Ryane Li
COE379L
Joe Stubbs, Anagha Jamthe
1 December 2025

Project 3 Proposal

Cross-Model Comparison for News Topic Classification: Advanced Classical Algorithms vs. Fine-Tuned Transformers.

1. Introduction

Automated text classification is now a fundamental task in many industries, especially media and news aggregation, due to the proliferation of digital content. The state-of-the-art in Natural Language Processing (NLP) is currently represented by sophisticated Large Language Models (LLMs) and transformer architectures (such as BERT and RoBERTa), but sophisticated classical machine learning algorithms like XGBoost and Support Vector Machines (SVMs) continue to be practical, computationally affordable alternatives.

The objective of this project is to perform a thorough, direct comparison on a standardized, multi-class text classification task between these two different methodological paradigms: feature-engineered classical models and refined deep learning models.

1.1 Problem Statement

The main goal is to determine the best modeling approach for classifying news topics by striking a balance between two important aspects: computational efficiency and predictive performance. To address the following, we will carefully contrast refined transformer models (using contextual embeddings) with optimized versions of sophisticated classical algorithms (using static feature representations like TF-IDF):

1. How significant is the performance gap between classical, feature-engineered models and fine-tuned transformer models on a modern news classification task?
2. What are the trade-offs in training and inference time for each methodology?
3. Can a resource-efficient classical model provide sufficient performance to justify avoiding the higher computational costs associated with fine-tuning a large transformer?

2. AG News Classification Dataset

- **Source:** Hugging Face Datasets Hub (`ag_news`).
- **Description:** This dataset consists of over 120,000 training samples and 7,600 test samples. Each sample includes the news article title and a short description (abstract). The dataset is pre-split into training and test sets.
- **Task/Classes:** The task is a multi-class classification into four distinct, balanced categories:
    - **World**
    - **Sports**
    - **Business**

    - **Sci/Tech**
- **Data Preparation:** The project will combine the title and description fields into a single text input for classification consistency across all models. The balanced nature of the dataset simplifies initial evaluation but will still require the use of macro-averaged metrics.

3. Products to be Delivered

The deliverables will consist of a comprehensive set of computational artifacts and an analytical report demonstrating the findings and comparative performance.

1. **Project Code Repository (Jupyter Notebooks):**
    - A notebook covering **Data Preprocessing and Exploratory Data Analysis (EDA)**.
    - A notebook dedicated to **Classical Model Implementation and Optimization** (XGBoost, SVM, TF-IDF).
    - A notebook dedicated to **Transformer Model Fine-Tuning and Evaluation** (RoBERTa).
2. **Final Technical Report:** A concise two-page (excluding figures/appendices) analytical report presenting the core project findings. This report will include:
    - **Detailed Methodology:** Steps taken for feature engineering, model selection, and optimization for all tested models.
    - **Quantitative Results Table:** A table comparing all final models based on the following key metrics:
        - **Performance Metrics:** Accuracy, Macro-Averaged F1-Score, and Log Loss.
        - **Efficiency Metrics:** Training time and Inference latency (per 1,000 samples).
    - **Discussion and Conclusion:** An analysis of the performance vs. efficiency trade-off, concluding which model is the most suitable choice for real-world news classification given different resource constraints.
3. **Visualization of Results (Appendix/Figures):** Inclusion of key visualizations, such as:
    - A **Confusion Matrix** for the best-performing classical and best-performing transformer model.
    - A **Bar Chart** visualizing the performance metrics (F1-score) of all finalized models.

Figure 1. High-Level Methods, Techniques, and Technologies

Category	Method/Technique	Implementation Details
Feature Extraction	**Term Frequency-Inverse Document Frequency (TF-IDF)**	Generate high-dimensional sparse vector representations (using unigrams and bigrams) for the entire corpus.
Model 1: Ensemble	**XGBoost (eXtreme Gradient Boosting)**	Use the `XGBClassifier` integrated with `scikit-learn` pipelines. Optimization will include searching for the optimal number of estimators, learning rate, and tree depth via cross-validation.
Model 2: Margin Classifier	**Support Vector Machines (SVM)**	Implement `LinearSVC` and potentially `SVC` with a Radial Basis Function (RBF) kernel, using the optimized TF-IDF feature space.
Optimization	**Grid Search/Randomized Search (Cross-Validation)**	Used for robust hyperparameter tuning of both XGBoost and SVM models.

Figure 2. Transformer (Deep Learning) Pipeline

Category	Method/Technique	Implementation Details
Base Model	**RoBERTa-base**	Chosen over BERT for its robust pre-training approach (dynamic masking, larger batch sizes, no Next Sentence Prediction task), which typically yields higher performance on downstream tasks like classification.
Training Method	**Fine-Tuning**	The pre-trained RoBERTa model will be loaded from the Hugging Face Transformers library, a dense classification layer will be added, and the entire network will be trained end-to-end on the AG News dataset.
Technology	**PyTorch/TensorFlow and Hugging Face Ecosystem**	Utilized for efficient model loading, tokenization, training loop management, and GPU acceleration.
Tokenization/Embedding	**Contextual Embeddings**	Input text is processed using the RoBERTa tokenizer, generating dynamic vector representations (embeddings) that capture word meaning based on surrounding context.