Classifying News Content From the Huffington Post

Aarad Ashraf, Marilyn Cleveland, Christopher Fornesa

December 7, 2025

B.1: Introduction and Problem Definition

B.1.1 Problem Description and Motivation

Predicting the correct category for an article can be difficult, but we believed that neural networks could use text from an article's headline and short description to properly categorize it into one of 30 categories. This is necessary for purposes such as ensuring that a news feed only serves relevant topics given a user's preferences (as a recommendation algorithm) or to help a new writer at an online newspaper determine the best fitting category for an article that they have just written. This is also extensible to social media platforms, such as YouTube where headlines would be titles while short descriptions would still be relevant.

B.1.2 Project Goals in Both Business and Data Science Terms

The data science goal was to use the HuffPost dataset as an input for a multiclass classification model (using the RoBERTa pretrained text classifier) to accurately label news articles into their appropriate categories. The specific inputs would be sentence embeddings using combined text (column "text") composed of concatenated text strings from the "headline" and "short description" columns. The success criteria for this project were a testing accuracy of 70% and a macro F1-score of above 0.70 to balance high performance with per-class accuracy (measured by per-class F1 score) for new media metadata.

In a business context, this model would serve as the basis for a recommendation algorithm served by a media platform. For example, a functional model could provide insights into trending news topics. Our chosen neural network algorithm supports ensuring that viewers of an online media publication (which may publish written, video, or other content) are recommended the correct piece of media according to their preferences. This, in turn, benefits both content creators (on platforms like YouTube) or authors (on online news media platforms),

who can be assured that their content is tagged more accurately, and viewers, who can be assured that content on their personalized feeds are relevant to their interests.

B.1.3 Dataset Description

The HuffPost dataset is composed of data from the Huffington Post, a news media publication, collected between 2012 and 2018 and stored on Hugging Face (Misra, 2018). Each record contained six features: category, headline, authors, link, short description, and date, where the headline and short description columns were consolidated into a single "text" column during training. Two columns contained missing values, authors (36,620 null values) and short description (19,712 null values), and duplicate entries were identified in both headline (2,226) and short description (23,415) fields, later addressed during preprocessing.

Key input modalities involved feature engineering, which introduced the combined "text" field, created by concatenating the headline and short description, alongside each piece of text's news category label. Only text and category were retained for modeling, as it was determined the other columns were not meaningful for this project's goal. A temporary "word count" feature was also generated during preprocessing and later removed. The final cleaned dataset retained five features: category, authors, link, date, and text, though only text and category were used for model training. This resulted in a modeling dataset of 178,353 observations across 30 consolidated categories.

The target variable, category, originally contained 41 classes (see Figure 1 in the Appendix). The most frequent categories were "POLITICS," "WELLNESS," and "ENTERTAINMENT," each with over 10,000 samples, while the least frequent included "EDUCATION," "CULTURE & ARTS," and "LATINO VOICES," with counts near 1,000–2,000. This produced a highly imbalanced class distribution with a noticeable drop in

frequency beyond the top categories (see Table 1 in the Appendix). This imbalance introduces a substantial risk of classification bias, as the model may preferentially predict high-frequency categories while underperforming on low-frequency classes. In addition, several categories exhibit semantic overlap, where articles share similar content despite belonging to different labels, which further increases the difficulty of reliable classification and raises the potential for systematic mislabeling.

B.2: Methodology

B.2.1 Preprocessing Pipeline

Preprocessing began by retaining only the first occurrence of duplicate entries in the short description field and removing additional duplicates. The cleaned headline and short description fields were then concatenated using a space delimiter to form a single text feature called "text." This resolved missing values, reduced near-duplicate ambiguity, and produced a single unified textual input for modeling. The resulting text was lowercased and stripped of excess whitespace. Basic text statistics on the cleaned data indicated a mean word count of 31 per entry and a 95th percentile sequence length of 56 tokens, with longer outliers truncated during vectorization (as evidenced by Figure 2 in the Appendix). Redundancy in the target feature, "category," was addressed through class consolidation based on shared semantic meaning. For example, "GREEN" was merged into "ENVIRONMENT," and "HEALTHY LIVING" was merged into "WELLNESS," reducing the total number of classes from 41 to 30. This improved label separability and produced a smoother class frequency distribution (Figure 3, Appendix). After preprocessing, the dataset contained 178,353 observations across 30 categories and five total columns: category, authors, link, date, and text.

To prepare for model experimentation across several various models, the category labels were converted to integers using label encoding. Tokenization and text vectorization were implemented implicitly using RoBERTa, with a maximum sequence length of 96 (different from the original figure of 56 based on the 95th percentile) and a vocabulary size of 235,259 unique tokens. This layer automatically handled tokenization, truncation, and padding and was integrated into a tf.data pipeline. The vectorized text sequences formed the feature matrix X, while the encoded category labels formed the target vector y. RoBERTa uses byte-pair encoding to split rare words into subword units rather than replacing them with a single placeholder token, allowing the model to learn meaningful representations for infrequent terms and improving generalization to new text. In contrast, BERT uses static masking, where identical masking patterns are applied across all epochs during training. This increases the risk of overfitting to masked tokens rather than generalizing across contexts (Massed Compute, 2024). It should be noted that while a maximum sequence length of 56 was used during the early testing phase, the final model had a maximum length of 96. Finally, reproducible dataset splits were generated using a fixed random seed of 42 and stratified sampling based on the category labels. The data was divided into training (70%), validation (15%), and test (15%) sets to preserve class proportions and support unbiased performance evaluation.

B.2.2 Model Architecture and Justification

The final model selected for this project was the RoBERTa-base pretrained transformer encoder with all layers frozen during training. RoBERTa was chosen based on empirical performance, as it consistently outperformed both BERT and DistilBERT across validation accuracy, weighted F1 score, and macro F1 score during model comparison experiments. Final evaluation of all candidate models was conducted using validation loss, validation accuracy, test

accuracy, weighted F1, macro F1, and per-class F1 scores. RoBERTa improves upon BERT through several architectural and training modifications, including removal of the next-sentence prediction objective, use of dynamic masking, and pretraining on a much larger text corpus with optimized batch sizes and learning rates. These design improvements enable RoBERTa to learn stronger contextual representations of short and semantically dense text. This capability is particularly well suited for the HuffPost dataset, which consists of short headlines and descriptions that may share similar tokens while conveying different meanings across categories. This model's enhanced contextual learning allowed it to better distinguish between semantically overlapping classes, which is reflected in its superior predictive performance relative to the previously evaluated pretrained architectures. The complexity involved also better suits the HuffPost dataset compared to the simpler baseline and custom architectures experimented upon in Milestone 2. Overall, RoBERTa's extensive pretraining base, optimized loss functions, and ease of use allowed it to yield better results in comparison to prior experiments for Milestone 2, as justified in the next section on training strategy.

B.2.3 Training Strategy

The model was trained using the cross-entropy loss function, which computes loss through the negative log-likelihood of the predicted class probabilities to ensure numerical stability during optimization (Romero, 2024). This approach avoids issues associated with direct softmax outputs that can lead to unstable gradient updates. Hyperparameter experimentation identified the most effective configuration as a learning rate of 1e-5, a batch size of 64, and a maximum sequence length of 96. Although this maximum length exceeds the 95th percentile threshold of 56 tokens identified during preprocessing, the expanded length improved the model's ability to capture semantic information in longer text descriptions without overfitting

extreme outliers. Training was limited to three epochs, as earlier experiments showed that model performance began to degrade due to overfitting beyond this point. No dropout, weight decay, or L2 regularization were applied because the model relied on a frozen pretrained encoder without custom trainable layers. Maintaining frozen RoBERTa layers throughout training further improved stability and reduced the need for extensive fine-tuning. The Adam optimizer was used to minimize loss due to its adaptive learning rate updates, stability, and efficiency with sparse gradients (Wei, 2024). This optimizer contributed to rapid convergence within three epochs and required minimal additional hyperparameter tuning. The sole callback employed was early stopping, which monitored validation loss with a patience of three epochs. While early stopping did not activate during final training due to the restricted number of epochs, it was included as a safeguard against potential overfitting.

B.2.4 Performance Validation

Model evaluation used a stratified 70/15/15 split of the dataset into training, validation, and test sets to preserve class distributions across all partitions. A fixed random seed of 42 ensured reproducibility and stratifying by the labels ensured that all sets would have the same frequency of category labels. The validation set (15%) was used during training to monitor convergence and guide hyperparameter selection. Performance was evaluated using overall accuracy, weighted F1 score, macro F1 score, and validation loss, enabling balanced assessment of both majority and minority classes while preventing data leakage. The final evaluation was performed exclusively on the held-out test set (15%) using the selected hyperparameters. Using the test set during final evaluation meant that data leakage was explicitly avoided by ensuring its separation from evaluation during the initial training and validation sets. Test performance was assessed using accuracy, weighted F1 score, macro F1 score, and per-class F1 scores, providing

both global and category-level insight. In this scenario, accuracy measured overall classification correctness, weighted F1 score accounted for class imbalance, macro F1 score emphasized performance on lower-frequency classes, and per-class F1 scores assessed prediction quality at the individual category level.

B.3: Results and Evaluation

B.3.1 Quantitative Results

This model yielded a training accuracy of 0.8004, a minimum validation accuracy of 0.7705, and a testing accuracy of 0.7664. It also yielded a training weighted F1 score of 0.8548, a minimum validation weighted F1 score of 0.7656, and a testing weighted F1 score of 0.7615. Finally, it yielded a training macro F1-score of 0.8008, a minimum validation loss macro F1 score of 0.6818, and a testing macro F1 score of 0.6782. The training and validation loss curves are plotted in Figure 4, and show that, as the training accuracy and loss improve, the validation accuracy and loss begin to plateau, peaking at the second epoch. Meanwhile, the confusion matrix for the test predictions can be found in Figure 5, which showed how accurately the model performed on each class, for instance impact and women articles faced the most inaccurate predictions, compared to stories about food and drink or politics, which had higher frequencies. This is further evidenced by the class frequency table for the predictions found in Table 2 and show a general link between higher frequency categories and higher per-class F1 scores, with notable exceptions.

B.3.2 Interpreting Results

The model had an easier time classifying the categories with higher frequencies of observation and a harder time of classifying the categories with lower frequencies of observations. The confusion matrix, Figure 5 in the appendix, revealed a more in depth

relationship between the number of class observations, the amount of predictions, and the ease of classification as based on per-category F1 score. Highly observed (over 10,000 observations) classes such as politics, wellness, and entertainment had over 2,000 predictions and the highest per-class F1 scores. Meanwhile, the lower observed classes (3,000 or less) such as good news, impact, fifty, and women had less than 500 predictions and represented the lowest per-class F1 scores. A summary of these values can be found in Table 1 in the Appendix which clearly shows that classes with less than 1000 do not pass a 0.70 per-class F1 score threshold. From this relationship, it is reasonable to conclude that class imbalance may affect a class's F1 score due to the model underfitting on lower observed classes. There are a few exceptions that don't follow this pattern such as weddings, home and living, divorce, sports, and queer voices with F1 scores between 0.85 to 0.88.

Looking at the lowest performing classes, good news and women, also provides insight into our model's performance. These categories have a per-class F1 score of 0.42 to 0.44, but there are over twice as many observations of the women class than the good times class. This sparked an investigation into why the model returned these results and highlighted potential causes of the observed underfitting. Consider the categories weddings, style and beauty, parenting, and home and living. Articles pertaining to these topics are often geared towards women, but were more accurately classified into their appropriate categories as evidenced by higher per-class F1 scores. This indicates further class consolidation could be required. Another pattern of note is that of the four identity based categories (women, latino voice, black voices, queer voices) the only one to perform above the success threshold is queer voices which contains twice as many observations than women and more observations than latino and black voices combined. This class imbalance could be another reason the model underfits certain categories.

These two patterns highlight representational issues that are present within the overall context of the dataset.

B.3.3 Summary of final model performance:

The RoBERTa model utilizing frozen layers and the imbalance dataset was selected from a total of 60 experiments that compared basic neural networks, custom neural networks, DistilBERT models, BERT model, and other RoBERTa models. Hyperparameters such as epochs trained, learning rate, batch size, subset frequency, use of frozen, partially frozen, or unfrozen layers, and balanced or balanced data were tuned for each type of model and the summary of these findings can be found in Milestone Two. This thorough experimentation highlighted the selected model as being the best performing due to its ability to consistently deliver the highest performance for our chosen HuffPost dataset. The only recognizable drawback of this model is the runtime. While returning the highest test accuracy and test F1 score, it also required the longest computation time. The run time never exceeded one hour for this assignment. While that is long by some standards, the HuffPost dataset is large and very complex which is supported by the other, more simple, models' lower results as compared to the RoBERTa model. Surprisingly, the imbalanced dataset outperformed the balanced dataset which is irregular. This can be explained by the imbalance causing the model to rely on more abstract context, thus increasing the semantic understanding, or balancing the data added too much noise. A stronger contextual understanding would improve the model's results.

B.4: Discussion and Reflection

B.4.1 Limitations and Sources of Error

The increased complexity for this model was a major limitation that resulted in higher computational cost and longer training time, though it also improved model results for overall

and per-class predictive capacity. There were also three main sources of error in this model: class imbalance, overall overfitting, and per-class underfitting. Though complete freezing is commonly used for RoBERTa text classifiers, unfreezing the proper layers (in scope and number) makes models more robust by enhancing adaptation and generalization to new data if done gradually and judiciously. Class imbalance also directly resulted in underfitting for some classes and overfitting overall, as evidenced by the difference between the weighted and macro F1 scores, where the macro F1 score was nearly 10% below the weighted F1 score and with the range of per-class F1 scores between 0.42 for women (which is moderate) and 0.88 for food and drink (which is good). Other than these three main sources of error, there are some limitations with the RoBERTa model itself. As mentioned previously, RoBERTa is computationally expensive to run, partly due to it being a larger model compared to traditional BERT models. This means longer training times and extra resources may be needed to run this model, which could harm its overall scalability (Amit, 2024). Another limitation of the model involves performance collapse when too many layers are frozen. It has been proven that when all layers are frozen, model performance typically drops. Despite this, our experiments show that the model with all frozen layers performed the best. These results are due to the size of our dataset and the class imbalance; this combination allowed for freezing the layers to prevent overfitting and preserve pretrained knowledge, yielding more stable generalizations (Lee et al., 2019).

B.4.2 Potential improvements

Moving forward, class imbalance could be addressed using the SMOTE (synthetic minority oversampling technique) and manual rebalancing methods, which would add more data points to improve model performance by enhancing existing underlying trends in the HuffPost dataset. We could also experiment with unfreezing various layers through hyperparameter tuning.

Additionally, using other (but similar) data, such as titles and descriptions from YouTube video content, during training can help ensure generalization to new data, so long as the proposed categories remain the same. Finally, further condensing categories and removing others that may be too broad or have poor per-class F1 scores (such as women) can improve results. This will ensure that these categories are not negatively influencing general measures, such as macro F1 score, weighted F1 score, and accuracy for testing.

B.4.3 Learning Reflections

From our experimentation we can conclude that higher complexity models, such as RoBERTa, provided the strongest overall performance compared to the simple and custom neural network models. Given the constrained time and computational resource restrictions associated with the RoBERTa model, though it may be costly compared to a simple or custom neural network model, its advantageous results outshine these costs. What surprised us the most about these results was the fact that class weights did not yield better macro F1 scores compared to using no rebalancing, as we expected that rebalancing would improve model generalization. What mattered the most for performance was avoiding overly large hyperparameter lists, using just enough of the data (such as a quarter of the available data) for training, validation, and testing, having a simplified model architecture, and ensuring that the news categories were a proper fit for each observation. If we were to approach modeling differently, we would have limited our experimentation to shorter hyperparameter lists, further condensed our categories, and ensured that additional measures for loss regularization, dropout, and selectively unfreezing layers were implemented. Another approach for us to consider is adding more dimensions, as it would allow the model to be more expressive by assigning more context to each sentence.

B.5: AI Use Disclosure

Gemini was used primarily for debugging support and technical brainstorming during model development. This included diagnosing potential causes of training instability, suggesting alternative hyperparameter configurations, and offering conceptual guidance when experimental results did not initially improve as expected. These suggestions informed additional testing but did not directly generate final model code. ChatGPT was used primarily for brainstorming and formatting assistance during the report writing phase. This included helping restructure sections for clarity, aligning content with rubric requirements, and improving the organization and flow of written explanations. While not AI, it should be noted that we also utilized Google docs spellcheck and Quillbot for fixing grammatical mistakes. AI tools were not used to generate final model implementations, results, or provide conclusions, as their outputs were sometimes limited due to incomplete context of either the dataset or our experimental setup. As a result, all AI outputs required manual verification, refinement, and validation by the project team. Advice for future students taking this module is to implement generative AI in a strategic way where it has a supportive role as opposed to a means to an end. Strategic implementation can include using accurate and concise language to prompt the AI tool to only fulfill the desired effects such as explaining complex material for individual learning or to proof-read a paper for grammatical mistakes only in a group context.

References

Amit, H. (2024, November 12). A Gentle Introduction to RoBERTa. Medium. https://medium.com/@heyamit10/a-gentle-introduction-to-roberta-1f1f9babb62d

Lee, J., Tang, R., & Lin, J. (2019, November 8). What Would Elsa Do? Freezing Layers During Transformer Fine-Tuning. arXiv.org. https://arxiv.org/abs/1911.03090

Massed Compute. (2024). How does dynamic masking in RoBERTa handle out-of-vocabulary words compared to static masking in BERT?. https://massedcompute.com/faq-answers/?question=How%20does%20dynamic%20masking%20in%20RoBERTa%20handle%20out-of-vocabulary%20words%20compared%20to%20static%20masking%20in%20BERT?

Misra, R. (2018). News category dataset [Data set]. https://doi.org/10.13140/RG.2.2.20331.18729 Retrieved from https://huggingface.co/datasets/khalidalt/HuffPost

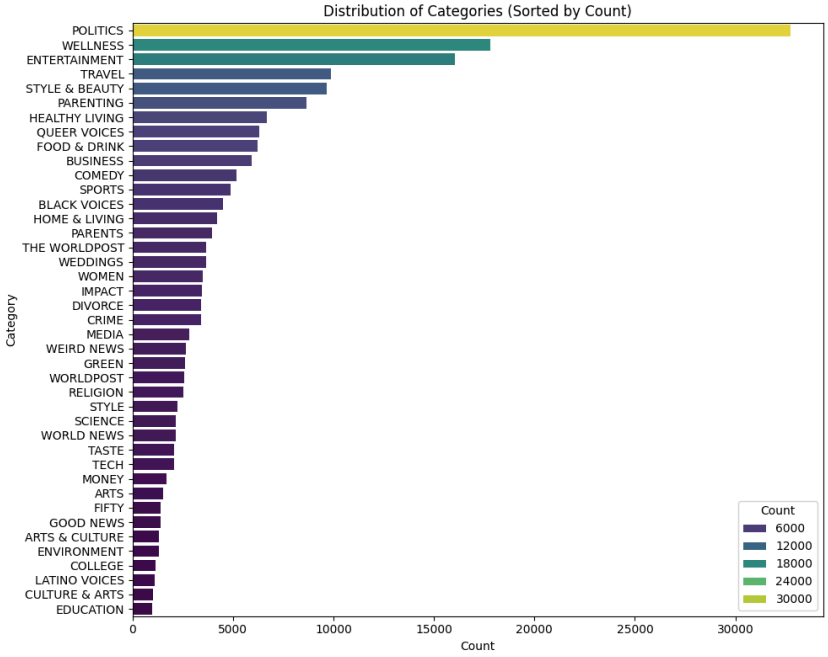
Romero, F. Y. (2024, September 19). Dealing With Encoder Language Model Tasks Using Pytorch. Towards AI. https://pub.towardsai.net/dealing-with-encoder-language-model-tasks-using-pytorch-18be8a38dde1

Wei, D. (2024, January 30). Demystifying the Adam Optimizer in Machine Learning. Medium. https://medium.com/@weidagang/demystifying-the-adam-optimizer-in-machine-learning-4401d162cb9e

Appendix

Figure 1

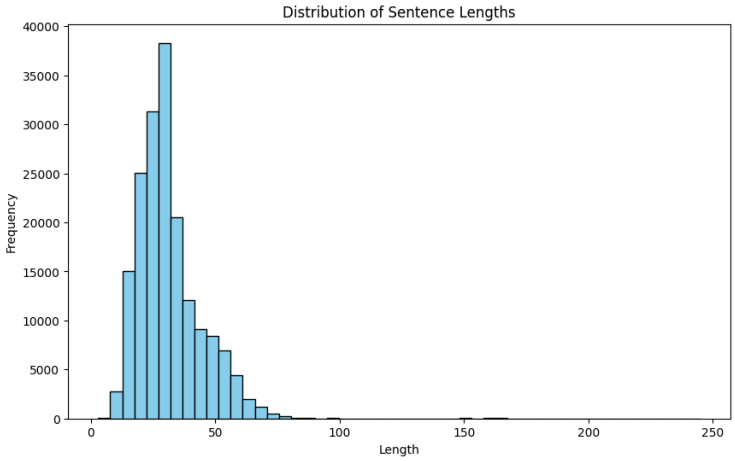
Distribution of Text By Original Categories



Note: The news articles are skewed towards politics, wellness, entertainment, travel, and the style and beauty domains.

Figure 2

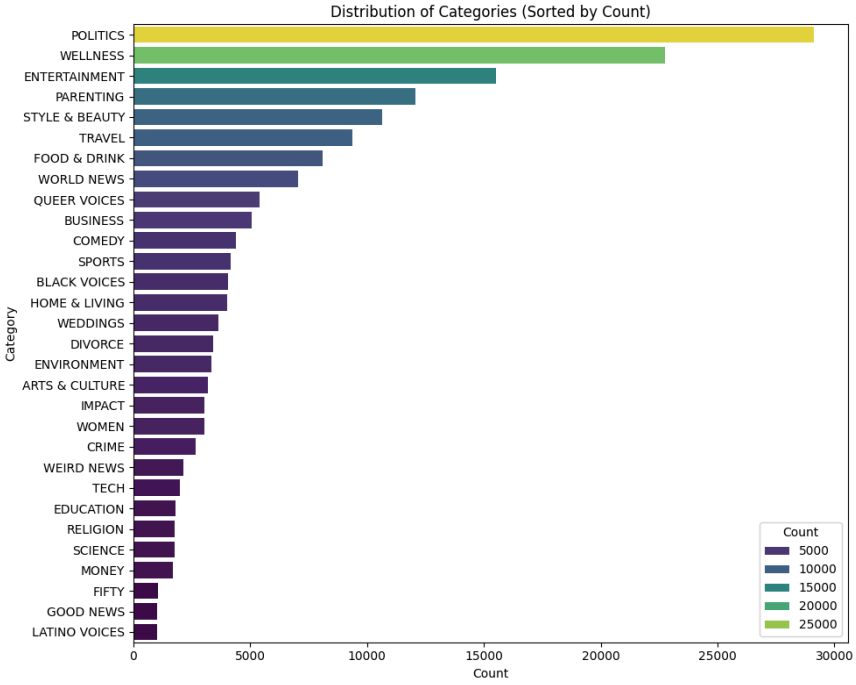
Distribution of Observations By Text Length



Note: The sentences had a median length of 29 and a mean length of 31, with a slight right skew.

Figure 3

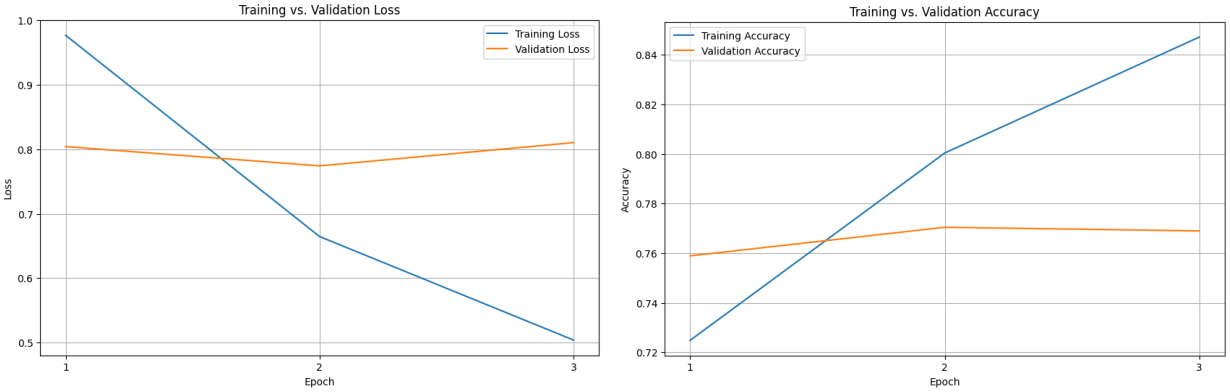
Distribution of Text By Condensed Categories



Note: The news articles are skewed towards politics, wellness, entertainment, parenting, and the style and beauty domains.

Figure 4

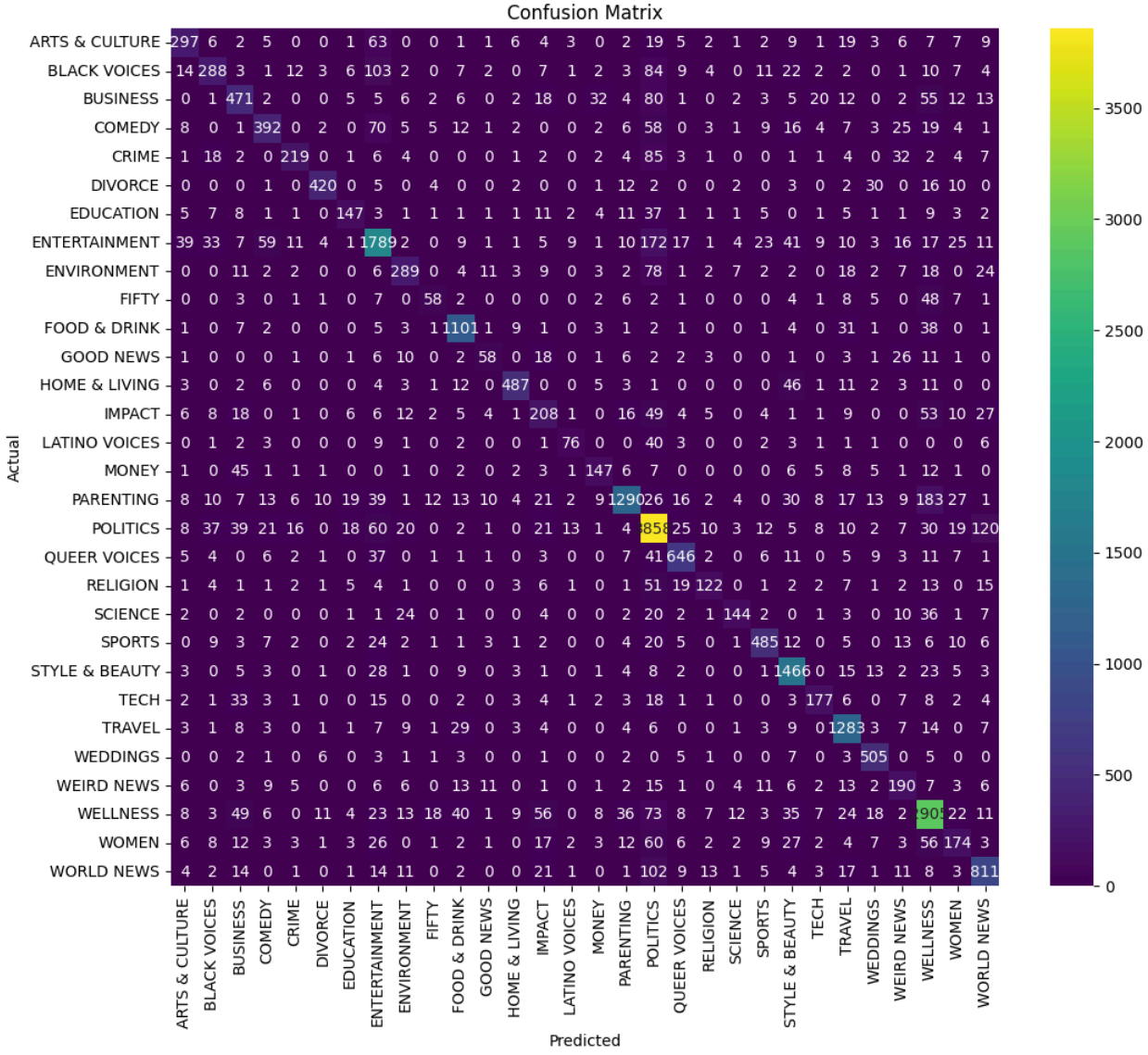
Training vs. Validation Loss and Accuracy Curves



Note: Validation loss was relatively stable, compared to the decrease for training loss, with the inverse visible for training accuracy and the plateauing of validation loss.

Figure 5

Confusion Matrix of Predictions By The Condensed Categories



Note: The lowest per-class score was 0.42 for WOMEN while the highest was 0.88 for FOOD & DRINK, with an overall macro F1 score of 0.6782.

Table 1

Counts Per Category Before (EDA) and After Preprocessing

Index	Category	Count Before Preprocessing (EDA)	Count After Preprocessing
0	ARTS	1,590	3,206
0	ARTS & CULTURE*	1,339	-
1	BLACK VOICES	4,528	4,070
2	BUSINESS	5,937	5,056
6	COLLEGE*	1,144	-
3	COMEDY	5,175	4,373
4	CRIME	3,405	2,667
0	CULTURE & ARTS*	1,030	-
5	DIVORCE	3,426	3,402
6	EDUCATION	1,004	1,812
7	ENTERTAINMENT	16,058	15,533
8	ENVIRONMENT	1,323	3,353
9	FIFTY	1,401	1,042
10	FOOD & DRINK	6,226	8,094

11	GOOD NEWS	1,398	1,026
8	GREEN*	2,622	-
27	HEALTHY LIVING*	6,694	-
12	HOME & LIVING	4,195	4,003
13	IMPACT	3,459	3,048
14	LATINO VOICES	1,129	1,017
7	MEDIA*	2,815	-
15	MONEY	1,707	1,705
16	PARENTING	8,667	12,065
16	PARENTS*	3,955	-
17	POLITICS	32,739	29,131
18	QUEER VOICES	6,314	5,399
19	RELIGION	2,556	1,774
20	SCIENCE	2,178	1,761
21	SPORTS	4,884	4,165
22	STYLE*	2,254	-

22	STYLE & BEAUTY	9,649	10,645
10	TASTE*	2,096	-
23	TECH	2,082	1,984
29	THE WORLDPOST*	3,664	-
24	TRAVEL	9,887	9,377
25	WEDDINGS	3,651	3,645
26	WEIRD NEWS	2,670	2,156
27	WELLNESS	17,827	22,746
28	WOMEN	3,490	3,030
29	WORLD NEWS	2,177	7,068
29	WORLDPOST*	2,579	-

Note: Labels that no longer exist after preprocessing were italicized and identified with an asterisk (*), while their counts after preprocessing are denoted by a dash (-).

Table 2

Macro F1 scores and Frequencies for Predictions

Index	Category	Macro F1 Score	Support	Frequencies
0	ARTS	0.65	481	1.80%

1	BLACK VOICES	0.55	610	2.28%
2	BUSINESS	0.62	759	2.84%
3	COMEDY	0.65	656	2.45%
4	CRIME	0.64	400	1.50%
5	DIVORCE	0.86	510	1.91%
6	EDUCATION	0.60	272	1.02%
7	ENTERTAINMENT	0.76	2330	8.71%
8	ENVIRONMENT	0.62	503	1.88%
9	FIFTY	0.44	157	0.59%
10	FOOD & DRINK	0.88	1214	4.54%
11	GOOD NEWS	0.44	154	0.58%
12	HOME & LIVING	0.85	601	2.25%
13	IMPACT	0.46	457	1.71%
14	LATINO VOICES	0.57	152	0.57%
15	MONEY	0.60	258	0.96%
16	PARENTING	0.79	1810	6.77%

17	POLITICS	0.82	4370	16.33%
18	QUEER VOICES	0.81	810	3.03%
19	RELIGION	0.54	266	0.99%
20	SCIENCE	0.63	264	0.99%
21	SPORTS	0.79	624	2.33%
22	STYLE & BEAUTY	0.87	1597	5.97%
23	TECH	0.64	297	1.11%
24	TRAVEL	0.86	1407	5.26%
25	WEDDINGS	0.86	546	2.04%
26	WEIRD NEWS	0.54	323	1.21%
27	WELLNESS	0.82	3412	12.75%
28	WOMEN	0.42	455	1.71%
29	WORLD NEWS	0.75	1060	3.96%

Note: Labels with over 1,000 predictions tended to have good macro F1 scores of at least 0.7, while those under 1,000 tended to have less predictable macro F1 scores that were sometimes good or only moderate (between 0.4 and 0.7).