Assessing Factual Accuracy in Machine Generated Cross Lingual Summaries using Logistic Regression and BERT

Kushaal Shyam Potta[1], Jaswanth Sridharan[1], Mahadev Ramesh Ramya[1], Shriram Gopalakrishnan[1] and Durairaj Thenmozhi[1]

[1]Sri Sivasubramaniya Nadar College Of Engineering, Rajiv Gandhi Salai (OMR), Kalavakkam, 603 110, Tamil Nadu, India

Abstract

Text classification poses a significant challenge, especially with the rise of AI-generated text on various social media platforms, where discerning authenticity and accuracy becomes critical. Through the ILSUM 2024 shared task, we aim to bridge this gap by applying supervised machine learning algorithms to classify text into various categories. Our team concentrated on the Gujarati and Hindi datasets, utilizing machine learning models such as logistic regression, logistic regression with class weights,and transformer models such as BERT, and BERT with focal loss to classify text. Notably, logistic regression with class weights produced a F1 score of 0.3371 in Gujarati, while BERT with focal loss produced a F1 score of 0.3426 in Hindi, indicating the effectiveness of specialized techniques for these languages. Our models achieved an overall rank of 1, based on their highest F1 scores.

Keywords

Text Classification, Machine Learning, Natural Language Processing, Transformer Models, Multiple Classification, Logistic Regression

1. Introduction

With the rapid growth in AI-driven content generation, machine-generated summaries are becoming more common, providing readers with concise information from lengthy documents. However, the factual reliability of these summaries is a critical concern, especially when generating content across languages. This research addresses this challenge, aiming to detect and categorize factual inaccuracies in machine-generated summaries. Specifically, we investigate these inaccuracies in cross-lingual contexts, where the source content is in English, and the summaries are in Hindi or Gujarati.

This work builds on previous efforts to analyze factual consistency in generated text but extends the focus to multilingual settings. The identified categories of factual errors—Misrepresentation, Inaccurate Quantities or Measurements, False Attribution, and Fabrication—cover various forms of distortion that can undermine the credibility of machine-generated summaries. Identifying these errors can improve the trustworthiness of generated content and pave the way for more robust cross-lingual AI applications.

Through our participation in the shared task 2 of Indian Language Summarization (ILSUM) namely: Detecting Factual Incorrectness in Machine Generated Cross Lingual Summaries[1] [2] [3] [4] [5] [6] [7], we have elucidated our approach towards the same in the following sections.

2. Related Works

In recent advancements, natural language processing (NLP) has leveraged both traditional machine learning algorithms and large language models (LLMs) for tasks like cross-lingual summary classification. Cross-lingual summary classification is particularly challenging as it requires the model to recognize and accurately classify summaries from multiple languages. Adding the aspect of multi-label classification brings extra complexity in terms of data preprocessing and classification of output labels.

Forum for Information Retrieval Evaluation, December 12-15, 2024, India

kushaalshyam2310513@ssn.edu.in (K. S. Potta); jaswanth2310325@ssn.edu.in (J. Sridharan); mahadev2310651@ssn.edu.in (M. R. Ramya); shriram2310156@ssn.edu.in (S. Gopalakrishnan); theni_d@ssn.edu.in (D. Thenmozhi)

0000-0003-0681-6628 (D. Thenmozhi)

Liu et al. [8] has expressed the increasing attention towards multilabel learning. Through performing an elastic net penalty on the logical regression model, the overfitting of high-dimensional data is reduced.

Aseervatham et al. [9] talked about the high efficiency of ridge logistic regression with respect to textual classification problems. The advantage of such classification is the computing of probability value instead of a score.

Shah et al. [10] created a comparative analysis of logistic regression, random forest and KNN models for text classification on a BBC news data set. The logistic regression classifier with the TF-IDF vectorizer feature attained highest accuracy of 0.97 with their dataset.

Yu et al. [11] proposed a scalable solution to the extreme multi-label text classification problem (XMC) and achieved state-of-the-art performance in the same by creating X-BERT, a deep learning approach built using fine-tuned versions of BERT models.

Further research by Bhargava et al. [12] involves utilizing multilingual models like mBERT, XLM-R for text classification of indian languages in the manner of transfer learning, which means the model is trained only on one or two languages and made to classify data in a new language.

Deroy et al. [13] delves into the classification of Gujarati and Hindi texts utilizing LLMs and zero shot prompting, highlighting the importance of LLMs, which have gained traction in the recent years, in the field of text classification. The text is preprocessed and used by the GPT-3.5 Turbo LLM model to generate accurate labels for classification of data.

3. Dataset Description

The dataset for this task includes both training and test sets with multilingual summaries and source articles. The training set comprises columns including `Id`, `Title`, `Headlines`, `Article`, `Incorrect_Summary`, `Incorrectness_Type`, `Correct_Summary`, `Incorrect_Summary_Hindi`, `Correct_Summary_Hindi`, `Incorrect_Summary_Gujarati`, and `Correct_Summary_Gujarati`. Each record represents an English article along with its machine-generated summaries in English, Hindi, and Gujarati, along with the respective correct summaries for each. The `Incorrectness_Type` column categorizes summaries based on factual errors. Instances labeled as `NaN` under `Incorrectness_Type` indicate summaries that were factually correct; these records were assigned an additional `Correct` label to distinguish them from other error categories.

For the test set, the data is split into Hindi and Gujarati summary subsets, each containing the columns `id_new`, `article`, and `summary`. Only the English article and its respective Hindi or Gujarati summary are provided in the test set, simulating a real-world setting where summaries are generated in local languages based on English source material.

This dataset structure allows for thorough model training on factual correctness across languages, covering scenarios from simple translation to complex semantic interpretation of facts in summaries across English, Hindi, and Gujarati. By including both correct and erroneous summaries, the dataset enables models to discern factual inaccuracies effectively while handling the nuances of multilingual data alignment. The provided dataset was then preprocessed according to the requirement of the models used, as explained in the later sections of the paper.

3.1. Task Description

The task involves detecting factual inaccuracies in machine-generated cross-lingual summaries based on an English source document. Given an English source document and its summaries in Hindi or Gujarati, the objective is to classify each summary as either factually correct or containing one of four error types:

- Misrepresentation: Information is presented misleadingly, often by exaggerating or downplaying certain aspects to alter the narrative.
- Inaccurate Quantities or Measurements: Errors occur when quantities.

- False Attribution: Statements, ideas, or actions are incorrectly credited to the wrong person or group.
- Fabrication: Involves creating fictitious data, events, or sources without any factual basis.

Task	Language	No. of Samples
Training	Gujarati	4975
Testing	Gujarati	200
Training	Hindi	4975
Testing	Hindi	200

Table 1
Distribution of text samples across training and testing for each language

4. Approach

We trained different machine learning and transformer models such as Logistic Regression, Logistic Regression with Class Weights and BERT base Uncased, BERT with Focal Loss on the training dataset, evaluated the models and submitted our runs by applying the ML and transformer models on the test dataset.

4.1. Data Preprocessing

Given the diverse models employed, the dataset required distinct preprocessing steps for each approach:

4.1.1. Logistic Regression:

- **Text Combination**: The `Article` and `Incorrect_Summary_Gujarati` columns were combined into a single text input for each record to ensure the model received context from both the source document and the summary.
- **TF-IDF Transformation**: We applied TF-IDF vectorization to convert the text data into numerical features, capturing the importance of terms within the document context. This transformation yielded a sparse representation suitable for Logistic Regression.
- **Label Encoding**: The target labels in the `Incorrectness_Type` column were label-encoded into integer values, preparing them for classification.

4.1.2. Logistic Regression with Class Weights:

- **Text Combination and TF-IDF**: The same text combination and TF-IDF vectorization steps were applied as in the Logistic Regression baseline.
- **Class Weighting**: Given the dataset's imbalance, where "Correct" labels were predominant, class weights were automatically set to "balanced" to assign greater importance to minority classes during training.

4.1.3. BERT Base Uncased:

- **Text Tokenization**: BERT required the input text to be tokenized, so we used the BERT tokenizer to split the combined `Article` and `Incorrect_Summary_Hindi` columns into tokens. The tokenizer handled wordpiece tokenization and added special tokens like `[CLS]` and `[SEP]` for BERT's input format.
- **Padding and Truncation**: To standardize input length, tokenized text was padded or truncated to a fixed length (512 tokens) suitable for BERT. This step ensured that the model could efficiently process each example within its input size limits.

4.1.4. BERT with Focal Loss:

- **Tokenization, Padding, and Truncation**: This approach followed the same tokenization and input standardization steps as BERT Base Uncased.
- **Focal Loss Integration**: Focal Loss was used to focus on harder-to-classify samples. This required configuring the labels and outputs to work within the custom loss function during training, helping mitigate the effects of the dataset's class imbalance.

4.2. Methodology

To detect factual inaccuracies in cross-lingual summaries, we utilized four approaches tailored to address the dataset's characteristics:

4.2.1. Logistic Regression:

Logistic Regression is a widely used supervised machine learning algorithm that is used to develop models used for data classification. As it assigns probabilities to each class, it allows for clear decision boundaries. In this study, the Logistic Regression model served as a baseline for text classification. Its simplicity made it an efficient tool for analyzing core patterns and detecting factual errors within the dataset.

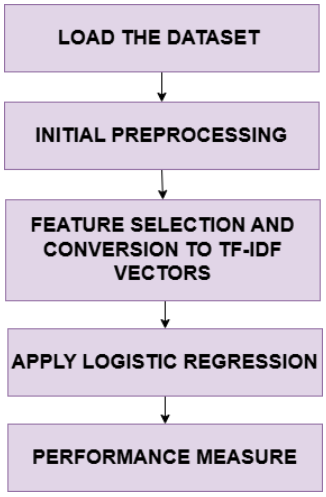

Figure 1: An overview of Logistic Regression for text classification of machine generated summaries in a multilingual context

4.2.2. Logistic Regression with Class Weights:

After evaluating our previous result with Logistic Regression and inferring that the dataset was heavily skewed towards "Correct" labels, we were at a risk of being heavily biased towards the majority class. Keeping that in mind, to address the imbalance observed, we implemented a model utilizing Logistic Regression with class weights to counteract this imbalance. By assigning proportionally higher weights to the minority class and lower weights to the majority class, this approach seeks to balance the influence of each class during the optimization process.

By modifying the parameter `class_weight='balanced'` in the LogisticRegression class, we ensure that the model correctly identifies classes that appear with less frequency in the data.

This approach improved the model's sensitivity to less frequent error categories by assigning higher importance to minority classes.

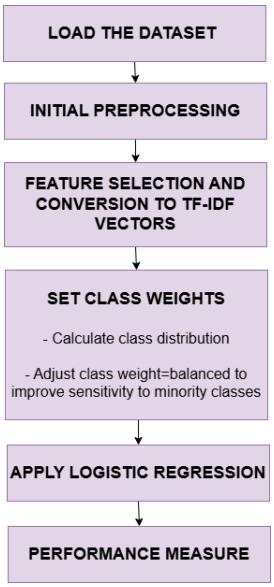

Figure 2: An overview of Logistic Regression with Class Weights for text classification of machine-generated summaries in a multilingual context

4.2.3. BERT Base Uncased:

To capture complex contextual dependencies within summaries, we employed a BERT model fine-tuned for sequence classification. BERT's contextual embeddings are effective for nuanced classifications, making it suitable for identifying subtle factual inconsistencies in the summaries.

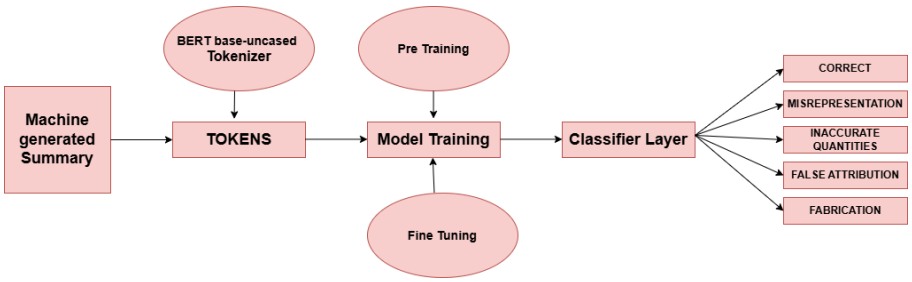

Figure 3: An overview of BERT for text classification of machine generated summaries in a multilingual context

The BERT Base Uncased model is loaded and the training parameters for the model are specified using the `TrainingArguments` class. The key parameters and their respective fine-tuned values are detailed below:

- **output_dir:** This parameter is set to `./results`, defining the directory where the model checkpoints and outputs will be stored during training. This is crucial for tracking the model's performance and for future use.
- **num_train_epochs:** The model is configured to train for 3 epochs, referring to the number of complete passes through the training dataset.
- **per_device_train_batch_size:** The training batch size is specified as 16, indicating that 16 samples will be processed in each training step per device (e.g., GPU or CPU).
- **per_device_eval_batch_size:** The evaluation batch size is set to 64.
- **warmup_steps:** A warm-up period of 500 steps is established, during which the learning rate gradually increases from zero to its initial value.
- **weight_decay:** A weight decay of 0.01 is employed to apply L2 regularization, which aids in preventing overfitting by penalizing large weights during optimization.

- **logging_dir:** This parameter specifies the directory `./logs` for storing logs generated during training.
- **logging_steps:** The logging frequency is set to every 10 steps, allowing for regular monitoring of training progress.
- **evaluation_strategy:** The evaluation strategy is defined as `"epoch,"` indicating that the model will be evaluated at the end of each training epoch.

4.2.4. BERT with Focal Loss:

This model further addressed the class imbalance by incorporating Focal Loss, which focuses more on challenging, misclassified instances. This strategy enhanced the model's performance on minority classes, allowing for more precise detection of factual inaccuracies like Fabrication and False Attribution.

- The optimizer is instantiated using the `AdamW` class, with a learning rate of 2×10^{-5}. This choice of optimizer facilitates efficient training by adapting the learning rate for each parameter, contributing to improved convergence.
- A training loop is established to utilize Focal Loss as the loss function, designed to address class imbalance by focusing more on harder-to-classify samples.
- The `train_epoch` function is defined to perform one epoch of training. Inside this function, the model is set to training mode, and the total loss is initialized to zero.
- The Focal Loss is calculated using the model's logits and the true labels. The loss is then back-propagated, and the optimizer updates the model's parameters. The total loss for the epoch is averaged and returned.
- A separate `eval_model` function is defined to evaluate the model's performance on the validation set. In this function, the model is switched to evaluation mode, and predictions along with true labels are collected without tracking gradients.
- The model is trained for a total of 3 epochs, with the training loss printed at the end of each epoch.

5. Results and Performance Analysis

For evaluating our approaches, we chose the F1 Score as our evaluation metric. The F1 Score is an evaluation metric that balances both precision and recall. In the face of imbalance as seen in our data, it keeps the approach robust and capable to classify data even if one label appears lesser number of times compared to other labels for our output.

For Gujarati, we used Logistic Regression and Logistic Regression with Class Weights approaches respectively. This gave us a baseline idea about how well such models can classify text. The Logistic Regression model yielded a F1 Score of 0.0969, highlighting its performance with respect to skewed data. Taking this into account, we used Logistic Regression with Class Weights which yielded a F1 Score of 0.3371, making this approach more robust and less susceptible towards an imbalance in data.

For Hindi, we used the BERT transformer base uncased model in its original and focal loss enabled form. The transformer model is able to classify to a greater extent due to the fact it also ensures the context of the text is processed. The BERT base Uncased transformer model yielded a F1 Score of 0.2133 This prompted us to use BERT with Focal Loss which yielded our best result, a F1 Score of 0.3426.

Table 2
F1-Score (Gujarati) Results for Different Teams

Rank	Team Name	Language	F1-Score
1	ivSUM	Gujarati	0.3371
2	Squad	Gujarati	0.296
3	Trojan Horses	Gujarati	0.2456

Table 3
F1-Score (Hindi) Results for Different Teams

Rank	Team Name	Language	F1-Score
1	ivSUM	Hindi	0.3426
2	Squad	Hindi	0.3153
3	CUET_SSTM	Hindi	0.2371

6. Conclusion

Through the medium of this shared task and research paper, we have gained a deeper understanding of text classification and the positive effect this can bring towards the users on the internet and expansion of the field of Natural Language Processing. We also learned how to process text in languages like Gujarati and Hindi, furthering Natural Language Processing in regional languages as well. Through our findings, the evaluation metric used validates the robustness of the model in case of imbalanced or skewed data.

BERT with Focal Loss gave the highest F1 Score for Hindi data at 0.3426 and Logistic Regression with Class Weights gave the highest F1 Score at 0.3371.

We conclude by stating that the problem of text classification is crucial and further research and multilingual support for the same is beneficial for a very large group of people on the internet. We can deploy such systems with ease online to help moderate content on different social platforms, identifying different types of misinformation in various languages and flagging them wherever it sees fit. Ensuring scalability of such solutions can ensure flexibility and reliability of such systems in various linguistic contexts, furthering the field of Natural Language Processing.

7. Declaration on Generative AI

During the preparation of this work, the author(s) used GPT-4 in order to: Drafting content, Improve writing style, and Paraphrase and reword concepts. After using these tool(s)/service(s), the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.

References

[1] S. Satapara, B. Modha, S. Modha, P. Mehta, Findings of the first shared task on indian language summarization (ILSUM): approaches challenges and the path ahead, in: K. Ghosh, T. Mandl, P. Majumder, M. Mitra (Eds.), Working Notes of FIRE 2022 - Forum for Information Retrieval Evaluation, Kolkata, India, December 9-13, 2022, volume 3395 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2022, pp. 369–382. URL: https://ceur-ws.org/Vol-3395/T6-1.pdf.

[2] S. Satapara, B. Modha, S. Modha, P. Mehta, FIRE 2022 ILSUM track: Indian language summarization, in: D. Ganguly, S. Gangopadhyay, M. Mitra, P. Majumder (Eds.), Proceedings of the 14th Annual Meeting of the Forum for Information Retrieval Evaluation, FIRE 2022, Kolkata, India, December 9-13, 2022, ACM, 2022, pp. 8–11. URL: https://doi.org/10.1145/3574318.3574328. doi:10.1145/3574318.3574328.

[3] S. Satapara, P. Mehta, S. Modha, D. Ganguly, Key takeaways from the second shared task on indian language summarization (ILSUM 2023), in: K. Ghosh, T. Mandl, P. Majumder, M. Mitra (Eds.), Working Notes of FIRE 2023 - Forum for Information Retrieval Evaluation (FIRE-WN 2023), Goa, India, December 15-18, 2023, volume 3681 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2023, pp. 724–733. URL: https://ceur-ws.org/Vol-3681/T8-1.pdf.

[4] S. Satapara, P. Mehta, S. Modha, D. Ganguly, Indian language summarization at FIRE 2023, in: D. Ganguly, S. Majumdar, B. Mitra, P. Gupta, S. Gangopadhyay, P. Majumder (Eds.), Proceedings of the 15th Annual Meeting of the Forum for Information Retrieval Evaluation, FIRE 2023, Panjim,

India, December 15-18, 2023, ACM, 2023, pp. 27–29. URL: https://doi.org/10.1145/3632754.3634662. doi:10.1145/3632754.3634662.

[5] S. Satapara, P. Mehta, S. Modha, A. Hegde, S. HL, D. Ganguly, Overview of the third shared task on indian language summarization (ilsum 2024), in: K. Ghosh, T. Mandl, P. Majumder, D. Ganguly (Eds.), Working Notes of FIRE 2024 - Forum for Information Retrieval Evaluation, Gandhinagar, India. December 12-15, 2024, CEUR Workshop Proceedings, CEUR-WS.org, 2024.

[6] S. Satapara, P. Mehta, S. Modha, A. Hegde, S. HL, D. Ganguly, Key insights from the third ilsum track at fire 2024, in: Proceedings of the 16th Annual Meeting of the Forum for Information Retrieval Evaluation, FIRE 2024, Gandhiinagar, India. December 12-15, 2024, ACM, 2024.

[7] S. Satapara, P. Mehta, D. Ganguly, S. Modha, Fighting fire with fire: Adversarial prompting to generate a misinformation detection dataset, CoRR abs/2401.04481 (2024). URL: https://doi.org/10.48550/arXiv.2401.04481. doi:10.48550/ARXIV.2401.04481. arXiv:2401.04481.

[8] H. Liu, S. Zhang, X. Wu, Mlslr: Multilabel learning via sparse logistic regression, Information Sciences 281 (2014) 310–320. URL: https://www.sciencedirect.com/science/article/pii/S0020025514005519. doi:https://doi.org/10.1016/j.ins.2014.05.013, multimedia Modeling.

[9] S. Aseervatham, A. Antoniadis, E. Gaussier, M. Burlet, Y. Denneulin, A sparse version of the ridge logistic regression for large-scale text categorization, Pattern Recognition Letters 32 (2011) 101–106. URL: https://www.sciencedirect.com/science/article/pii/S0167865510003272. doi:https://doi.org/10.1016/j.patrec.2010.09.023.

[10] K. Shah, H. Patel, D. Sanghvi, M. Shah, A comparative analysis of logistic regression, random forest and knn models for the text classification, Augmented Human Research 5 (2020) 12.

[11] H.-F. Yu, K. Zhong, I. S. Dhillon, W.-C. Wang, Y. Yang, X-bert: extreme multi-label text classification using bidirectional encoder representations from transformers, in: NeurIPS 2019 Workshop on Science Meets Engineering of Deep Learning, 2019. URL: https://www.amazon.science/publications/x-bert-extreme-multi-label-text-classification-using-bidirectional-encoder-representations-from-transformers.

[12] M. Bhargava, K. Vijayan, O. Anand, G. Raina, Exploration of transfer learning capability of multilingual models for text classification, in: Proceedings of the 2023 5th International Conference on Pattern Recognition and Intelligent Systems, 2023, pp. 45–50.

[13] A. Deroy, S. Maity, S. Ghosh, Prompted zero-shot multi-label classification of factual incorrectness in machine-generated summaries., in: FIRE (Working Notes), 2023, pp. 734–746.