## I. Data Source(s)

The MIND, or Microsoft News Dataset, was released in 2020 as a benchmark for personalized news recommendation research. It was constructed from user click logs of the Microsoft News website, collected over a 6-week period in 2019 (Microsoft). There are roughly 1 million users, about 160,000 news articles, and over 15 million news impression logs that make up over 24 million click events captured in the dataset. All user IDs were hashed and delinked from the production system (Microsoft).

Each impression log in MIND represents a single session where the user visits the news site's homepage at a certain time, and is presented with a list of news articles. The log records which news articles were displayed in that impression and which ones the user clicked. The MIND training set comprises over 15 million impressions and captures implicit feedback signals for our model.

There are two primary tables in MIND: a behaviors file (behaviors.tsv), and a news content file (news.tsv). behaviors.tsv contains the impression logs and user history over 5 tab-separated columns:

- Impression ID; a unique identifier for the impression instance,
- User ID; an anonymized identifier for each user,
- Time; or timestamp of the impression,
- History; or the list of news article IDs clicked by the user before the current impression, and
- Impressions; or the list of news presented in the impression with a label for each indicating whether it was clicked or not.

The behaviors file not only logs which articles were shown and clicked, but also provides each user's recent click history to model the user's interests.

news.tsv contains metadata for each news article that appears in the behaviors log, delineated over 7 tab-separated columns:

- News ID; the unique identifier for each article,
- Category; the high level category of the news (e.g. sports, politics, entertainment),
- Subcategory,
- Title,
- Abstract; or a short summary or abstract of the article,
- URL; or a link to the full article on MSN,
- Title entities; or keywords that appear in the title, and
- Abstract entities; or keywords in the abstract.

Due to licensing restrictions, the dataset does not include the full body text of each article, but it does provide a URL to get to the article as needed (Microsoft).

## II. Related Works

Svensson & Blad applied topic modeling to cluster Swedish news articles with NMF and LDA, finding that both methods can produce meaningful topic groupings of news content. However, the authors provide the caveat that while LDA and NMF are valuable for large-scale news categorization, they can sometimes yield incoherent or misleading results. These are still useful models as content analysis tools to segment news articles by theme (Svensson and Blad).

Fan *et al.* proposed an online news topic detection and tracking approach with a hierarchical Bayesian nonparametric model, clustering news articles into emerging topics over time by allowing an adaptive number of topics to be shared across news stories. This clustering-based approach discovers topics on the fly as news streams in. This hierarchical Bayesian framework groups related news events and tracks their evolution, improving the detection of new topics in a news stream (Fan et al. 2126).

Vaidehi Patel and Arpita Patel have done similar work that uses agglomerative clustering to group news articles into a number of topics to determine themes and when something new has happened (Patel and Arpita). Our group's work is different because we use a different dataset and different clustering methods.

## III. Unsupervised Learning

### a. Motivation

We are going to group articles based on the words that appear in their titles into a predefined number of topics. To group similar articles together, we constructed a Latent Dirichlet Allocation (LDA) model and a Non-negative Matrix Factorization (NMF) model. Topic modeling can help: organize articles which makes it easier to find articles related to the same topic; identify patterns in what is being talked about; and potentially enhance the supervised learning part of this project.

### b. Data Source

We utilized the same MIND dataset, described in detail above, for both unsupervised and supervised learning.

### c. Unsupervised Learning Methods

First, we preprocessed the titles of the articles. To reduce noise, we removed stop words, or those that have less than 3 characters or have little meaning, such as *a*, *to*, *the*, *he*, and *cause*. By removing them, the models can focus on meaningful words. We also removed punctuation and numbers from the text to limit the number of features (words) being processed. Additionally, we lemmatized the text to return words back to their base form and reduce redundancy. Lastly, because LDA and NMF models require text to be converted to a numerical representation, we fitted a TF-IDF (Term Frequency-Inverse Document Frequency) vectorizer on 1-word phrases in the train set to represent the importance of a word across different articles.

To group articles into topics, we fitted an LDA model for an unsupervised, probabilistic approach to topic modeling. There are two main parameters in LDA, alpha ($\alpha$) and eta ($\eta$). $\alpha$ represents a prior belief about the distribution of each topic in each document. $\eta$ represents a prior belief about the distribution of each word in each topic. LDA works to find a topic-document distribution and a topic-word distribution that optimizes the likelihood of observing the text. It is a well-known algorithm for topic modeling and easy to adjust to a different dataset.

We also fitted an NMF model. NMF is an algorithm that implements matrix factorization. In NMF, a matrix X is broken into 2 matrices, W and H. All the elements of the matrices W and H are non-negative. To obtain matrices W and H, the NMF algorithm works to minimize $||X - WH||$. The resulting matrix W gives the importance of each article to each topic, while the matrix H gives the importance of each word to each topic. We chose an NMF model because it is highly interpretable and reasonably efficient.

For the LDA and NMF models, it is crucial to specify an optimal number of topics beforehand. We fitted the models multiple times on the training set and each time provided a different number of topics to fit (n_components). The range we tried for n_components was 2 through 10. For each model fitted on a different value for n_components, we calculated the highest topic coherence. Those with a higher topic coherence are easier to interpret because of their ability to find clear underlying themes in the data. To calculate topic coherence, we used the training set and fitted a Word2Vec model to generate word embeddings, which allowed us to represent a word as a numeric vector and calculate how similar two words are to each other. Once we fitted the Word2Vec model, we obtained the word embeddings for the top 10 words in each topic. For each top word in each topic, we calculated its cosine similarity to each of the other top words and took the average to obtain the topic coherence. Then we took the median topic coherence over all topics.

d. Unsupervised Evaluation

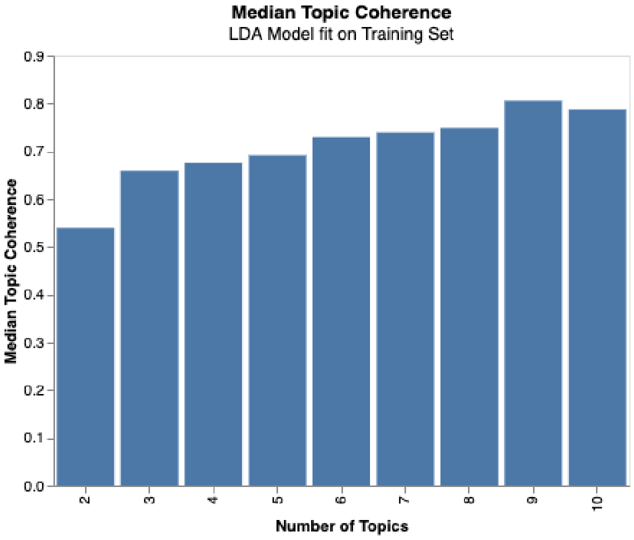


**Figure 1: Finding the optimal number of topics (LDA)**

Based on the method described in the *Unsupervised Learning Methods* section for the LDA model, there did not seem to be a choice for the number of topics that stuck out. To balance coherence and human interpretability, we decided to fit 4 topics. Fitting 4 topics gave a median topic coherence of 0.68 on the training data. After determining the number of topics, we fit the LDA model on the entire dataset.

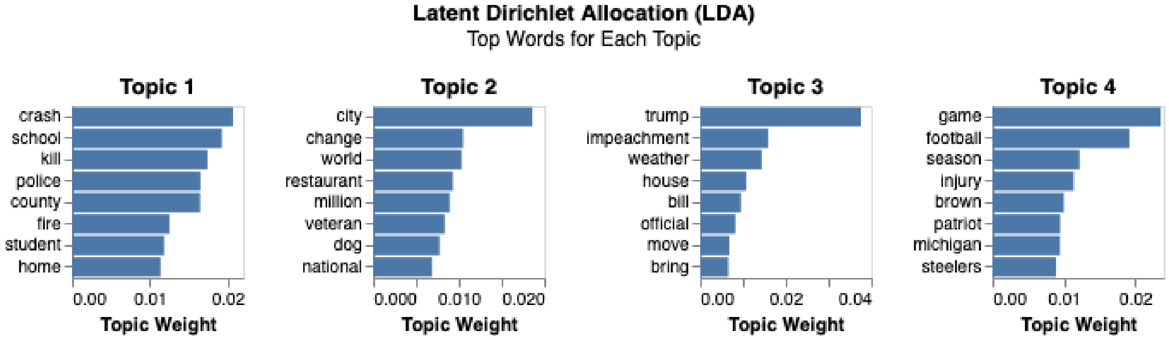


**Figure 2: Top words for each topic (LDA)**

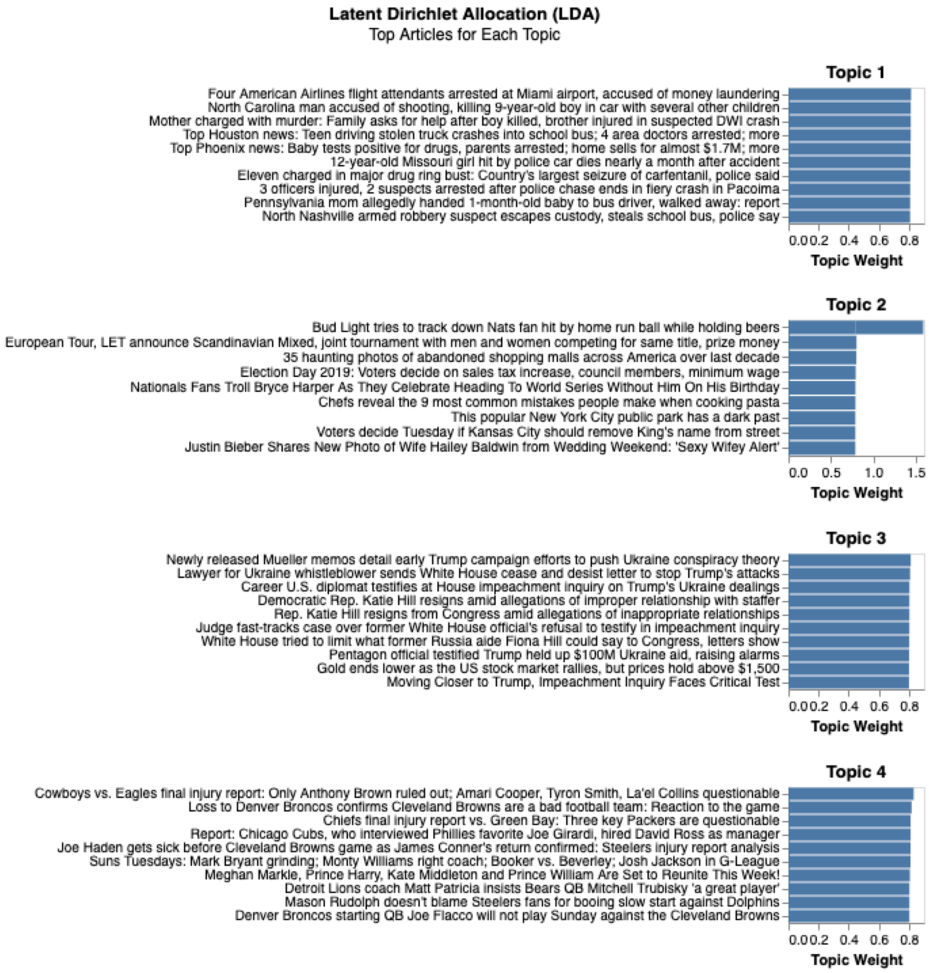


**Figure 3: Top articles for each topic (LDA)**

The LDA model fitted with 4 topics looks to build coherent topics. Though it is a little unclear what topic 2 is about. Using Figure 2 and Figure 3, we arrived at the conclusions below.

- Topic 1 looks to be related to crime, accidents, and disasters.
- Topic 2 seems to be related to lifestyle but is a little unclear.
- Topic 3 is related to government and legislation.
- Topic 4 looks to be related to sports.

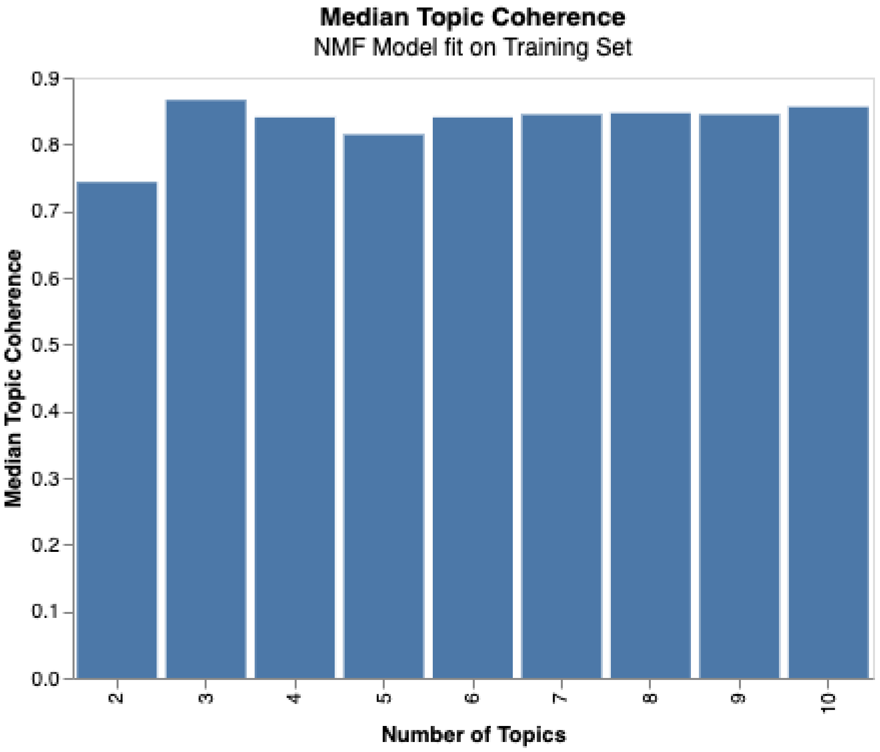


**Figure 4: Finding the optimal number of topics for the NMF model**

We also fitted an NMF topic model. Based on the method previously described in the *Unsupervised Learning Methods* section, we decided to fit an NMF model with 3 topics. Fitting 3 topics gave a median topic coherence of 0.87 on the training data. Again, after determining the number of topics, we fit the NMF model on the entire dataset.

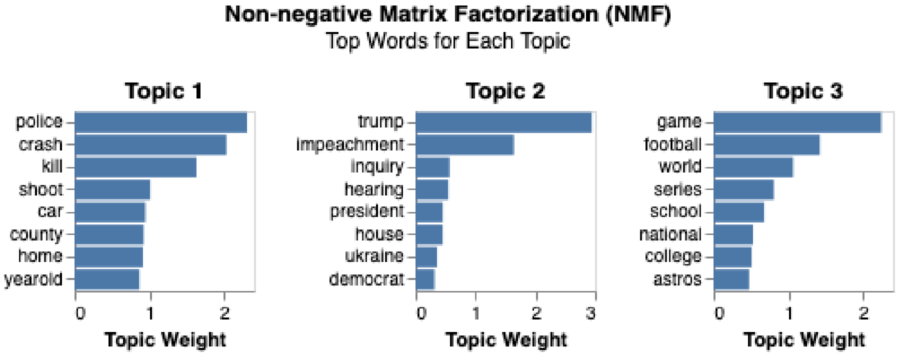


**Figure 5: Top words for each topic (NMF)**

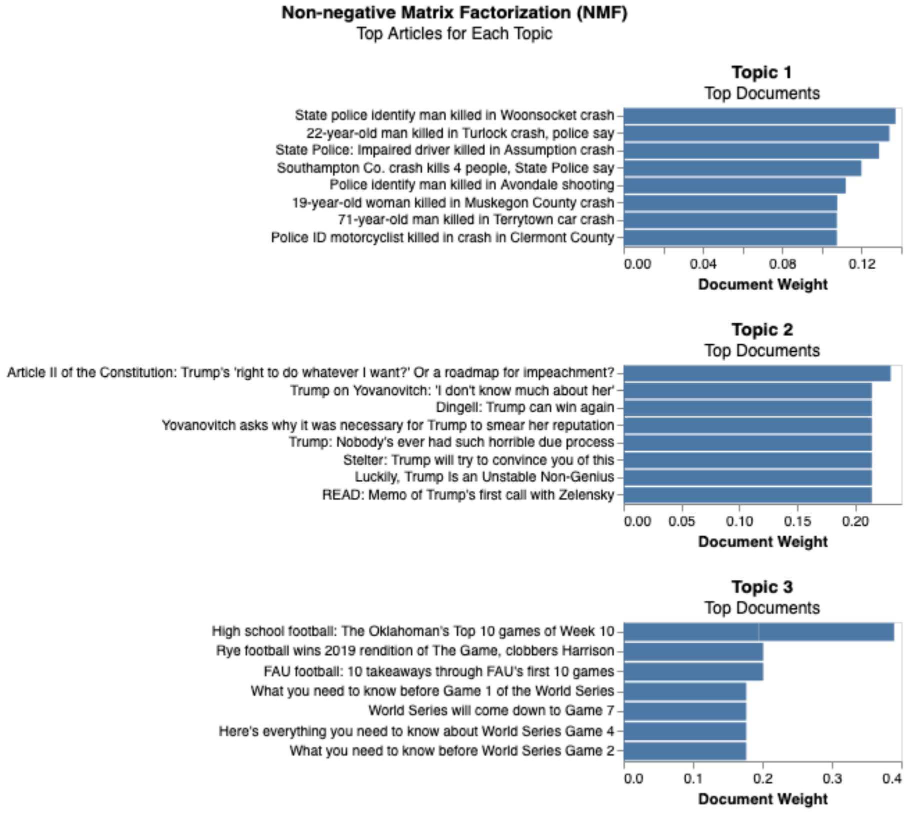


**Figure 6: Top articles for each topic (NMF)**

The NMF model fitted with 3 topics also looks to build coherent topics. Using Figure 5 and Figure 6, we arrived at the conclusions below.

- Topic 1 looks to be related to crime, accidents, and disasters.
- Topic 2 is related to government and legislation.
- Topic 3 looks to be related to sports.

Given the LDA or NMF topic model, a majority of articles we see today could fit into one of the topics generated by the topic models. For example, though it is unfortunate, we hear a lot of stories that involve natural disasters because they garner a lot of attention from the public. The same applies to stories about the government and sports. Given the LDA and NMF models, our group would choose the LDA model because the LDA model captures a 4th category, that being the lifestyle or miscellaneous category. Many articles we see today could fit into the lifestyle or miscellaneous category.

e. Sensitivity Analysis

We looked at the LDA model's sensitivity to the parameter *alpha (α)*, or prior belief about the distribution of each topic in each document. A high *α* means that documents contain many topics, and a lower *α* means that documents are composed of only a few topics. To test the LDA model's sensitivity to *α*, we fitted an LDA model on the training set multiple times, each time with a different value of *α* but keeping the number of topics the same (4). To add, the number of trials for each value of *α* was 50. The number of trials is because when fitting an LDA model, different random initializations lead to different results. Lastly, for each value of *α*, we took the average of the median topic coherence score of trials 1 through 50.

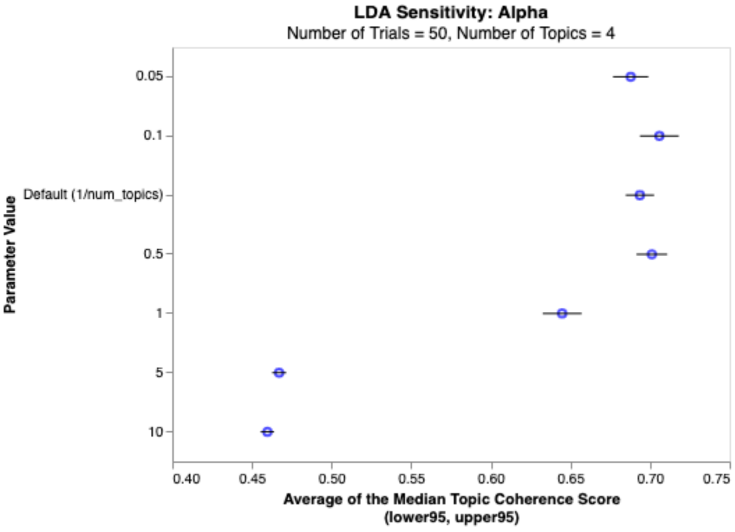


Figure 7: Sensitivity analysis (LDA)

Based on Figure 7, an LDA model is sensitive to the value of *α*. Values of 5 and 10 gave a lower average median topic coherence score. To add, a value of 1 gave an average median coherence score of about 0.65, and the rest of the values that were tested gave an average median coherence score around 0.7. Overall, it looks like as the value of *α* increases, the LDA model discovers less coherent topics, but more testing needs to be done.

### f. Discussion

Our group learned many things from topic modeling. First, it is important to remove stop words from text when performing topic modeling. This way, the topic model is able to focus on more important words and create more interpretable topics. Secondly, we learned that it is important to balance coherence and interpretability. Sometimes, as the number of topics that are fitted increases, so does the topic coherence. Coherence may have to be sacrificed in order to create a more interpretable topic model. Something that surprised our group was the topic model's ability to capture a topic centered around crime, accidents, and disasters. It is unfortunate that such events occur, but it was neat to see that topic being captured by the topic models. A challenge that we encountered was deciding which stop words to include. Besides the basic stop words, such as *a*, *the*, and *to,* our group met and brainstormed a list of additional stop words to include. Lastly, if given more time/resources, we would like to incorporate dissimilarity of two topics because we would like the terms in the same topic to be similar and the terms of different topics to be dissimilar.

There are ethical considerations that could arise from the clustering of news articles. Topic models could learn to group together articles that amplify stereotypes, which could lead to increased discrimination against those that fall under the stereotype (Media). To add, clustering new articles could limit the different views and perspectives that the public hears. It is important that the public hear news that is unbiased. To address the ethical issues, our group, co-workers, and professionals could check that the resulting topics are being interpreted in an unbiased way and are broad so they could apply to many news articles that involve different views and groups. If the resulting topics appear biased or discriminatory, then we suggest refitting the topic model with a different number of topics or making a change to stop words.

## IV. Supervised Learning

a. Motivation

The goal of our project is to predict user engagement with news articles, specifically focusing on the number of clicks each article receives on the platform. More accurate click predictions can help editors prioritize which stories to feature and better understand what drives reader interest.

To address this, we formulated both a classification problem (predicting whether an article will receive any clicks) and a regression problem (predicting the total number of clicks). We used features such as article text, news category, and detected named entities, aiming to identify the key factors that contribute to news article popularity.

b. Data Source

We utilized the same MIND dataset, described in detail above, for both unsupervised and supervised learning.

c. Methods and Evaluation

To address the click prediction task, we explored a range of supervised learning algorithms for both classification and regression. The models evaluated included:

- Logistic Regression (classification) and Ridge Regression (regression) – these serve as strong, interpretable linear baselines.
- Naive Bayes (classification only) – a commonly used method for text-based features.
- Random Forest (classification and regression) – a popular non-linear, tree-based ensemble model that can capture more complex interactions.
- Word2Vec-based models – where we replaced TF-IDF features with averaged Word2Vec embeddings to test the effect of richer text representations.

All models were trained and evaluated using the predefined train/validation split in the dataset's "set" column. We one-hot encoded categorical features and experimented with both TF-IDF and Word2Vec representations for the text fields. To address the significant class imbalance in the data (with over 86% of articles not clicked), we applied random oversampling to the training set in our classification tasks. Model performance was evaluated using accuracy, precision, recall, and F1-score for classification, and Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and $R^2$ for regression.

d. Key Results

Applying oversampling had a substantial impact on classification performance. For example, the F1-score for the minority "clicked" class improved from 0.07 (original logistic regression) to 0.20 after oversampling, with an acceptable decrease in overall accuracy (from 0.91 to 0.67). Naive Bayes and Random Forest classifiers showed similar improvements in minority class detection when oversampling was applied. However, regression models, including Ridge Regression and Random Forest Regressor, continued to exhibit low predictive power for the exact number of clicks, as reflected by high RMSE values and negative $R^2$ scores.

We also experimented with combining Word2Vec embeddings and categorical features, applying oversampling during training. This approach led to a moderate improvement in the detection of the minority "clicked" class: the recall for clicked articles increased to 0.49 from 0.04

and the F1-score reached 0.20 from 0.06. However, this came with a reduction in overall accuracy to 0.67 from 0.91, reflecting the trade-off between identifying rare events and maintaining general performance. While these results are a step forward, the challenge of reliably predicting engagement for rare articles remains significant, and no single feature set or modeling technique proved decisive.

| | Logistic Regression | Ridge Regression | Naive Bayes | Random Forest Classification | Random Forest Regression | Logistic Regression (w/Word2Vec) | Ridge Regression (w/Word2Vec) |
|---|---|---|---|---|---|---|---|
| Accuracy | 0.91/0.67 | | 0.91/0.60 | 0.95/0.60 | | 0.91/0.67 | |
| F1 Score | 0.07/0.20 | | 0.036/0.20 | 0.00/0.11 | | 0.06/0.20 | |
| RMSE | | 61.97 | | | 45.56 | | 61.67 |
| $R^2$ | | -0.07 | | | -0.02 | | -0.06 |

To illustrate the model’s performance, we present two key visualizations. The confusion matrix (see notebook) shows that while the classifier is able to identify a substantial number of clicked articles (1183), it also misclassified many not-clicked articles as clicked, resulting in a high number of false positives. This reflects the challenge of predicting rare engagement events, where increasing sensitivity to the minority class often leads to reduced overall precision.

The scatter plot (Figure 8) of true versus predicted click counts from the regression model highlights a strong bias toward lower values. Most predictions are clustered near zero, regardless of the true number of clicks, and the model fails to capture articles with unusually high engagement. This pattern underscores the difficulty of accurately modeling such highly imbalanced and skewed data.

Overall, oversampling proved to be a critical step for improving the practical utility of our classification models.

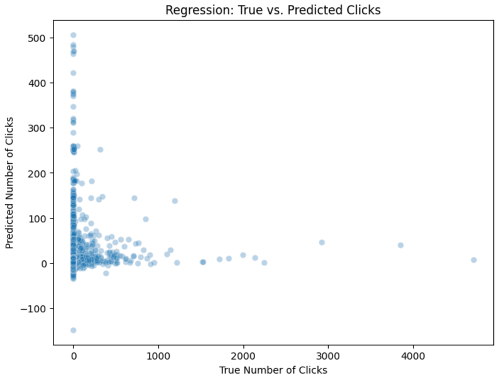


Figure 8: Regression: True vs. Predicted Clicks

e. Feature Analysis

We conducted feature importance analysis using both logistic regression coefficients and random forest feature importances, based on the combined TF-IDF, categorical, and Word2Vec feature set.

For logistic regression, the most influential features were specific named entities and one subcategory. The top positive coefficients were associated with entity labels such as “O'Hare International Airport,” “Matt Kuchar,” “Michigan–Ohio State football rivalry,” and “Union County.” Additional entities like “Venice,” “Mike Norvell,” “Colorado Buffaloes,” “Milky Way,” and “Violence Against Women Act” also ranked highly. The only non-entity in the top features was the subcategory “awardstyle,” it suggests that articles tagged with awards or style-related content may attract more clicks. This pattern indicates that stories mentioning prominent individuals, places, sporting events, and specific themes are more likely to receive user engagement.

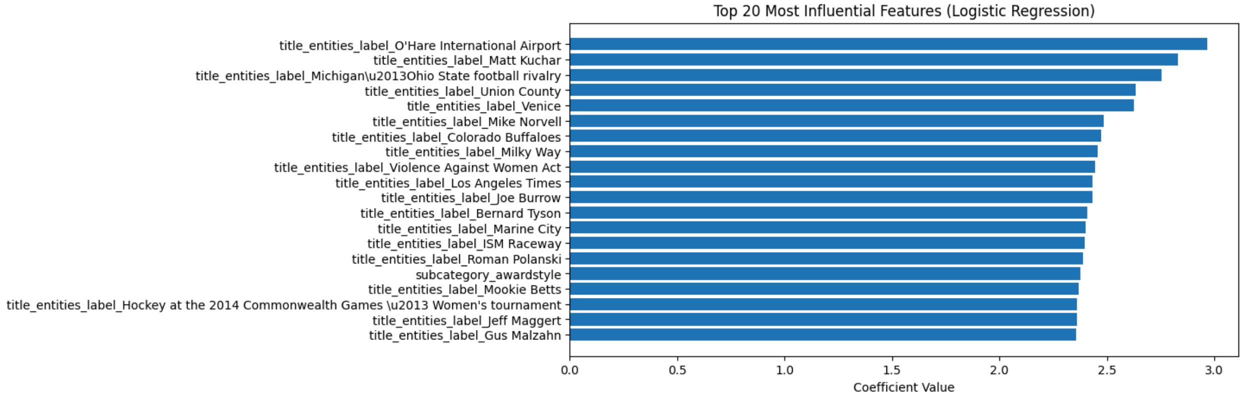


**Figure 9: Top 20 Most Influential Features (Logistic Regression)**

For the random forest model, the most important features were primarily TF-IDF text features (keywords). The top-ranked words included “shot,” “team,” “nfl,” “impeachment,” “family,” and “area.” Other influential terms such as “november,” “white,” “injured,” “california,” “shooting,” and “food” also appeared in the top 20. This suggests that the model identifies newsworthy words tied to sports, major events, and societal issues as strong predictors of engagement, aligning with common trends in news consumption.

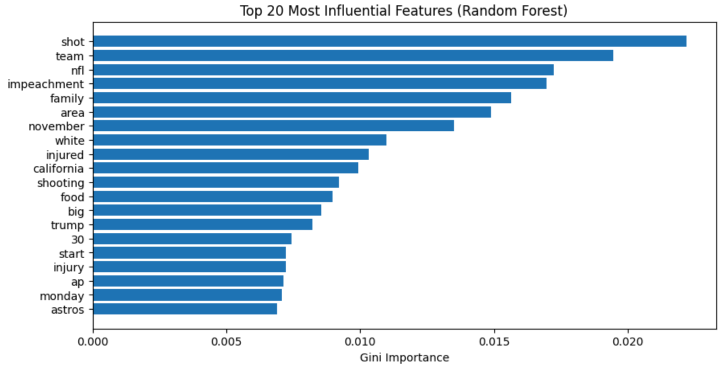


**Figure 10: Top 20 Most Influential Features (Random Forest)**

Key Findings:

- Named entities like “O'Hare International Airport,” “Matt Kuchar,” and “Michigan–Ohio State football rivalry” were among the most predictive features for article clicks.
- The subcategory “awardstyle” also had a notable positive association with engagement.
- For random forest, sports- and news-related keywords such as “shot,” “team,” “nfl,” and “impeachment” were the most important. It confirms that topical relevance and event-driven content drive user interest.

- The differences results between two models provide complementary perspectives on which article attributes are linked to higher engagement.
- Overall, both structured entity metadata and content keywords are critical for predicting which news stories are likely to generate higher engagement.

f. Learning Curve Analysis

To understand how model performance scales with additional training data, we plotted learning curves for both the logistic regression classifier (using F1 score) and the ridge regression model (using RMSE) across increasing training set sizes.

For classification, the learning curve (Figure 11) shows that both training and validation F1 scores are quite low overall, it reflects the inherent difficulty of the task and the strong class imbalance. While both scores improve as more data is used, the validation F1 remains well below the training F1, and the gap narrows only slightly with larger datasets. This suggests that simply increasing the amount of training data may offer limited improvement for this classification problem, and that performance is likely constrained by the data's imbalance and feature informativeness rather than sample size alone.

For regression, the ridge regression learning curve (Figure 12) shows that the validation RMSE remains high and relatively flat as the training set grows, indicating persistent difficulty in predicting the exact number of clicks. The gap between training and validation RMSE also increases with more data, a sign of mild overfitting and the challenge of generalizing from noisy and skewed targets. The consistently high error further highlights that regression models struggle to accurately capture rare, high-engagement events in the data.

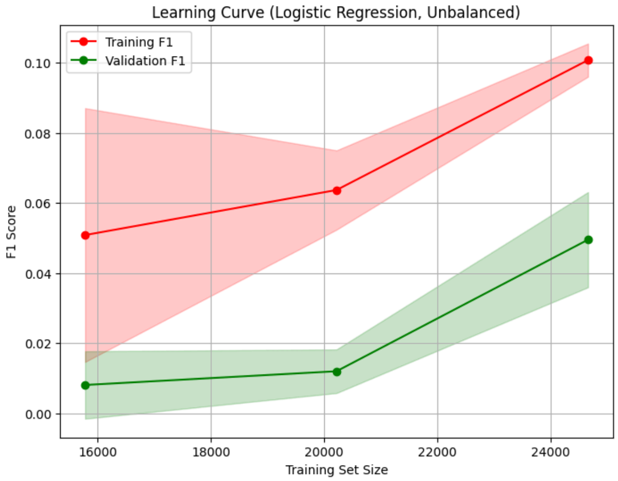


Figure 11: Learning Curve (Logistic Regression, Unbalanced)

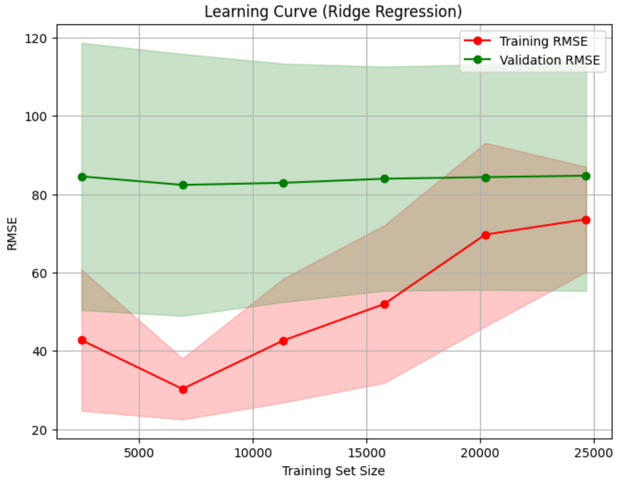


Figure 12: Learning Curve (Ridge Regression)

Key Findings:

Both learning curves suggest that under current conditions, model performance is not fundamentally limited by training set size. Instead, feature engineering, data imbalance, and the nature of the engagement signal are the main bottlenecks for further improvement.

g. Sensitivity Analysis

To evaluate how sensitive our model's performance is to regularization strength, we trained logistic regression classifiers using a range of C values (inverse of regularization). The plot (Figure 13) shows F1 scores (for the clicked class) on both the training and validation sets.

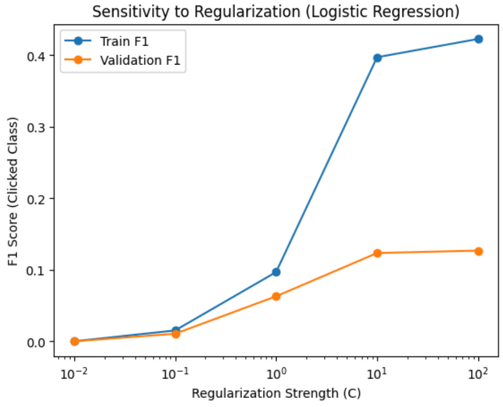


**Figure 13: Sensitivity to Regularization (Logistic Regression)**

As C increases (i.e., regularization weakens), the model becomes more flexible and training F1 rises rapidly, eventually approaching 0.4. However, the validation F1 increases more gradually and plateaus at around 0.13, with a persistent gap between training and validation performance. At very low regularization (high C), the model may begin to overfit as indicated by the widening gap. We find that the model's ability to identify clicked articles is not highly sensitive to the choice of regularization parameter within the tested range, but gains from reducing regularization are limited by overfitting and the imbalanced nature of the data. The modest overall validation F1 suggests that factors such as class imbalance and feature informativeness have a stronger influence on performance than hyperparameter tuning alone.

h. Error (failure) Analysis

To better understand where our models struggle, we examined concrete misclassifications and large regression errors from the validation set, inspecting actual article titles, text, and feature contributions.

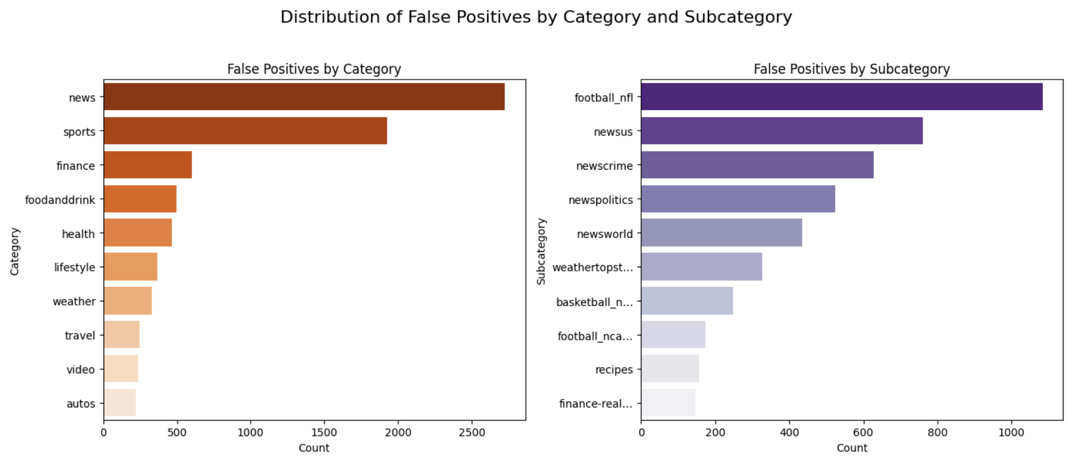
1. Classification Errors

We analyzed false positives (articles predicted as "clicked" but not actually clicked) and false negatives (clicked articles missed by the model) by inspecting the actual article titles, text, categories, subcategories, and named entities.

- False Positives:

  To systematically identify where our classifier over-predicts engagement, we analyzed all false positives in the validation set. The most common categories were news (2,726 cases), sports (1,925), finance (602), food and drink (496), and health (466). Within subcategories, football_nfl (1,084 cases), newsus, newscrime, newspolitics, and newsworld dominated the list.

**Figure 14: Distribution of False Positives by Category and Subcategory**
**(the next page)**



  - Example: A news article in the "newsus" or "football_nfl" subcategory predicted as "clicked" but which did not actually receive engagement.
  - Possible Reason: These results suggest that the model overestimates engagement for articles in high-volume categories and especially for popular subcategories like major league sports or U.S. news. This may be due to overrepresentation of these topics in the training data or to strong feature weights learned for specific entities or keywords.
  - Category: Systematic error due to overreliance on category/subcategory frequency.
  - Suggested Fix: Refine feature engineering to better distinguish between truly engaging content and high-frequency but low-engagement articles within these categories.
- False Negatives:
  We also analyzed all false negatives where the model failed to identify genuinely engaging ("clicked") articles. The majority of these errors were found in the news (339 cases) and sports (237) categories, with additional representation from lifestyle, travel, finance, and food and drink. Among subcategories, newsus (182), football_nfl (78), and newstrends (46) were most common.

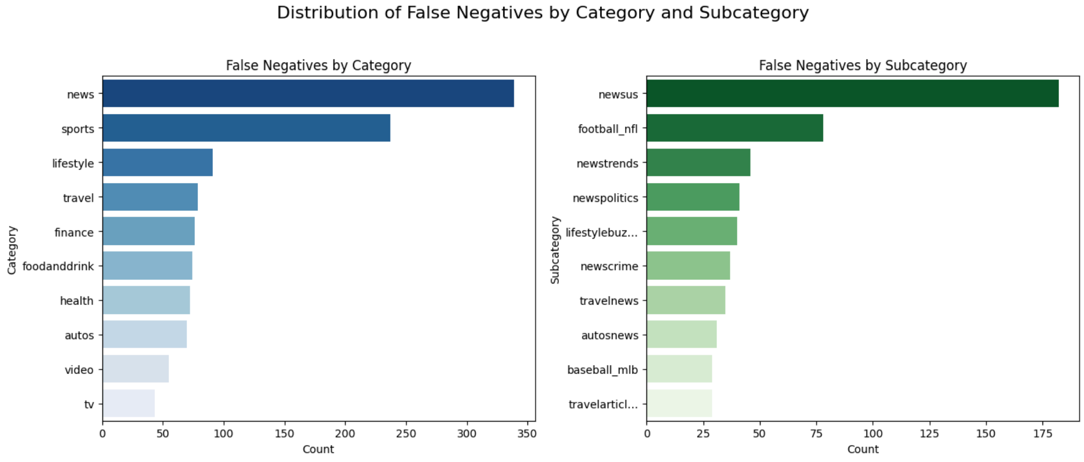


**Figure 15: Distribution of False Negatives by Category and Subcategory**

    - Example: A “newsus” or “football_nfl” article that actually received clicks, but the model predicted as “not clicked”.
    - Possible Reason: These findings indicate that even within high-engagement categories, the model struggles to recognize all click-worthy articles, potentially due to feature overlap with unclicked stories, insufficient signal from the text, or underrepresentation of certain content variations in the training data.
    - Category: Coverage and representation gaps, or/and ambiguous feature patterns.
    - Suggested Fix: Improve feature engineering for distinguishing between engaging and non-engaging articles within major categories. Expand training data for underrepresented types of “clicked” content, and consider additional contextual or temporal features to better capture what differentiates genuinely engaging articles.

2. Regression Errors

In the regression task, the largest errors occurred on articles with exceptionally high engagement (e.g., viral sports stories or seasonal shopping guides). The model consistently underpredicted these, which shows the challenge of capturing rare outliers in a skewed dataset.

- Example: A sports article with thousands of clicks predicted at under 50.
- Possible Reason: Model regresses to mean due to target skewness and rare extreme events.
- Category: Outlier/extreme value.
- Suggested Fix: Explore quantile regression, oversample high-engagement cases, or add event/context features.

The histogram of regression residuals (Figure 16) shows that most prediction errors are tightly clustered around zero, it indicates that the model fits the majority of low-engagement articles reasonably well. However, the long tail of large residuals reflects a recurring issue: the model systematically underpredicts the click counts for rare, highly engaging articles. This skewed error distribution further supports our earlier finding that regression performance is limited by the prevalence of outliers in the data. Future work should explore more robust modeling techniques, such as quantile regression or targeted sampling, to better capture these rare events.

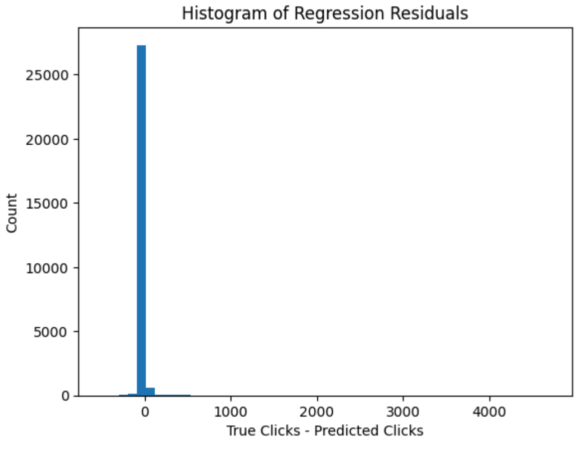


**Figure 16: Histogram of Regression Residuals**

Key Findings:

- Most false positives and false negatives come from news and sports categories, showing the model's difficulty in telling which stories in these broad areas will get real engagement.
- The model often predicts clicks for articles with generic titles for popular subcategories, even when actual user interest is low.
- Many missed clicks happen for genuinely engaging articles with less distinctive text, rare topics, or underrepresented categories. Suggest a need for more diverse and detailed features.
- The regression model generally predicts low-engagement articles well, but consistently underestimates clicks for rare, highly popular stories, as shown by a long-tailed residual error distribution.

## I. Discussion

One of the main challenges in this project was dealing with the class imbalance that affects model results. Most news articles received zero clicks, so our models easily learned to predict "not clicked". This meant that even though accuracy looked high, the models weren't very good at finding the rare articles that actually got clicks. Oversampling the minority class (clicked articles) helped, but it was still tough to predict true engagement for special cases or very popular stories.

Another important takeaway was the value of error analysis. Looking closely at the articles our models got wrong for both false positives and false negatives, it helped us see patterns that weren't obvious from overall metrics alone. For example, we found that the models sometimes overpredicted engagement for generic stories with trending entities, and missed genuinely interesting articles in crowded categories like sports and news. This showed us that adding more meaningful features and possibly more training data in certain areas could make a bigger difference than just switching to a more complex algorithm.

Finally, working through this project made it clear how much time and effort real-world data science projects demand, especially in preparing data and interpreting results. It's not just about running models, it's about understanding the data, making thoughtful improvements, and learning from mistakes.

## VI. Statement of Work

| Yingying Sun | Aditya Tangirala | Matthew Vue |
|---|---|---|
| Data preprocessing & supervised learning | Data cleaning, final report editing, team discussion coordination | Data preprocessing & unsupervised learning |

# Works Cited

Fan, Wentao, et al. “Clustering-Based Online News Topic Detection and Tracking Through Hierarchical Bayesian Nonparametric Models.” *In Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, vol. SIGIR, no. 21, 2021, pp. 2126 - 2130. *Clustering-Based Online News Topic Detection and Tracking Through Hierarchical Bayesian Nonparametric Models*, https://dl.acm.org/doi/10.1145/3404835.3462982. Accessed 17 June 2025.

Media, Media Helping. “Dealing With Algorithmic Bias in News.” *Media Helping Media*, 12 Apr. 2025, mediahelpingmedia.org/advanced/dealing-with-algorithmic-bias-in-news/#:~:text=MHM:%20What%20is%20algorithm%20bias,of%20false%20or%20misleading%20information.

Microsoft. “Microsoft News Recommendation Dataset - Azure Open Datasets.” *Learn Microsoft*, 28 August 2024, https://learn.microsoft.com/en-us/azure/open-datasets/dataset-microsoft-news?tabs=azureml-opendatasets. Accessed 17 June 2025.

Microsoft. “MIND: MIcrosoft News Dataset.” *MIND: MIcrosoft News Dataset*, Microsoft, 2020, https://msnews.github.io/. Accessed 17 June 2025.

PATEL, VAIDEHI, and ARPITA PATEL. *Clustering News Articles for Topic Detection*, May 2018, www.irejournals.com/formatedpaper/1700671.pdf.

Svensson, Karin, and Johan Blad. “Exploring NMF and LDA Topic Models of Swedish News Articles.” *Publications from Uppsala University*, Uppsala University, 2020, https://uu.diva-portal.org/smash/record.jsf?pid=diva2:1512130&dswid=-2922. Accessed 17 June 2025.