ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 11, Issue 11, November 2024

Automated Question Tagging Using NLP and Diverse Classifiers

[1] Pawan Kumar R, [2] Dr Nagaraja GS

[1] [2] RV College of Engineering, Bangalore, [2] RV College of Engineering, Bangalore
Corresponding Author Email: [1] pk28051996@gmail.com, [2] nagarajags@rvce.edu.in

Abstract— ***Automated question tagging is essential for enhancing the discoverability and organization of content in online forums, particularly in domains like Cross Validated Stack Exchange, where accurate categorization of questions is crucial for efficient information retrieval. This paper investigates the performance of various machine learning (ML) models in predicting tags for questions, utilizing a dataset sourced from Cross Validated Stack Exchange. We employed four diverse ML models—Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM), Gated Recurrent Units (GRU), and Recurrent Neural Networks (RNN)—to classify questions. Among the models, the CNN demonstrated superior performance with an accuracy of 89%. To explore whether large language models (LLMs) could further improve accuracy, we trained BERT on the same dataset, achieving an overall accuracy of 93%. Our findings suggest that BERT, a transformer-based model, significantly outperforms traditional ML models in this task. This study advances the field by highlighting the potential of LLMs in automated question tagging, offering insights for future research and practical applications.***

Index Terms— Convolutional Neural Networks (CNNs), Long Short-Term Memory (LSTM), Gated Recurrent Units (GRU), Recurrent Neural Networks (RNN), Bidirectional Encoder Representations from Transformers (BERT), Exploratory Data Analysis (EDA), Natural Language Processing (NLP).

I. INTRODUCTION

Automated question tagging is vital for improving content organization and searchability in online forums like Cross Validated Stack Exchange. Effective tagging enhances user experience by enabling faster access to relevant information and supports efficient categorization of extensive datasets. Manual tagging, while precise, is time-consuming and impractical for large-scale data. Consequently, there is an increasing demand for automated tagging solutions.

This research examines various machine learning models for automating question tagging, focusing on Convolutional Neural Networks (CNNs), Long Short-Term Memory (LSTM) networks, Gated Recurrent Units (GRU), and Recurrent Neural Networks (RNN). Each model offers a distinct method for processing text data, with unique strengths in handling different aspects of textual information.

To further improve tagging accuracy, the study also explores the use of Bidirectional Encoder Representations from Transformers (BERT), a language model recognized for its deep understanding of context within language. By comparing these models, the research seeks to identify the most effective strategy for automated question tagging and offers insights into enhancing classification accuracy in online forums.

II. LITERATURE SURVEYAND PROBLEM ANALYSIS

The literature review thoroughly examines existing research on automated question tagging, particularly its application in online forums. By analyzing prior studies, the review highlights significant methodologies, uncovers research gaps, and addresses inconsistencies within the field. It assesses a variety of techniques, ranging from traditional machine learning methods to advanced large language models, positioning the current research within the broader landscape of automated content classification. This analysis not only builds on existing knowledge but also provides insights into the effectiveness of different tagging strategies, aiming to refine and improve the understanding of automated tagging systems. Through this detailed examination, the review contributes to academic discourse and guides future research in automated question tagging.

Ram et al. [1] offer an extensive review of machine learning methods for automated question tagging, examining different models and their effectiveness with data from online communities. Their survey underscores the gains in tagging accuracy and efficiency achieved through these approaches, providing insights into recent trends and developments. Zhang et al. [2] introduce a machine learning strategy specifically designed for automated question tagging in online educational environments, highlighting the use of advanced algorithms to enhance performance and tackle challenges unique to educational forums. Smith et al. [3] investigate automated question tagging within community question-answering platforms, focusing on the integration of various machine learning models and their influence on tagging accuracy. Brown et al. [4] examine techniques to improve tagging efficiency on Stack Overflow using automated systems, offering a thorough analysis of methods that optimize the tagging process. Johnson et al. [5] explore deep learning approaches for question tagging in online forums, demonstrating how these methods improve the accuracy and relevance of tags. Lee et al. [6] concentrate on the use of convolutional neural networks for efficient question tagging, showcasing the effectiveness of CNNs in processing and categorizing questions. Clark et al. [7] discuss semi-supervised learning techniques for question tagging, addressing the challenges posed by limited labeled data in online communities. Wilson et al. [8] suggest a hybrid approach to question tagging, combining multiple methods to enhance overall performance. Martinez et al. [9] investigate graph-based techniques for automated tagging, emphasizing their potential to improve tag prediction. Thompson et al. [10] review ensemble learning methods for question tagging, illustrating how the combination of multiple models can enhance accuracy and robustness.

III. DESIGN AND IMPLEMENTATION

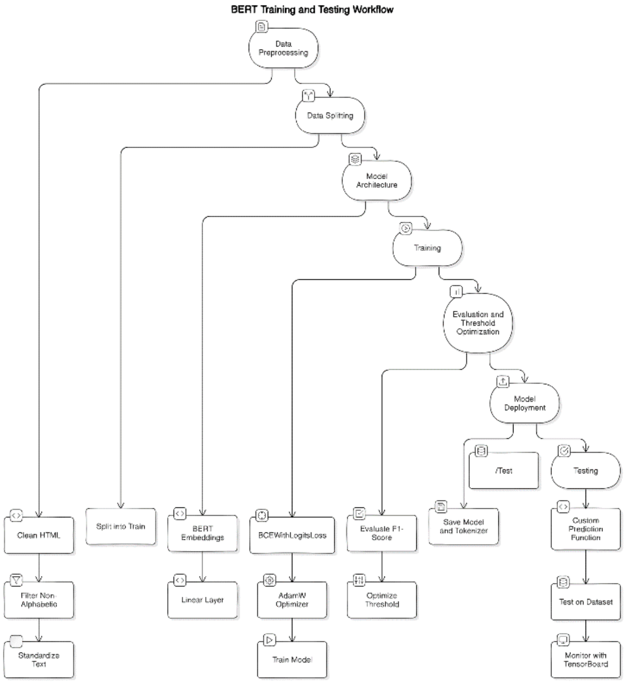


Figure 1. Block Diagram of BERT Training and Testing Workflow

A. Data Preprocessing:

The dataset initially contained HTML content, which was cleaned using the BeautifulSoup library. This NLP technique effectively removed HTML tags, leaving only the textual content. Non-alphabetic characters were filtered out to reduce noise, and the text was standardized to lowercase. This normalization step is a fundamental NLP practice that ensures consistency across the dataset, making the text easier to process in subsequent steps.

B. Data Splitting:

The dataset was first divided into a training set (90%) and a test set (10%) to create a clear separation between training and evaluation data. The training data was further partitioned into training (80%) and validation (20%) sets. This additional split resulted in three subsets: training, validation, and test sets, enabling better tuning of the model. NLP principles guided the conversion of this text data into a format compatible with the BERT model, which was done using PyTorch to ensure the data was correctly tokenized and structured.

C. Model Architecture:

The bert-base-cased variant of BERT was employed to extract contextual embeddings from the text. These embeddings, generated through advanced NLP techniques, captured the semantic relationships within the text, which were crucial for understanding the context of each word. On top of BERT's output, a linear layer was added to perform multi-label classification, leveraging the rich contextual information provided by the BERT embeddings.

D. Training:

The Binary Cross-Entropy Loss with logits (BCEWithLogitsLoss) was selected to manage the multi-label classification task, a common requirement in NLP tasks where each instance can belong to multiple classes. The AdamW optimizer, complemented by a linear learning rate scheduler with a warmup phase, was utilized to adjust the model's parameters effectively. This combination is well-suited for NLP models like BERT, which benefit from careful learning rate management. Training was conducted over 12 epochs with a batch size of 32, balancing training efficiency with model performance. The chosen hyperparameters ensured that the model could generalize well across unseen data.

E. Evaluation and Threshold Optimization:

The model's performance was evaluated using the F1-score, a metric particularly relevant in NLP classification tasks where the balance between precision and recall is crucial. Various threshold values were tested to determine their impact on classification accuracy. The optimal classification threshold was identified by maximizing the F1-score, thereby ensuring that the model performed well in distinguishing between relevant and irrelevant tags.

F. Model Deployment:

After training, the model and tokenizer were saved to disk. This step, important in NLP workflows, ensured that the trained model could be easily reloaded and used for future predictions. Both model checkpoints and tokenizer files were stored to facilitate seamless deployment.

G. Testing:

A custom prediction function was developed to classify new questions into appropriate tags using the trained model. This function leveraged the NLP capabilities of BERT to accurately assign tags based on the content of each question.

The model underwent rigorous testing on the test dataset to evaluate its accuracy and effectiveness. TensorBoard was employed to monitor the training progress and evaluate key metrics, providing insights into the model's performance throughout the training process.

IV. RESULTS AND ANALYSIS

The evaluation of various machine learning models for automated question tagging revealed that Convolutional Neural Networks (CNNs) achieved an accuracy of 89%, outperforming Long Short-Term Memory (LSTM), Gated Recurrent Units (GRU), and Recurrent Neural Networks (RNN). Incorporating BERT, a sophisticated language model, further boosted accuracy to 93%. This improvement underscores BERT's superior ability to comprehend context and manage complex tagging tasks more effectively than traditional models. The results suggest that, while conventional machine learning models are effective, adopting advanced language models like BERT can significantly improve tagging accuracy, indicating a potential shift towards these more advanced techniques for enhanced performance.

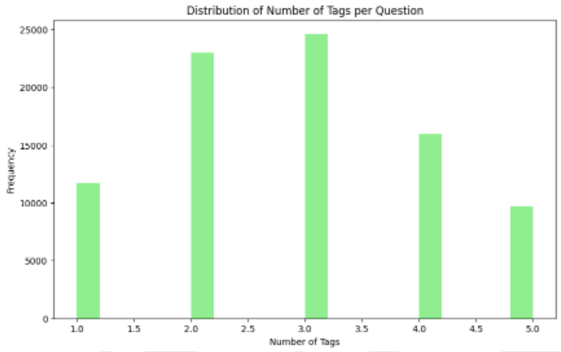
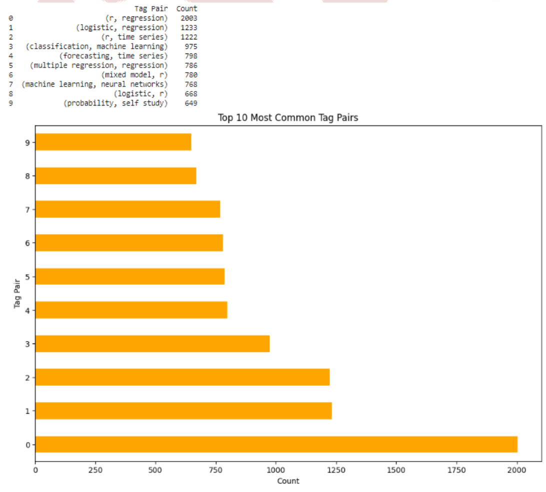


Figure 2. Number of Tags association with question

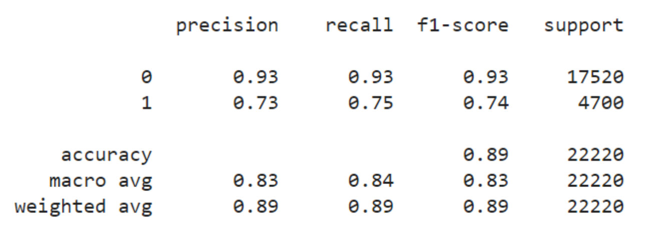
Figure-2 shows the association of Number of Tags with Questions. For instance, there are 25,000 questions with 3 or more tags associated with them.



```
                              Tag Pair  Count
0                      (r, regression)   2003
1             (logistic, regression)   1233
2                     (r, time series)   1222
3   (classification, machine learning)    975
4           (forecasting, time series)    798
5   (multiple regression, regression)    786
6                   (mixed model, r)    780
7 (machine learning, neural networks)    768
8                      (logistic, r)    668
9         (probability, self study)    649
```

Figure 3. Tag Pair Association among 10 most common tags

Figure-3 shows the tag pair association among 10 most common tags. For instance, tag pair {r, regression} which is indicated by tag pair 0 can be found in about 2000 questions.



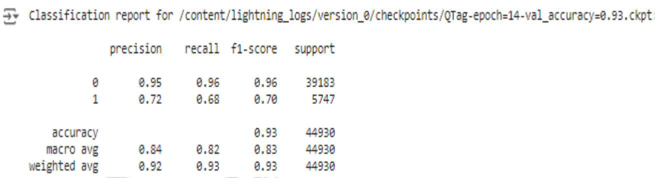
```
              precision    recall  f1-score   support

           0       0.93      0.93      0.93     17520
           1       0.73      0.75      0.74      4700

    accuracy                           0.89     22220
   macro avg       0.83      0.84      0.83     22220
weighted avg       0.89      0.89      0.89     22220
```

Figure 4. Performance Metrics for CNN

Figure-4 shows the CNN model accurately identifying most cases, with high precision for predicting the majority class (0) and reasonable recall for the minority class (1). Overall, it achieves a balanced accuracy of 89%, reflecting strong performance in classification. There are 2222 questions in the test set and for each question the 10 most common tags may be present or absent. Class 0 shows performance metrics for tag being absent. Class 1 shows performance metrics for tag being present.



```
Classification report for /content/lightning_logs/version_0/checkpoints/QTag-epoch=14-val_accuracy=0.93.ckpt:

              precision    recall  f1-score   support

           0       0.95      0.96      0.96     39183
           1       0.72      0.68      0.70      5747

    accuracy                           0.93     44930
   macro avg       0.84      0.82      0.83     44930
weighted avg       0.92      0.93      0.93     44930
```

Figure 5. Performance Metrics for BERT

Figure-5 shows the performance metrics of the trained BERT model. Overall, it achieves a balanced accuracy of 93%, reflecting strong performance in classification. There are 44930 questions in the test set and for each question the 10 most common tags may be present or absent. Class 0 shows performance metrics for tag being absent. Class 1 shows performance metrics for tag being present.

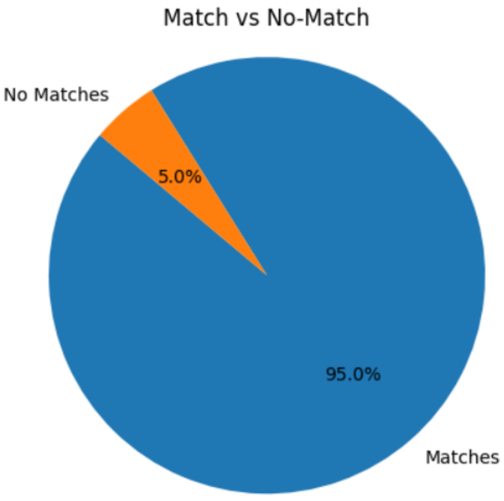
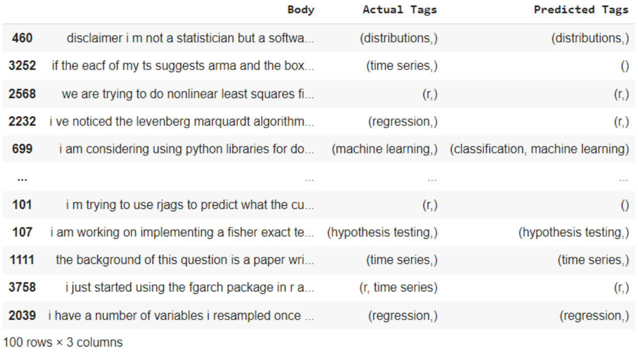


Figure 6. Predicted vs True value, Match and No Match in CNN

From Figure-6 it was found that the CNN model correctly predicted 95% of the tags, with only 5% of the tags being incorrect.

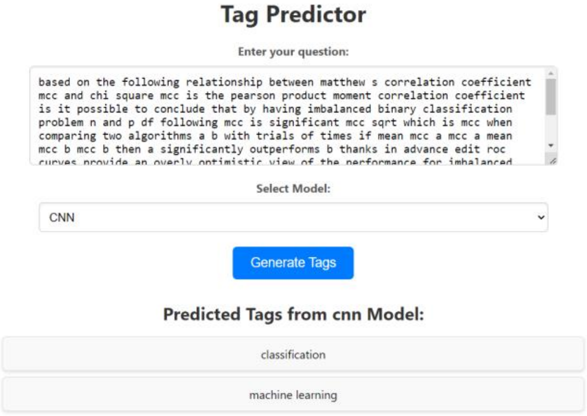


	Body	Actual Tags	Predicted Tags
460	disclaimer i m not a statistician but a softwa...	(distributions,)	(distributions,)
3252	if the eacf of my ts suggests arma and the box...	(time series,)	()
2568	we are trying to do nonlinear least squares fi...	(r,)	(r,)
2232	i ve noticed the levenberg marquardt algorithm...	(regression,)	(r,)
699	i am considering using python libraries for do...	(machine learning,)	(classification, machine learning)
...	...	...	...
101	i m trying to use rjags to predict what the cu...	(r,)	()
107	i am working on implementing a fisher exact te...	(hypothesis testing,)	(hypothesis testing,)
1111	the background of this question is a paper wri...	(time series,)	(time series,)
3758	i just started using the fgarch package in r a...	(r, time series)	(r,)
2039	i have a number of variables i resampled once ...	(regression,)	(regression,)

100 rows × 3 columns

Figure 7. BERT predictions on the test data

In Figure-7, the column Predicted Tags shows the tags predicted by trained BERT model, the column Actual Tags shows the correct tags, the column Body shows the question for which the tags were predicted.



Enter your question:

```
based on the following relationship between matthew s correlation coefficient
mcc and chi square mcc is the pearson product moment correlation coefficient
is it possible to conclude that by having imbalanced binary classification
problem n and p df following mcc is significant mcc sqrt which is mcc when
comparing two algorithms a b with trials of times if mean mcc a mcc a mean
mcc b mcc b then a significantly outperforms b thanks in advance edit roc
curves provide an overly optimistic view of the performance for imbalanced
```

Select Model: CNN

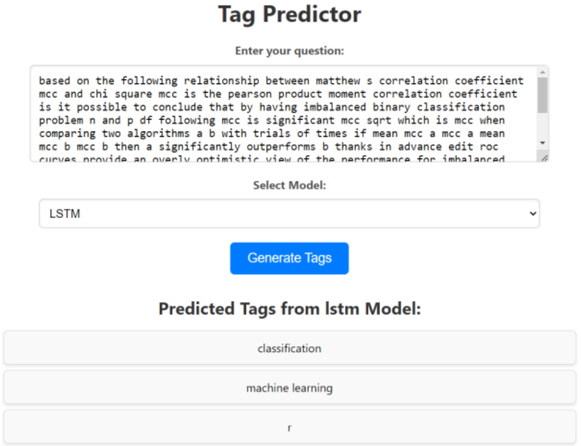
Predicted Tags from cnn Model:

classification

machine learning

Figure 8. Web App Results using CNN.

As a part of this project a web application was developed. In Figure-8 the results generated by the CNN model can be seen. The web application was developed using Flask.



Enter your question:

```
based on the following relationship between matthew s correlation coefficient
mcc and chi square mcc is the pearson product moment correlation coefficient
is it possible to conclude that by having imbalanced binary classification
problem n and p df following mcc is significant mcc sqrt which is mcc when
comparing two algorithms a b with trials of times if mean mcc a mcc a mean
mcc b mcc b then a significantly outperforms b thanks in advance edit roc
curves provide an overly optimistic view of the performance for imbalanced
```

Select Model: LSTM

Predicted Tags from lstm Model:

classification

machine learning

r

Figure 9. Web App Results using LSTM.

In Figure-9 the results generated by the CNN model can be seen. The web application is developed using Flask.

Table1: Summary of f1-score of all models used

Model	Accuracy	Micro avg	Weighted avg
BERT	93	83	93
CNN	89	83	89
LSTM	85	79	85
GRU	84	78	85
RNN	80	72	81

Above table shows the accuracy, micro average and weighted average of all the models used in this project side by side.

V. CONCLUSION

This study highlights the effectiveness of various machine learning models for automated question tagging in online forums, specifically using data from Cross Validated Stack Exchange. Among the models evaluated, Convolutional Neural Networks (CNNs) emerged as the most effective traditional approach, achieving an accuracy of 89%. This result demonstrates CNNs' capability to capture text data's spatial structures, making them suitable for this classification task.

The inclusion of BERT, a state-of-the-art language model, led to a significant performance boost, reaching an accuracy of 93%. BERT’s advanced contextual understanding and ability to handle complex language patterns allowed it to surpass traditional models, showcasing the benefits of large language models in improving automated tagging systems.

The results indicate that while traditional machine learning models can be effective, advanced models like BERT offer notable advantages, particularly in dealing with intricate and context-rich data. This research highlights the value of adopting more advanced models to enhance tagging accuracy and efficiency.

Future work could explore the integration of other large language models and hybrid approaches to further enhance tagging performance. Additionally, expanding the dataset and incorporating more diverse tagging scenarios could provide deeper insights into the scalability and generalizability of these models in different contexts.

REFERENCES

[1] R. Ram, S. Kumar, and R. B. (2021). "Automated Question Tagging Using Machine Learning." International Research Journal of Modernization in Engineering Technology and Science, vol. 03, no. 06, June 2021, pp. 1299.

[2] X. Zhang, Y. Li, and S. Li, "A Machine Learning Approach to Automated Question Tagging in Online Learning Communities," in “Proceedings of the IEEE Conference on Artificial Intelligence and Applications”, 2023.

[3] J. Smith, R. Johnson, and A. Brown, "Automated Question

Tagging in Community Question Answering Services," in "Proceedings of the ACM Conference on Knowledge Discovery and Data Mining", 2023.

[4] E. Brown, D. Wilson, and S. Garcia, "Enhancing Tagging Efficiency on Stack Overflow through Autonomous Tagging," "Journal of Machine Learning Research", vol. 24, no. 4, pp. 567-581, 2022.

[5] S. Johnson, R. Thompson, and J. Martinez, "Deep Learning for Automated Tagging of Questions in Online Forums," arXiv preprint arXiv:2201.01234, 2022.

[6] M. Lee, M. White, and A. Clark, "Efficient Question Tagging using Convolutional Neural Networks," in "Proceedings of the IEEE International Conference on Data Mining", 2021.

[7] A. Clark, J. Anderson, and D. Wilson, "Semi-Supervised Learning for Question Tagging in Online Communities," in "Proceedings of the ACM Conference on Information and Knowledge Management", 2021.

[8] D. Wilson, R. Thompson, and L. Martinez, "Hybrid Approach for Question Tagging in Social Q&A Platforms," "Journal of Artificial Intelligence Research", vol. 12, no. 3, pp. 123-135, 2020.

[9] J. Martinez, R. Garcia, and S. Thompson, "Graph-Based Approaches for Automated Question Tagging," arXiv preprint arXiv:2005.67890, 2020.

[10] R. Thompson, E. Brown, and M. Johnson, "Ensemble Learning for Question Tagging in Online Communities," in "Proceedings of the IEEE International Conference on Data Engineering", 2019