/Issue-3s/288-297
Gundla Harshitha *et. al.,* / International Journal of Engineering & Science Research

Comparing Human and LLM Annotations in Low-Resource Language NLP Tasks

A.Hima Bindu, Gundla Harshitha, Sangannagare Chandana

[1]Assistant Professor, Department Of Cse, Bhoj Reddy Engineering College For Women, India.

[2,3]B. Tech Students, Department Of Cse, Bhoj Reddy Engineering College For Women, India.

ABSTRACT:

In Natural Language Processing (NLP), annotated datasets play a crucial role in training and evaluating machine learning models. However, in low-resource languages, the availability of high-quality annotated data is extremely limited due to linguistic complexity, lack of standardization, and scarcity of expert annotators. With the rise of Large Language Models (LLMs), such as GPT and similar models, there is growing interest in using these models to generate annotations automatically. This study compares human-generated annotations with those generated by LLMs for NLP tasks such as part-of-speech tagging, named entity recognition, and sentiment analysis in low-resource languages. The comparison is based on precision, recall, and F1-score, along with qualitative analysis. Our findings show that while LLMs can provide reasonable annotations in many cases, human annotations still outperform them in linguistic nuance, context understanding, and domain specificity. However, LLMs show potential in speeding up the annotation process and supporting human annotators through pre-annotation. This research highlights the complementary strengths of humans and LLMs and proposes a hybrid annotation workflow for building better NLP resources in low-resource settings.

1. INTRODUCTION

Annotated data plays a pivotal role in the development of machine learning (ML) models, especially in natural language processing (NLP) tasks. Annotation is the process of labeling or tagging raw data to make it meaningful and useful for machine learning (ML) and artificial intelligence (AI) models. In natural language processing (NLP), annotation involves adding information such as parts of speech, named entities, sentiment labels, or syntactic structures to text data. This labeled data becomes the foundation for training ML algorithms to understand, predict, or generate language.Traditionally, human annotation has been the gold standard for creating high-quality labeled datasets. However, the emergence of large language models (LLMs), such as OpenAI's GPT or Google's Bard, offers an alternative approach: automated or semi-automated annotation.

2. LOW-RESOURCE LANGUAGES

Low-resource languages refer to those that have limited availability of annotated data, computational resources, and linguistic expertise, making the development of natural language processing (NLP) models more challenging. These languages often lack extensive corpora, annotated datasets, and tools like parsers, stemming from smaller speaker populations, fewer computational resources, and less investment in linguistic research compared to high-resource languages such as English, Spanish, or Chinese.

In NLP, models heavily rely on large amounts of labeled data for training, yet many low-resource languages suffer from data scarcity, making it difficult to achieve high-performance results. As a

result, tasks such as machine translation, sentiment analysis, named entity recognition, and others face significant hurdles due to the lack of foundational resources.

The interest in Natural Language Processing (NLP) systems has grown significantly over the past few years and software products containing NLP features are estimated to globally generate USD 48 billion by 2026. However, current NLP solutions majorly focus on one of the few high-resource languages like English, Spanish or German although there are about 3 billion low-resource language speakers (mainly in Asia and Africa). Such a large portion of the world population is still underserved by NLP systems because of various challenges that developers face when building NLP systems for low-resource languages

Challenges for low-resource languages:

Lack of annotated datasets: Annotated datasets are necessary to train Machine Learning (ML) models in a *supervised* fashion. These models are commonly used to solve specific tasks very accurately, like hate speech detection. However, creating annotated datasets requires human intervention by labelling training examples one by one, making the process usually time-consuming and very expensive given the thousands of examples advanced deep learning models require. Thus, it becomes infeasible to rely on only manual data creation in the long run.

Lack of unlabelled datasets: Unlabelled datasets like text corpora are the precursors to their annotated versions. They are essential for training *base* models that are later fine-tuned for specific tasks. Hence, approaches to circumvent the lack of unlabelled datasets also become very important.

Limited Linguistic Resources: Low-resource languages often face a severe lack of standardized linguistic tools such as dictionaries, grammars, and annotated corpora. These resources are essential for key NLP tasks like part-of-speech tagging, named entity recognition, and syntactic parsing. Without such tools, it becomes difficult to develop accurate models that can effectively capture the structure and semantics of the language.

Supporting multiple dialects of a language: Languages that have multiple dialects are also a tricky problem to solve, especially for speech models. A model trained in a language usually won't perform great in its different dialects. For example, most unlabelled and annotated datasets available for Arabic are in Modern Standard Arabic. However, for a human-like feeling when interacting with voice or chat assistants for daily use it is too formal for many Arabic speakers. Thus, supporting dialects become necessary for practical use cases.

Natural Language Processing (NLP) in low-resource languages

Natural Language Processing (NLP) in low-resource languages focuses on developing computational techniques for understanding and generating human language in scenarios where linguistic resources such as annotated datasets, lexicons, or pre-trained models are limited or unavailable. These languages, often spoken by smaller or marginalized communities, face challenges such as a lack of digital documentation, inconsistent orthographies, and limited research interest. As a result, conventional NLP techniques that rely on large-scale data struggle to perform effectively.

To address these challenges, researchers employ strategies like transfer learning, where models trained on high-resource languages are adapted to low-resource contexts, and unsupervised or semi-supervised learning techniques, which minimize the need for labeled data. Community-driven

efforts to crowdsource linguistic data and collaborative projects like Universal Dependencies also play a critical role. Additionally, advances in multilingual pre-trained language models like mBERT, XLM-R, and LLaMA enable significant progress by leveraging shared linguistic features across languages.

3.HUMAN GENERATED ANNOTATIONS

Human-annotated data is essentially information that has been manually reviewed, labeled, or classified by individuals. This process involves human annotators who understand the context of the data, whether it's text, images, audio, or video. The human element in annotation provides a layer of cognitive understanding that purely automated systems may not fully capture.

Human annotation played a pivotal role in ensuring the quality and reliability of the labeled data used for our Natural Language Processing(NLP) tasks. The process encompassess annotator selection,training, the formulation of annotation guidelines, the choice of annotation platform, assessment of inter-annotator agreement, and strategies to address encountered challenges.The selection of human annotators was a critical step in annotation process.

Human-generated annotations involve the process of labeling, categorizing, or tagging data based on human judgment, expertise, and contextual understanding. This approach is essential, especially in scenarios where automated systems struggle due to a lack of sufficient training data, complexity, or domain-specific knowledge. In tasks like Natural Language Processing (NLP), human annotators play a critical role in ensuring accurate and contextually rich annotations, particularly in low-resource languages or specialized domains. Humans are capable of understanding nuanced meanings, idiomatic expressions, cultural contexts, and subtleties that automated systems may overlook. For example, they can distinguish between ambiguous phrases, recognize sarcasm, or interpret complex linguistic structures.

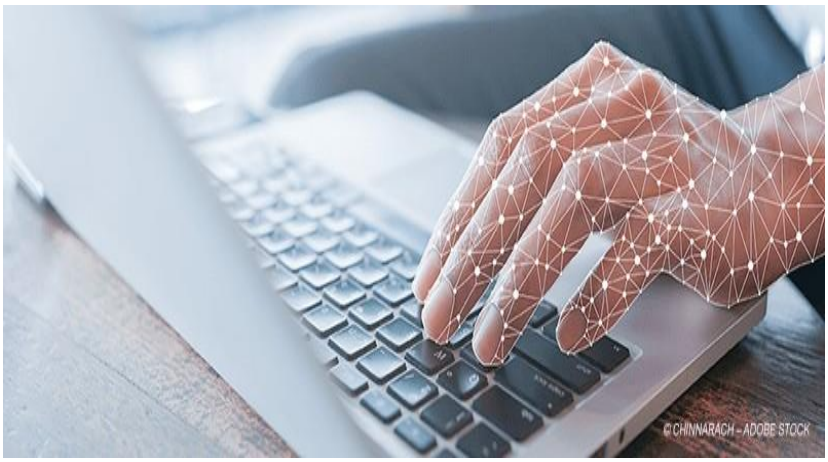


Fig. 3.1 Diagram of Human Generated Annotation

Moreover, in low-resource settings where there is a scarcity of annotated data, human annotators help fill the gap by providing annotated datasets crucial for training models. However, despite these advantages, human annotations come with limitations. They are time-consuming, expensive, and prone to inconsistencies due to subjectivity or fatigue. Additionally, finding qualified annotators, especially for niche domains or low-resource languages, can be difficult. Despite these

challenges, human-generated annotations remain valuable, especially when high accuracy and contextual relevance are required, as they contribute to building high-quality training data that automated systems rely on.

Steps in Human Annotation Process

1. **Defining Annotation Guidelines**

Defining annotation guidelines is a critical initial step to ensure consistency, accuracy, and clarity in the annotation process. These guidelines serve as the foundation for the entire annotation task, helping annotators apply consistent standards and interpretations across the dataset. Clear guidelines reduce ambiguity and ensure that all annotators follow the same rules, resulting in high-quality annotations.

The first component of annotation guidelines is providing clear definitions of the annotation task. For example, if the task involves tagging named entities in text, annotators need precise definitions of what constitutes a "named entity" (e.g., person names, dates, locations, etc.). Without clear definitions, annotators may interpret these entities differently, leading to inconsistencies in the annotations.

Next, examples and counterexamples play a key role in guiding annotators. Providing annotated examples showing correct and incorrect annotations helps illustrate how to apply the guidelines. For instance, if annotating sentences for sentiment analysis, annotators should be shown examples of sentences labeled as “positive,” “negative,” or “neutral” with explanations of why those labels were chosen. Counterexamples (incorrect annotations) also help identify common pitfalls, ensuring annotators understand what not to do.

By offering detailed instruction sets and examples, annotation guidelines reduce confusion and provide annotators with the necessary tools to make accurate judgments.

2. **Selecting Annotators**

The success of the annotation process heavily relies on selecting the right annotators. Annotators need specific expertise, domain knowledge, or familiarity with the language or data type being annotated. The first step in this phase is recruiting domain experts, linguists, or trained individuals who have the required background and understanding of the task.

For example, if the task involves annotating medical data, hiring medical professionals or individuals with experience in healthcare is crucial. Similarly, for annotating low-resource languages, native speakers or linguists with familiarity in those languages are essential.

Matching annotators' skills with the specific requirements of the task is critical. A mismatch can lead to low-quality annotations. For instance, if annotators lack knowledge of the context, cultural nuances, or domain-specific jargon, their annotations may be less accurate. Matching annotators with tasks they are familiar with ensures they can apply guidelines effectively and avoid errors.

Finally, vetting and training annotators is another essential part of this process. Annotators should undergo training sessions where they are familiarized with the annotation guidelines, expected standards, and tools they’ll be using. Training also provides an opportunity to clarify any doubts or ambiguities about the annotation process.

3. **Data Annotation**

Once annotators are selected and trained, they begin the actual task of manually labeling or tagging data using annotation tools. Annotation tools provide structured environments that facilitate data tagging and help annotators maintain

consistency across a dataset.

During this step, annotators apply the predefined annotation guidelines, often guided by annotation tools like Prodigy, brat, Labelbox, or CVAT, depending on the data type (text, images, audio). For example, in text annotation tasks, annotators might highlight named entities, classify sentiment, or categorize text segments. In image annotation, they may draw bounding boxes around objects or label specific regions.

To ensure quality control and reduce errors, feedback loops are implemented. Annotators may be required to work iteratively, where feedback from supervisors or other annotators is provided. This feedback ensures that any errors or inconsistencies are caught early and corrected. Regular communication with annotators is necessary to address any questions they may have regarding the guidelines or tasks.

4. Quality Control

Quality control is a crucial step to ensure the consistency and reliability of the annotations. Thism process involves measuring how well annotators adhere to the guidelines and identify inconsistencies or errors in annotations.

4.LARGE LANGUAGE MODELS

Large Language Models (LLMs) are deep learning models designed to understand and generate human-like text. They are built on the foundation of Transformer architectures, which allow them to capture long-range dependencies and context from vast amounts of text data. LLMs are trained on massive datasets, often comprising billions to trillions of words from diverse sources, such as books, articles, websites, and other textual content. This extensive training enables them to acquire a wide range of knowledge, grammatical rules, and patterns of human language.

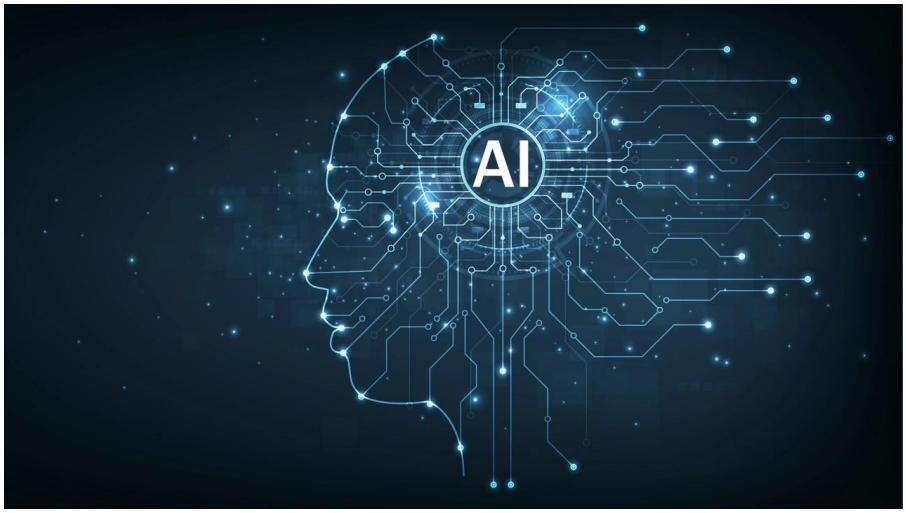


Fig. 4.1 Diagram of Large Language Model

The key idea behind LLMs is the self-attention mechanism, which allows the model to weigh the significance of different words in a sequence, helping it understand the relationships between words and capturing long-range dependencies. Unlike traditional NLP models, LLMs have a much larger number of parameters—sometimes in the order of billions—allowing them to learn complex representations of language with greater generalization capability.

One of the defining characteristics of LLMs is their ability to perform a wide range of tasks with minimal task-specific training, thanks to their pre-training and fine-tuning framework. They excel at

tasks such as text generation, translation, summarization, question answering, sentiment analysis, and more. Pre-training on large corpora ensures that LLMs develop a broad understanding of language, which can then be fine-tuned on specific tasks or domains to adapt to particular needs.

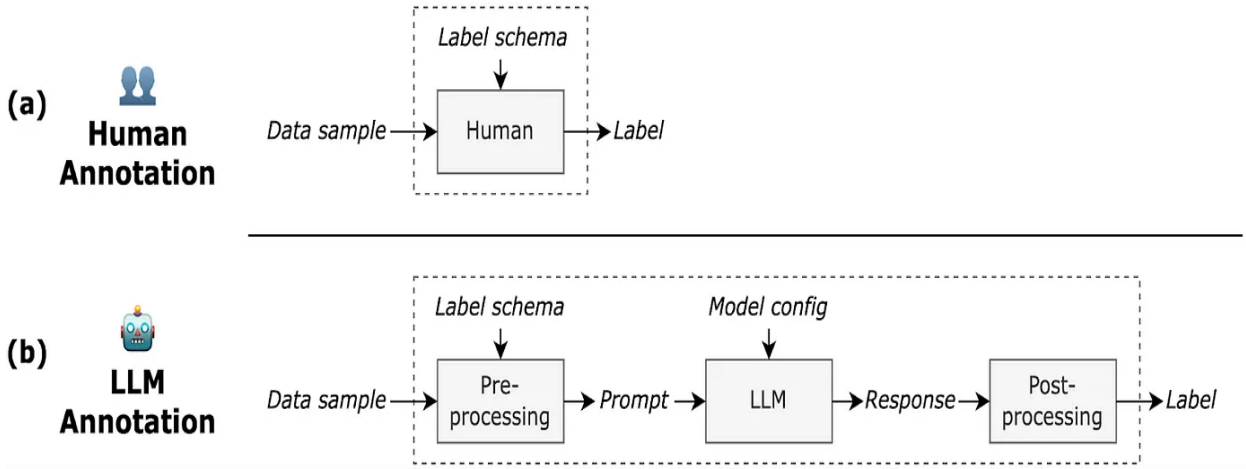
Popular examples of LLMs include GPT (Generative Pre-trained Transformer), BERT (Bidirectional Encoder Representations from Transformers), and newer models like GPT-4 and LLaMA. These models have demonstrated remarkable capabilities in generating coherent and contextually relevant text, answering complex questions, and even creative tasks like content generation and storytelling.

However, LLMs also pose challenges, such as high computational costs due to their large model size, environmental concerns related to energy consumption, and potential biases in the training data. Despite these limitations, LLMs are at the forefront of NLP advancements, pushing the boundaries of what machines can understand and generate in human language. Their versatility and scalability make them critical tools in a wide array of applications, from improving search engines to enhancing AI-driven customer interactions.

LLM generated annotations

Large Language Models (LLMs) generate annotations by leveraging their extensive training on diverse datasets. These models can produce detailed and contextually relevant labels or explanations for text in various languages. The annotations are created based on patterns and semantic understanding acquired during training, making them highly adaptable across different domains and languages, including low-resource ones. LLMs can analyze context, syntax, and semantics simultaneously, providing annotations that capture nuances often missed by traditional rule-based methods.

Additionally, LLM-generated annotations are efficient, enabling large-scale annotation tasks to be completed in significantly less time than manual efforts. They can also handle noisy or incomplete data, offering plausible annotations even when context is limited. However, their performance might still depend on the amount and diversity of data in their training corpus. In low-resource NLP, LLM annotations can significantly boost development by acting as a scalable solution for data scarcity, though ensuring cultural and linguistic accuracy remains a challenge.



Steps in LLM Annotation Process

Fig.5.1:Inputs and outputs of steps in human annotation and LLM annotation. LLM annotation requires

additional pre-processing and post-processing steps.

Step 1: Input Data Preprocessing

Before annotation begins, the input data must be preprocessed to ensure compatibility with the LLM. This involves cleaning the data to remove noise, tokenizing text into manageable units, and potentially translating it into a language the LLM is trained on if working with low-resource languages. Preprocessing ensures the input is in a format that the model can interpret effectively, minimizing errors during annotation. This step may also include identifying specific fields or tags for annotation to narrow the LLM's focus.

Step 2: Model Selection and Configuration

Choosing the appropriate LLM and configuring it for the task is crucial. The selection depends on the nature of the data and the desired output. For instance, general-purpose models like GPT-4 or task-specific fine-tuned models might be used. Configuration involves setting parameters such as temperature (controlling randomness) and max tokens (defining the length of annotations). These settings ensure the model generates consistent and relevant annotations suited to the task's requirements.

Step 3: Generating Annotations

The core step is the annotation process, where the LLM analyzes the input text and generates labels, tags, or explanations. This is done by leveraging its pre-trained knowledge and contextual understanding. The model generates annotations based on predefined instructions or prompts tailored to the task. For example, in sentiment analysis, the LLM might tag text as positive, negative, or neutral, while in a translation task, it could provide linguistic or syntactic annotations.

Step 4: Quality Assessment and Refinement

Once the LLM generates the annotations, they must be evaluated for accuracy and relevance. This step might involve manual review by experts or automated validation methods comparing the output against a ground truth dataset. For tasks in low-resource languages, additional checks might be necessary to ensure cultural and linguistic appropriateness. If the annotations are not up to the mark, adjustments in the prompts or model parameters are made, and the process is repeated. This iterative approach refines the output and improves the annotation quality.

Step 5: Post-Processing and Integration

After quality assessment, the annotations are post-processed to fit the specific format or structure required for the application. This might include converting annotations into JSON, XML, or other formats suitable for integration into downstream tasks like machine learning models or databases. For multilingual data, annotations may also be harmonized across languages to ensure consistency. The final annotations are then integrated into the workflow, ready to be used for training models, building datasets, or enhancing applications.

Step 6: Feedback and Model Improvement

The last step involves gathering feedback from users or domain experts to identify any shortcomings in the LLM’s performance. This feedback is invaluable for improving the annotation process, either by refining the prompts or fine-tuning the model with additional domain-specific data. Iterative feedback loops ensure that the annotation process evolves over time, delivering better results and adapting to the nuances of specific tasks or languages.

5-Comparing Human generated and LLM generated Annotations

When evaluating annotations generated by humans versus LLMs (Large Language Models), it is essential to understand the underlying strengths, limitations, and use cases of both approaches. Human-generated annotations rely on contextual understanding, domain expertise, and subjective judgment, allowing for a nuanced, informed approach that considers complex and subtle distinctions in language. Humans can leverage cultural knowledge, regional variations, and linguistic intricacies, ensuring high-quality and contextually accurate annotations. On the other hand, LLM-generated annotations rely on vast amounts of pre-trained data and sophisticated language models, providing high-speed and scalable solutions, especially for tasks involving large volumes of data. LLMs excel in capturing broad statistical patterns and general language structures, making them well-suited for tasks that require large-scale annotation and consistency. However, their performance often falls short in handling domain-specific knowledge, nuanced contextual understanding, and low-resource languages.

Below, we compare key metrics to highlight their differences and the implications for tasks, especially in low-resource languages:

Comparison Metrics

1. **Accuracy and Consistency**

- **Human-Generated Annotations:**
  Human annotators bring domain expertise, cultural understanding, and nuanced judgment to the annotation process. This makes human annotations highly accurate, especially for complex or subjective tasks. However, consistency can vary across annotators, and personal biases may influence results.
- **LLM-Generated Annotations:**
  LLMs are consistent in generating annotations, as they rely on learned patterns rather than subjective judgment. While they can replicate general knowledge accurately, they may struggle with nuances, leading to occasional errors, especially in domain-specific or culturally sensitive contexts.

2. **Speed and Scalability**

- **Human-Generated Annotations:**
  Humans are slower compared to LLMs. Annotating large datasets can take weeks or months, and the process becomes resource-intensive as dataset size increases.
- **LLM-Generated Annotations:**
  LLMs can annotate massive datasets within minutes or hours, making them highly scalable. This is particularly advantageous for tasks requiring quick results or when dealing with enormous volumes of data.

3. **Cost-Effectiveness**

- **Human-Generated Annotations:**
  Hiring, training, and compensating annotators can be expensive, especially for large-scale projects. Costs increase further for tasks requiring domain experts.
- **LLM-Generated Annotations:**
  LLMs reduce the need for human labor, lowering overall costs for annotation tasks. However, the computational resources required to run LLMs can still be expensive, especially for large-scale or real-time tasks.

6-HYBRID APPROACH

1. Combining human-generated annotations with those from Large Language Models (LLMs) offers a robust framework for enhancing the

accuracy and efficiency of annotation processes in NLP tasks. Human annotations typically carry domain expertise and contextual understanding, crucial in tasks that require nuanced interpretation, such as sentiment analysis, part-of-speech tagging, or entity recognition. LLM-generated annotations, on the other hand, bring scalability and rapid processing capabilities due to their ability to leverage vast amounts of pre-existing knowledge. The integration of both types of annotations can lead to a more balanced approach, combining the precision of human expertise with the breadth of coverage and efficiency of LLMs.

2. Human annotations are often detailed, nuanced, and context-aware, which is especially useful in complex, low-resource languages or specialized domains where rule-based approaches may struggle. However, they tend to be time-consuming and costly due to the manual effort required. LLMs, conversely, can quickly generate annotations by predicting patterns based on vast datasets, making them suitable for tasks where large-scale data processing is needed. However, the accuracy of LLM-generated annotations can be compromised when handling rare or domain-specific concepts due to their generalist training data.
3. By combining the two, organizations can benefit from the strengths of both approaches. Human annotators can focus on validating, correcting, and contextualizing LLM-generated outputs, ensuring that high-level decisions remain informed by expert knowledge. Meanwhile, LLMs can be employed to preprocess data, reduce redundancy, and enhance coverage across vast datasets. This hybrid approach not only reduces manual effort but also maintains high-quality annotation standards while ensuring scalability.
4. Lastly, future research could explore more dynamic models that allow for adaptive blending of LLM outputs and human corrections. This would enable a more efficient allocation of resources by balancing human oversight and automated predictions. For example, LLMs could flag low-confidence annotations, prompting human reviewers to validate or adjust these outputs, while routine or well-understood entity recognition can be handled by LLMs alone. By
5. continuously evolving hybrid systems and enhancing LLM capabilities, the field is likely to witness a more efficient, scalable, and high-quality approach to annotation tasks, especially in low-resource settings

5. CONCLUSION

The comparison between human-generated and LLM-generated annotations reveals distinct strengths and limitations for each approach. Human annotations excel in capturing nuanced cultural, contextual, and linguistic intricacies, particularly in low-resource NLP languages. However, they are time-consuming and require domain expertise, leading to scalability challenges. On the other hand, LLM-generated annotations offer speed, scalability, and cost-effectiveness, but they may struggle with cultural context and linguistic diversity, especially in underrepresented languages.

To maximize the benefits, a hybrid approach that combines the contextual richness of human annotations with the efficiency of LLMs can be a promising direction. Future work should focus on refining LLMs for low-resource languages

through improved training data, fine-tuning, and active learning techniques, paving the way for more inclusive and accurate NLP systems.

REFERENCES

[1] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, ‘‘Bert: Pre-training of deep bidirectional transformers for language understanding,’’ 2018, arXiv:1810.04805.

[2] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V.Stoyanov, ‘‘Roberta: A robustly optimized BERT pretraining approach,’’ 2019, arXiv:1907.11692.

[3]A. Mastropaolo, S. Scalabrino, N. Cooper, D. Nader Palacio, D. Poshyvanyk, R. Oliveto, and G. Bavota, ‘‘Studying the usage of text-to-text transfer transformer to support code-related tasks,’’ in Proc. IEEE/ACM 43rd Int. Conf. Softw. Eng. (ICSE), May 2021, pp. 336–34