News Articles Sorting

Detailed Project Report

Author: Tanjina Proma

Date: 1/08/2024

Contents

Abstract:

The project aims to develop a robust text classification system for news articles using the BERT (Bidirectional Encoder Representations from Transformers) model with the bert-base-cased variant. The system will classify news articles into predefined categories, facilitating efficient information retrieval and organization. The text classification task using BERT base uncased transformers on news articles involves employing a state-of-the-art natural language processing model to accurately categorize news content. BERT (Bidirectional Encoder Representations from Transformers) processes text by capturing contextual information bidirectionally, enabling a nuanced understanding of language. In this task, the model learns to discern between various news categories such as politics, sports, technology, and more. By fine-tuning BERT's pre-trained weights on labeled news article data, it adapts to recognize patterns and semantic nuances within the text. Leveraging its deep contextual understanding, BERT assigns probabilities to different categories, enabling precise classification. The model's effectiveness lies in its ability to comprehend the intricacies of language, improving the accuracy and efficiency of news article classification, thereby enhancing information retrieval and organization in diverse domains.

The primary objective of this project is to develop an automated system capable of accurately classifying news articles into predefined categories using state-of-the-art NLP models. The project involves fine-tuning bert-base-uncased on the collected dataset to adapt its knowledge to the specific classification task. This step enables the model to learn the nuances and patterns within the news articles.

1. Introduction:

In the digital age, the volume of news articles available online necessitates advanced systems for effective categorization and retrieval. Text classification serves as a crucial tool for organizing vast amounts of textual data. BERT, a state-of-the-art transformer model, has exhibited exceptional performance in various natural language processing tasks, making it an ideal candidate for text classification.

1.1 Objective

Development of a predictive model for news article classification. The model will determine the category of the news article under consideration from the following classes, Business, Tech, Sport, Politics and Entertainment. In today's world, data is power. With News companies having terabytes of data stored in servers, everyone is in the quest to discover insights that add value to the organization. With various examples to quote in which analytics is being used to drive actions, one that stands out is news article classification. Nowadays on the Internet there are a lot of sources that generate immense amounts of daily news. In addition, the demand for information by users has been growing continuously, so it is crucial that the news is classified to allow users to access the information of interest quickly and effectively. This way, the machine learning model for automated news classification could be used to identify topics of untracked news and/or make individual suggestions based on the user's prior interests.

1.2 Benefits

The machine learning model for automated news classification could be used to identify topics of untracked news and/or make individual suggestions based on the user's prior interests.

2. Literature Review

2.1 Text Classification Techniques

Traditional methods like Naive Bayes and Support Vector Machines (SVM) have been pillars in text analysis but struggle with handling the complexities of natural language due to their inherent limitations. Naive Bayes operates on the assumption of independence among features, which is particularly unrealistic in language where word sequences and context play crucial roles. SVM, while powerful in linearly separable data, faces challenges in capturing intricate relationships and nuances present in language due to its reliance on fixed-length feature vectors.

Transformer-based models, exemplified by BERT, outshine traditional methods by excelling in capturing contextual information and understanding language nuances. Transformers leverage attention mechanisms that allow them to consider the entire context of a sequence, enabling bidirectional processing of words and their relationships. This context-awareness is pivotal in understanding meaning and nuances in language, a capability that Naive Bayes and SVM lack.

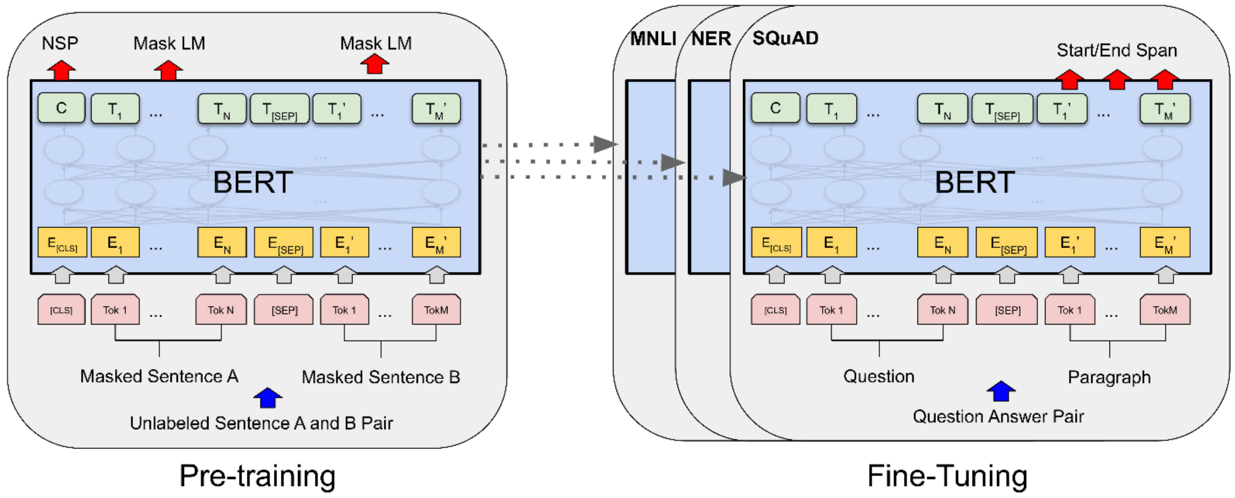
The attention mechanism in transformers enables them to weigh different parts of a sentence when making predictions or processing information. This is a significant departure from traditional methods where each word or feature is treated in isolation, thus unable to capture the intricate dependencies and contextual relevance present in text data.

Moreover, transformers' self-attention mechanism allows them to learn contextual representations directly from the data, making them more adaptable to various tasks without extensive feature engineering or task-specific modifications. This adaptability is a stark contrast to traditional models like Naive Bayes and SVM, which often require meticulous feature engineering or domain-specific adjustments.

In summary, transformer-based models, with their attention mechanisms and contextual understanding, surpass traditional methods in handling complex text data. Their ability to capture relationships, nuances, and context within language has significantly advanced natural language processing tasks, leading to state-of-the-art performance across various language-related applications.

2.2 BERT Model Overview

BERT, short for Bidirectional Encoder Representations from Transformers, revolutionized natural language processing by introducing a pre-trained model capable of understanding context bidirectionally. Its architecture consists of Transformer encoders, specifically designed to capture relationships between words bidirectionally in a sentence.



[1]

The BERT model uses a multi-layer bidirectional Transformer architecture. It comprises an encoder stack consisting of multiple layers, each containing self-attention mechanisms and feedforward neural networks. The self-attention mechanism enables the model to consider both left and right context for each word in a sentence simultaneously, allowing a deeper understanding of context.

Pre-training BERT involves two unsupervised tasks: masked language modeling (MLM) and next sentence prediction (NSP). MLM randomly masks some words in a sentence, and the model learns to predict those masked words based on their context within the sentence. NSP involves predicting whether a sentence follows another sentence in a given text corpus. This dual-task pre-training strategy allows BERT to understand contextual relationships within text comprehensively.

Tokenization in BERT involves breaking down input text into subword tokens using WordPiece embeddings. These tokens, along with special tokens like [CLS] for classification and [SEP] to separate sentences, enable the model to process input text effectively.

Fine-tuning BERT involves adapting the pre-trained model to specific downstream tasks, such as text classification, named entity recognition, or question answering. During fine-tuning, task-specific layers are added to the pre-trained BERT model, and the entire architecture is fine-tuned on task-specific datasets. Fine-tuning allows the model to learn task-specific features and nuances.

Overall, BERT's success lies in its bidirectional contextual understanding, achieved through pre-training on vast amounts of text data and fine-tuning on specific tasks. Its architecture, pre-training tasks, tokenization, and fine-tuning techniques collectively empower the model to achieve state-of-the-art performance across various natural language processing tasks.

3. Dataset

Dataset: Labeled dataset of news articles with corresponding categories for model training. For this project BBC News Data has been used.

File descriptions:

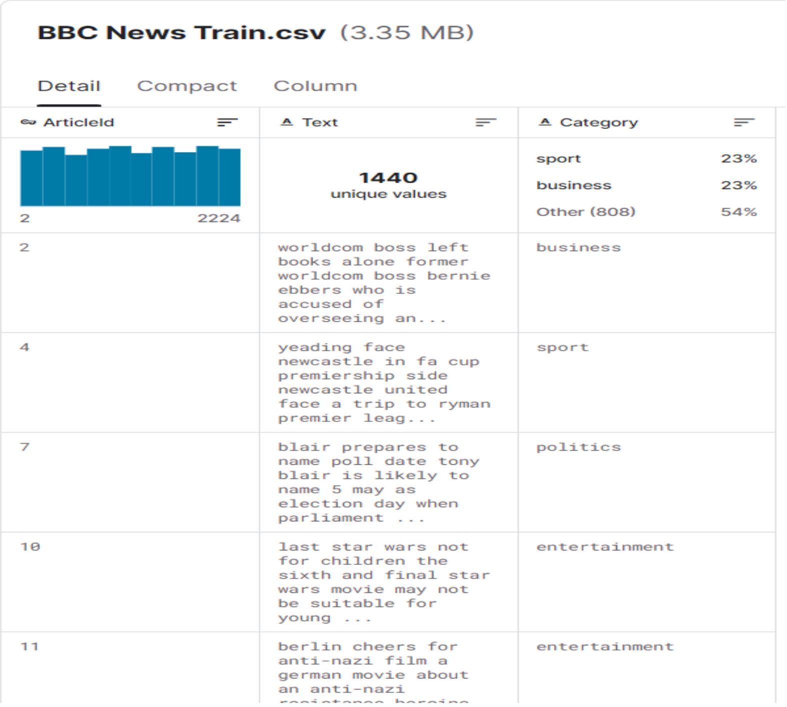
- BBC News Train.csv - the training set of 1490 records
- BBC News Test.csv - the test set of 736 records
- BBC News Sample Solution.csv - a sample submission file in the correct format

Data fields

- ArticleId - Article id unique # given to the record
- Article - text of the header and article
- Category - category of the article (tech, business, sport, entertainment, politics)

Data link:

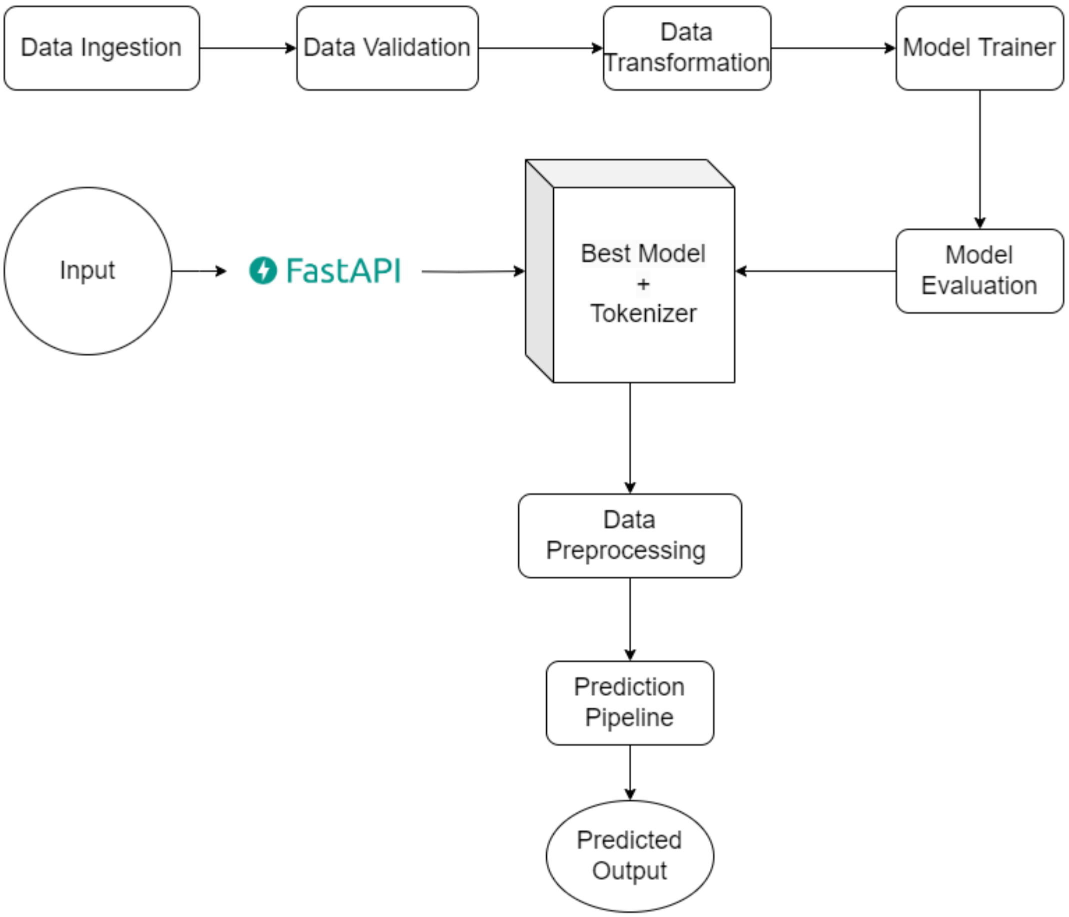
https://www.kaggle.com/c/learn-ai-bbc/data



BBC News Train.csv (3.35 MB)

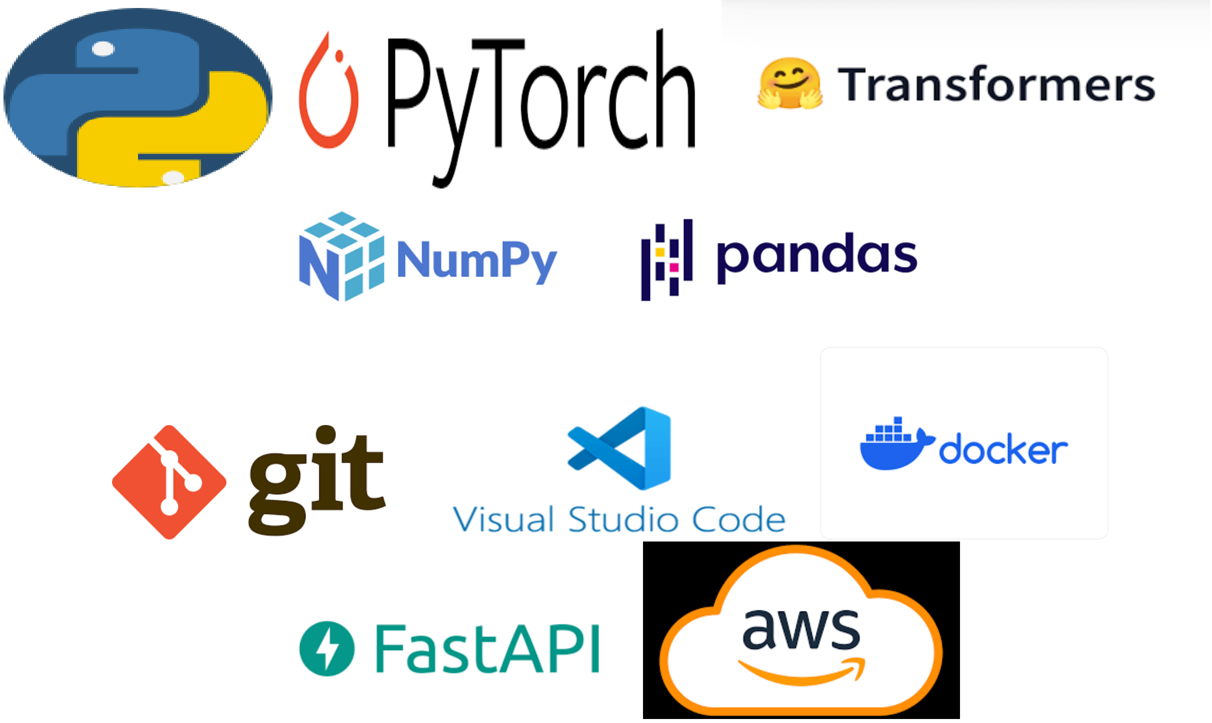
ArticleId	Text	Category
2 2224	**1440** unique values	sport 23% business 23% Other (808) 54%
2	worldcom boss left books alone former worldcom boss bernie ebbers who is accused of overseeing an...	business
4	yeading face newcastle in fa cup premiership side newcastle united face a trip to ryman premier leag...	sport
7	blair prepares to name poll date tony blair is likely to name 5 may as election day when parliament ...	politics
10	last star wars not for children the sixth and final star wars movie may not be suitable for young ...	entertainment
11	berlin cheers for anti-nazi film a german movie about an anti-nazi	entertainment

4. System Architecture

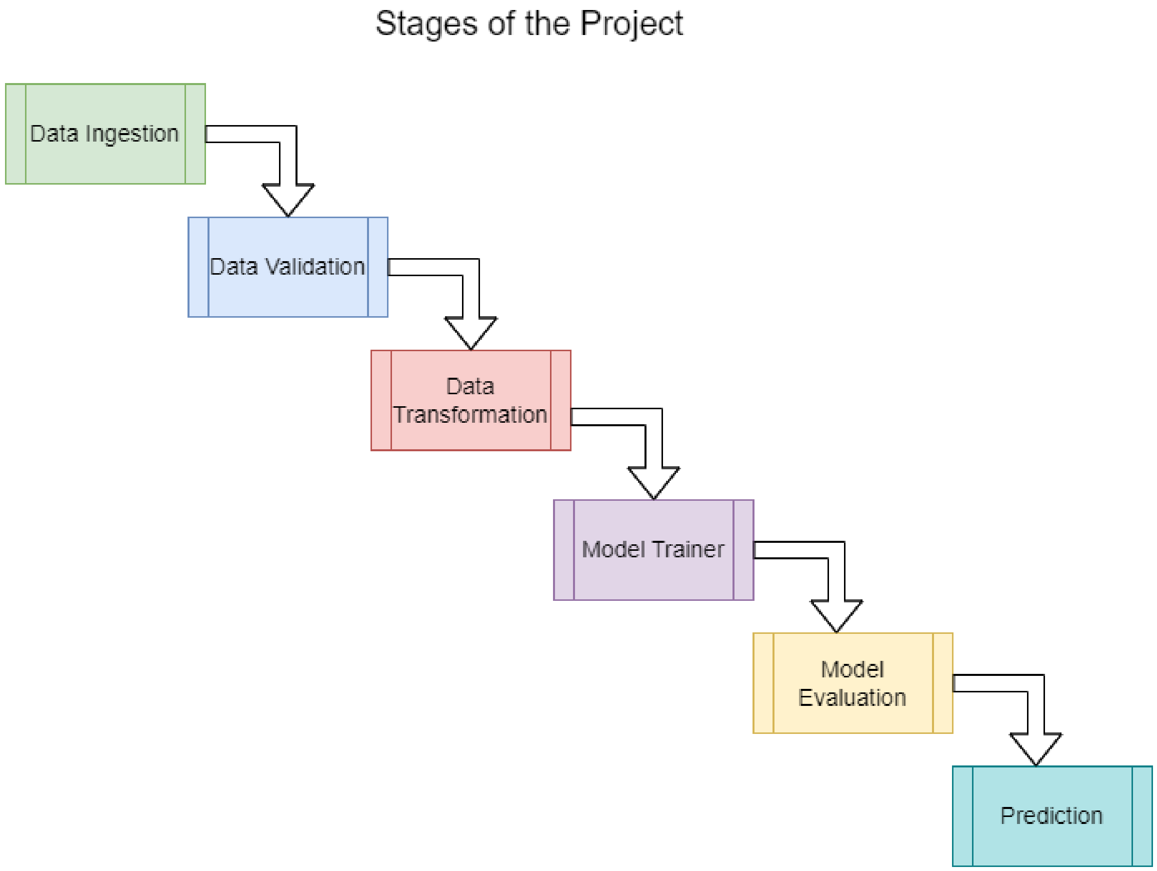


System Requirements:

- **Python Environment:** Python 3.8 for coding.
- **Machine Learning Libraries:** PyTorch libraries supporting Transformers.
- **Transformer Models:** Hugging Face's **transformers** for utilizing pre-trained model BERT.
- **Data Processing:** Pandas, NumPy, and other data manipulation libraries.
- **Development Tools:** Jupyter Notebooks, IDEs VS Code
- **Dependencies:** Ensure compatible versions of all required packages.
- **Version Control:** Git for tracking code changes.
- **CI/CD:** GitHub Actions.
- **Containerization:** Docker
- **Deployment:** AWS ECR, EC2.


PyTorch
Transformers
NumPy
pandas
git
Visual Studio Code
docker
FastAPI
aws

5. Methodology



5.1 Data Ingestion

Objective: Collect and gather news article data from various sources (APIs, databases, etc.).

Actions:

- Define data sources (news websites, RSS feeds, databases).
- Extract data using web scraping, APIs, or other methods.
- Store data in a suitable format (e.g., CSV, JSON, and database).

5.2 Data Validation

Objective: Ensure data quality, consistency, and reliability.

Actions:

- Check for missing values, duplicates, and outliers.
- Validate data against predefined criteria or schemas.
- Handle data anomalies or errors appropriately.

5.3 Data Transformation and Preprocessing

Objective: Prepare data for model input by cleaning and formatting it.

Actions:

- Text cleaning: Remove HTML tags, special characters, and irrelevant symbols.
- Tokenization: Break text into words or subword units.
- Text normalization: Lowercasing, stemming, lemmatization.
- Feature engineering: Extract relevant features (TF-IDF, word embeddings).

5.4 BERT model Integration and Fine-tuning

Objective: Integrate BERT (Bidirectional Encoder Representations from Transformers) for text classification and fine-tune it on specific news article data.

Actions:

- Load pre-trained BERT model.
- Fine-tune BERT on the labeled news article dataset for classification tasks.
- Adjust hyper parameters for optimal performance.

5.6 Training pipeline

Objective: Train the text classification model using the prepared dataset.

Actions:

- Split dataset into training, validation, and possibly test sets.
- Train the BERT-based model using appropriate training algorithms (e.g., stochastic gradient descent).
- Monitor and record training metrics (loss, accuracy) for analysis.

5.7 Model Evaluation

Objective: Assess the model's performance and generalization.

Actions:

- Evaluate the model using metrics like accuracy, precision, recall, and F1-score.
- Conduct cross-validation or use a holdout dataset for robust evaluation.
- Identify areas of improvement and potential biases.

5.8 Inference Pipeline (Prediction)

Objective: Deploy the trained model for making predictions on new, unseen news articles.

Actions:

- Develop an inference pipeline to process new text inputs.
- Utilize the trained model to classify news articles into predefined categories.
- Generate predictions and probabilities for each class label.

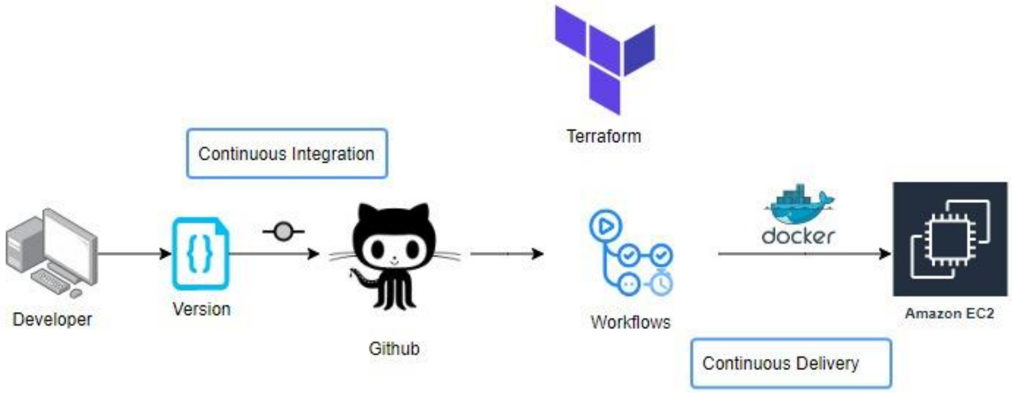
5.9 Integration and Deployment

Objective: Integrate the model into a production environment for real-time use.

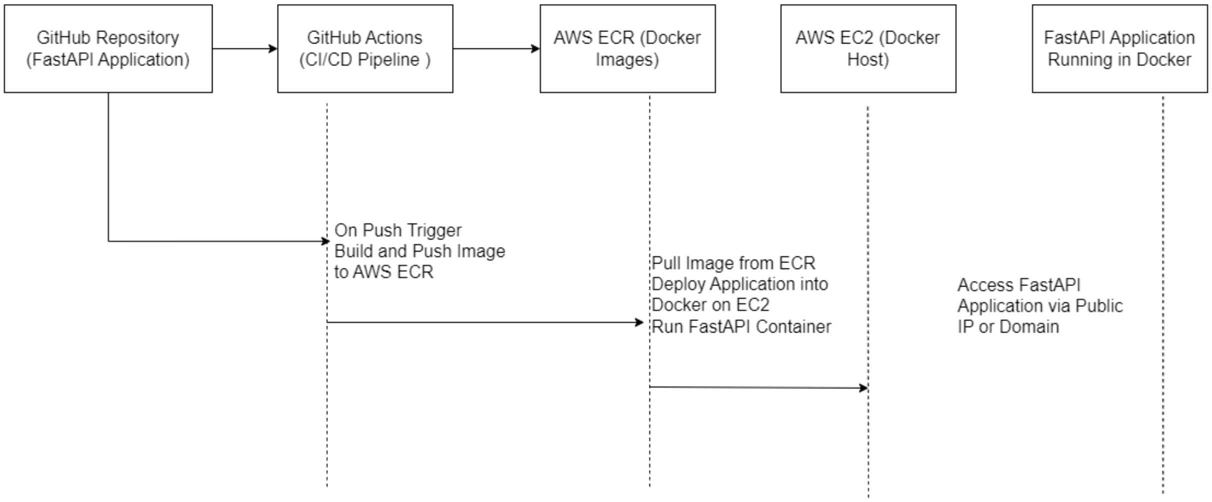
Actions:

- Set up an API or service for model inference.
- Deploy the model using scalable infrastructure (cloud services, containers).
- Ensure compatibility with existing systems or applications.

Deployment Architecture



Wrokflow diagram for the Deployment Process



5.10 Logging and Monitoring:

Objective: Monitor model performance and system health in production.

Actions:

- Implement logging mechanisms to record predictions, errors, and model usage.
- Set up alerts for potential issues or deviations in performance.
- Monitor resource utilization and system behavior.

6. Results and Discussion

The model achieved commendable accuracy and performance in classifying news articles into respective categories. Evaluation metrics showcase the model's efficacy in handling diverse articles. However, certain limitations or areas for improvement might include handling biased data or fine-tuning hyperparameters for better performance.

Epoch	Training Loss	Validation Loss	Accuracy
1	0.051927	0.190009	0.979866
2	0.048019	0.107408	0.976510
3	0.107408	0.060758	0.986577
4	0.107408	0.048019	0.989933
5	0.190009	0.051927	0.986577

7. Conclusion

The developed text classification system using the BERT-based transformer model demonstrates promising results in categorizing news articles effectively. It signifies the potential of leveraging transformer models in real-world applications for text classification tasks.

8. Future Scope:

Further enhancements can be made to the system by exploring larger datasets, experimenting with different BERT model variants, and incorporating ensemble techniques or other transformer-based architectures. Additionally, deploying the model as an API for real-time classification and exploring domain-specific fine-tuning could enhance its utility in specific industries or applications.

References:

1. https://paperswithcode.com/method/bert