Optimizing News Categorization on Imbalanced Data: A Hybrid Feature and Gradient Boosting Approach

Gianbattista Busonera, Paolo Malugani
Politecnico di Torino
Student ids: s353805 — s359857
s353805@studenti.polito.it — s359857@studenti.polito.it

***Abstract*—We propose a robust hybrid approach to multiclass news categorization, addressing high-dimensional text data and severe class imbalance. The proposed approach employs a hybrid feature extraction strategy that integrates structured metadata, statistical keyword selection via an improved Chi-squared test per label (ImpCHI), and semantic embeddings (Word2Vec). We benchmark Gradient Boosting Decision Trees (LightGBM) against a LinearSVC baseline. Our experiments demonstrate that LightGBM, optimized for class imbalance, significantly outperforms the baseline, achieving a Macro F1 score of 0.754 on a held-out test set and 0.743 on the public leaderboard.**

I. Problem Overview

This project aims to develop a machine learning model capable of performing a multiclass classification task based on the available features like `title`, `article` and associated metadata, `source`, `timestamp`, `page_rank`.

The goal of this project is to identify the category associated with each news article, specifically optimizing for the Macro F1 score. This metric is particularly challenging given the severe class imbalance (see Fig. 1), as it treats all categories equally, requiring the model to perform well even on minority classes like 'Health' (6). The dataset is divided into two parts:

- a *development* set, containing ≈ 80,000 news for which a label is provided;
- an *evaluation* set, containing 20,000 news.

We will need to use the development set to build a classification model to correctly label the points in the evaluation set. We can make some considerations based on the development set. First, significant class imbalance, where categories such as "International News" dominate, while others like "Health" are underrepresented. (see Fig. 1). Second, some `articles`, `sources` and `timestamps` are missing (see Fig. 2). Third, specific `articles` provide additional contextual metadata within the text body. Fourth, `Id` is not correlated with both the `timestamp` and the `label`, hence it has been dropped.

Data exploration reveals interesting patterns:

- `page rank` correlation: some labels are highly correlated with their page rank (see Fig. 4), even if it is highly skewed towards 5 (92%).
- `timestamp` distribution: suggests that some categories (i.e. International News, Business, Technology) are more present in certain years (Fig. 3).

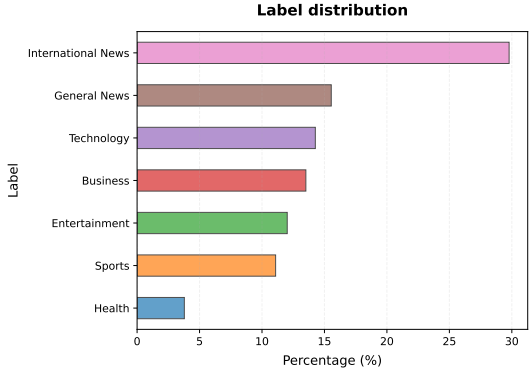

Fig. 1: Label distribution highlighting severe class imbalance.

Fig. 2: Missing values among different labels

- `source` distribution: indicating that some `sources` are highly specialized in fixed topics.

It is worth mentioning that, considering as a subset `source`, `title`, `article` and `label`, the dataset contains both duplicated rows and controversial rows.

To better understand the data at hand, we manually inspected the dataset, especially `title` and `article`. Titles are typically short and contain immediate and informative keywords. Articles may contain valuable metadata such as links `<href>`, images `<img>` with their description `<alt>`. That suggests we need to develop some approaches to extract the valuable insights from the text such as extracting the most valuable words and the semantic meaning of the sentences.

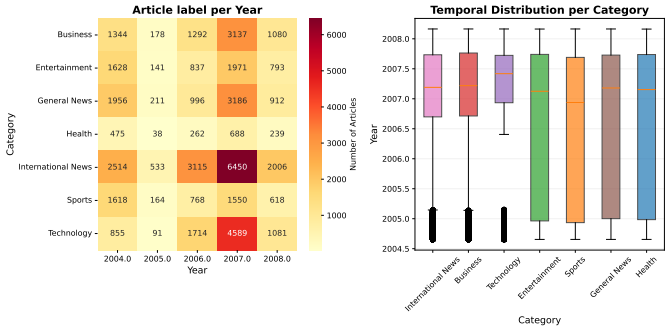

Fig. 3: Per class temporal distribution of news articles

PR	Count	%
2	43	0.05
3	2371	2.96
4	3692	4.62
5	73891	92.37

TABLE I: Page Rank distribution

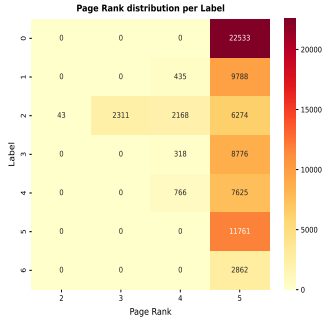

Fig. 4: Heatmap of Page Rank vs. Label

II. Proposed Approach

A. Preprocessing

1) Data Cleaning: The initial dataset contained 80,000 development samples. A rigorous data cleaning pipeline was implemented:

- **Duplicates:** We identified 1,368 exact duplicate rows (identical across all features and labels). These were reduced to a single instance to prevent data leakage. Furthermore, we found a small number of ambiguous cases where identical `source`, `article` and `title` text was associated with different `labels`. Consequently, these 2,969 conflicting records were excluded to ensure label consistency.
- **Missing Values:** Analysis revealed missing values primarily in the `article` and `source` fields, with a higher incidence in the "General News" and "Entertainment" categories. This pattern suggests these might be short, breaking news snippets where full text or source attribution is occasionally omitted. We imputed missing text values with an empty string.

After cleaning, the development set consisted of **75,660** unique and consistent records.

2) Data Splitting: To ensure reliable evaluation, the cleaned development set was split using stratified sampling to preserve the original class distribution: **80% Training** (60,528 samples), **10% Validation** (7,566 samples), and **10% Test** (7,566 samples). The validation set was used for hyperparameter tuning, while the test set provided a final, unbiased performance estimate.

3) Feature Engineering: Our feature engineering strategy aimed to capture information from structured metadata, textual statistics, and deep semantic content.

- **Timestamp Features:** For valid timestamps, we extracted `year`, `month`, `day`, `hour`, and `weekday`, otherwise flagged as `missing`. To model cyclical continuity (e.g., hour 23 is close to hour 0), we applied sine and cosine transformations to `hour` and `weekday`. A binary `is_weekend` indicator was introduced to capture publication patterns; for instance, sports-related content often exhibits higher frequency during weekends.
- **Text Statistics:** We computed simple yet informative scalar features: character and word counts for `title` and `article`, their ratio, average word length, and counts of specific HTML elements (e.g., `<img>` tags, hyperlinks) which can indicate article richness.
- **Source Target Encoding:** Directly one-hot encoding the `source` categorical variable would create high dimensionality. Instead, we used target encoding: for each source s, we computed the empirical probability $P(\text{label} = c_i | \text{source} = s)$ for each of the seven categories c_i, smoothed as follows:

$$\text{SourceFeature}_{c_i} = \frac{\text{count}(s, c_i) + 1}{\text{count}(s) + 7}$$

This yields a 7-dimensional dense vector per article representing the historical topic distribution of its publisher, a strong predictive signal. Whenever a source was missing, it was filled with 0s.

- **Agency news encoding**: we extracted them from the text since they are often inside parenthesis. We retained only top $K_{agency} = 40$ best agency news that appeared at least $min\_agency = 5$ times and according to the Chi-squared (χ^2) test for independence to avoid exploding in dimensionality. We hence applied constrained one hot encoding.

4) Hybrid Text Feature Extraction: Applying TF-IDF to the entire corpus resulted in a high-dimensional feature space (>100k features), which is computationally expensive and prone to overfitting. To reduce noise and dimensionality while preserving discriminative power, especially for minority classes, we adopted a hybrid approach that combines semantic embeddings with a refined statistical feature selection method.

Our initial approach for feature selection followed established literature: we considered using the Chi-squared (χ^2) test, a classical feature selection method for text classification introduced by Yang [1]. The χ^2 statistic measures the independence between a term's occurrence and a class label, with higher values indicating stronger dependence. However, as noted by Forman, in highly imbalanced datasets, traditional χ^2 feature selection suffers from a critical bias: it tends to select features predominantly associated with the majority classes [2]. This occurs because the global ranking of features by χ^2 score is dominated by terms that appear frequently in the prevalent categories, potentially discarding rare but highly

discriminative terms for minority classes (e.g., "vaccine" for Health or "blockchain" for Technology).

To overcome this limitation, we employed an **Improved Chi-squared (ImpCHI)** method [3]. Instead of selecting the top-K features globally based on their aggregate χ^2 scores, The Improved Chi-squared (ImpCHI) approach mitigates the majority-class bias by localizing feature selection. By selecting the top K features per label, we ensure that the specific lexicon of underrepresented categories is not overshadowed by the high-frequency generic terms. This results in a more 'democratic' feature space where each label contributes an equal number of dimensions to the final sparse representation. The union of these per-class selections forms our final vocabulary.

Our hybrid text representation pipeline therefore consists of three complementary components:

1) **Semantic Embeddings (Word2Vec):** To capture contextual meaning and synonymy, we trained a Word2Vec model (CBOW, window = 10, min_count = 10) on the corpus obtained by concatenating the `title` and `article` fields. The model was trained on all available text (development and evaluation sets), which is acceptable since Word2Vec is an unsupervised method, allowing it to better capture semantic relationships and enrich the vocabulary [4]. To ensure reproducibility, training was performed using a single core and with the `PYTHONHASHSEED` fixed, following the recommendations in the Gensim Word2Vec documentation.
2) **Text Preprocessing:** HTML tags were parsed to extract alt-text from images and URLs from links before removal. At a first glance, we decided to maintain only few parts of the link to remove noise. Initial experiments suggested that aggressive URL filtering removed predictive signals. Therefore, a minimal cleaning approach was adopted, preserving informative path segments (e.g., `rss/health`) and numbers, since some of them are highly predictive.
3) **Statistical Bag-of-Words with ImpCHI Selection:** We applied TF-IDF vectorization (unigrams and bigrams) separately to `title` and `article`, using the vocabulary selected by the ImpCHI method described above.

The final feature vector for each article is the concatenation of: metadata/scalar features ($\approx$ 30 dim), source probability vector (7 dim), agency news one-hot-encoding (40 dim), combined Word2Vec (75 dim), title TF-IDF-ImpCHI (sparse, $\approx$ 210 dim), and article TF-IDF-ImpCHI (sparse, $\approx$ 560 dim). The total dimensionality is thus approximately 1,200 features, which is computationally tractable while retaining discriminative power across all classes.

B. Model Selection and Training

The following algorithms have been tested:

- **Linear Baseline (LinearSVC):** This model requires feature scaling; therefore, we applied a StandardScaler with `with_mean=False`, which preserves sparsity and avoids explicit densification. It also assumes linear separability, which may not hold for complex text data.
- **Gradient Boosting Tree-Based Ensembles (LightGBM):** These models can natively handle sparse data and mixed feature types without the need for densification and standardization. They are inherently capable of modeling non-linear interactions and complex decision boundaries [5].

For both classifiers, the best-working configuration of hyperparameters has been identified through a parameter grid search, as explained in the following section.

C. Hyperparameter Optimization

There are two main sets of hyperparameters to be tuned:

- $K_{article}$, K_{title} and $K_{embeddings}$ for the preprocessing
- LightGBM and LinearSVM parameters

By assuming that the two are orthogonal, we can first select the preprocessing parameters and then move on to the classifiers ones.

We can use an 80/10/10 train/val/test split on the development set and run a parameter grid search on $K_{embedding}$, K_{title} and $K_{article}$ separately. To this end, we will train a LightGBM and a LinearSVC with their default configurations (only class_weight = 'balanced' and random state was applied to ensure reproducibility) and assess their performance based on the resulting Macro f_1-score.

TABLE II: Model Parameter Values

Model / Step	Parameter Values
Preprocessing	$K_{embedding} \in \{30, 50, 75, 100, 150, 300\}$ $K_{title} \in \{20, 30, 40, 50\}$ $K_{article} \in \{60, 70, 80, 90, 100\}$
LightGBM	objective = multiclass — num_class = 7 metric = multi_logloss — colsample_bytree: 0.4 subsample = 0.75 — subsample_freq = 7 max_depth $\in \{10, 12, 14, 16, 18, 20\}$ n_estimators $\in \{300, 400, 500, 600, 700\}$ num_leaves $\in \{31, 67, 127, 217, 255\}$ min_child_samples $\in \{31, 67, 127, 255\}$ reg_alpha $\in \{0, 0.15, 0.3, 0.45, 0.60\}$ reg_lambda $\in \{2, 4, 6, 8, 10\}$
LinearSVC	penalty = l2 — max_iter = 2500 loss = {square_hinge, hinge} $C \in \{0.0001, 0.001, 0.01, 0.1, 1, 2, 5\}$ tol $\in \{10^{-4}, 10^{-5}, 10^{-6}\}$

Once selected the preprocessing parameters, we can run a parameter grid search on both LinearSVC and LightGBM, based on the hyperparameters defined in Table II.

Parameter grid search was preferred due to the fact it is time consuming to run a Stratified GridSearchCV on a dataset with that many rows on a laptop.

III. Results

We proceeded in the following order, selecting the following preprocessing parameters: $K_{embedding} = 75$, fixing $K_{title} = 30$ and $K_{article} = 80$. Second, $K_{title} = 30$, fixing $K_{embedding} = 75$ and $K_{article} = 80$. Third, $K_{article} = 80$, fixing $K_{embedding} = 75$ and $K_{title} = 30$. All these results can be seen in Fig. 5

Even if $K_{embedding} = 75$ resulted in the best configuration, preliminary features importance analysis showed that these

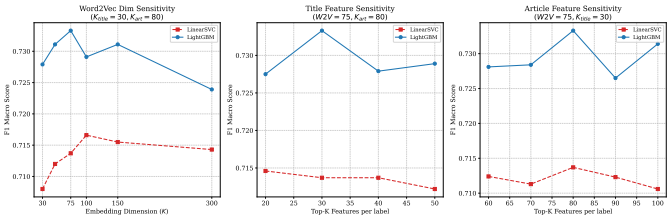

Fig. 5: Preprocessing hyperparameters tuning

TABLE III: Best hyperparameters and comparison

Model	**Hyperparameters**	**Macro f_1 (val)**	**Macro f_1 (test)**
LightGBM	max_depth: 14 learning_rate: 0.02 num_leaves: 217 min_child_samples: 67 reg_alpha: 0.15 reg_lambda: 4.0	0.7519	**0.7541**
LinearSVC	C: 5 loss: squared_hinge tol: 10^{-6}	0.7152	0.7181

dense components tended to dominate the decision trees, effectively marginalizing the specific keywords selected via ImpCHI. To achieve a more balanced hybrid feature space, we reduced the embedding size to **50 dimensions**. This reduction forced the model to leverage the high-precision categorical lexicon provided by the statistical features (ImpCHI), using Word2Vec primarily for context disambiguation rather than as the sole source of truth. We have subsequently increased the number of $K_{article}$ for classes having a niche and non-overlapping vocabulary by analyzing them to $K_{art,int} = 160$, $K_{art,tech} = 160$ and $K_{art,sport} = 240$.

The final model, trained on the combined training and validation sets with the optimal hyperparameters, achieved a **Macro F1 score** of 0.7541 on the held-out **test set**, demonstrating strong generalization capabilities and negligible overfitting. Notably, the proposed solution is computationally efficient: training on the entire development set and generating predictions for the evaluation set requires only 3 minutes.

Error Analysis: Inspection of the confusion matrix (Fig. 6) on the test set revealed that the most frequent confusion occurred between **"International News"** (class 0) and **"General News"** (class 5). This is semantically intuitive, as these categories share a significant portion of their vocabulary, making decision boundaries blurry for Bag-of-Words models. In contrast, well-defined categories like **"Sports"** and **"Technology"** achieved recall scores of 94% and 86%, demonstrating the model's effectiveness when feature spaces are distinct.

IV. Discussion

The proposed hybrid framework substantially outperforms the linear baseline and the public leaderboard benchmark (Macro F1 0.743 vs 0.443) achieving top-tier performance on the public leaderboard, validating the hypothesis that combining localized feature selection (ImpCHI) with semantic

Fig. 6: LightGBM Confusion matrix on test set.

embeddings effectively mitigates class imbalance. We have successfully shown that using ImpCHI [3] works robustly also for English datasets and proposed a hybrid combination of ImpCHI and Word2Vec.

The main limitation remains the semantic overlap between some news categories, which is an inherent ambiguity in the data. Future work could explore more advanced semantic models (e.g., BERT embeddings), hierarchical classification approaches to better disentangle broad categories like "International" and "General" news and outliers removal strategies. Furthermore, feature importance analysis confirmed the effectiveness of our hybrid strategy: while the Source Encoding and Word2Vec provided the strongest predictive signal, the ImpCHI-selected keywords were decisive for distinguishing minority classes, validating the need for localized feature selection over global methods.

In conclusion, our pipeline, centered on a hybrid feature set and a specifically tuned LightGBM classifier for class imbalance, provides a robust, efficient, and effective solution for the automated categorization of high-dimensional, imbalanced news data.

References

[1] Y. Yang and J. O. Pedersen, "A comparative study on feature selection in text categorization," in *International Conference on Machine Learning*, 1997.

[2] G. Forman, "An extensive empirical study of feature selection metrics for text classification [j]," *Journal of Machine Learning Research - JMLR*, vol. 3, 03 2003.

[3] S. Bahassine, A. Madani, M. Al-Sarem, and M. Kissi, "Feature selection using an improved chi-square for arabic text classification," *Journal of King Saud University - Computer and Information Sciences*, vol. 32, 05 2018.

[4] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," 2013.

[5] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "Lightgbm: A highly efficient gradient boosting decision tree," 12 2017.