# News Multi-class Classification Problem

Giovanni Casati
*Politecnico di Torino*

***Abstract*—In this report, we present a supervised learning approach to classify news articles into seven predefined categories. Our methodology involves a multi-stage pipeline that addresses complex textual noise and significant class imbalances. We implement a Hard Voting ensemble that combines Linear Support Vector Machine, Logistic Regression, and Multinomial Naive Bayes models.**

## I. Problem Overview

The objective is a supervised multi-class classification task within the online news domain. Beyond semantic content, the challenge requires integrating metadata signals, such as source authority (*PageRank*) and publication provenance, to classify articles into seven predefined categories: **International News (0), Business (1), Technology (2), Entertainment (3), Sports (4), General News (5), and Health (6)**. The dataset contains approximately 100,000 instances, partitioned as follows:

- **Development Set:** 79,997 labeled articles used for training and validation.
- **Evaluation Set:** 20,000 unlabeled articles for final prediction.

A primary challenge lies in the thematic overlap between categories, such as *International News* and *General News*, requiring the model to discern primary topics amidst semantic ambiguity. Furthermore, the significant class imbalance observed in the development set requires a robust strategy to ensure high performance across all labels:

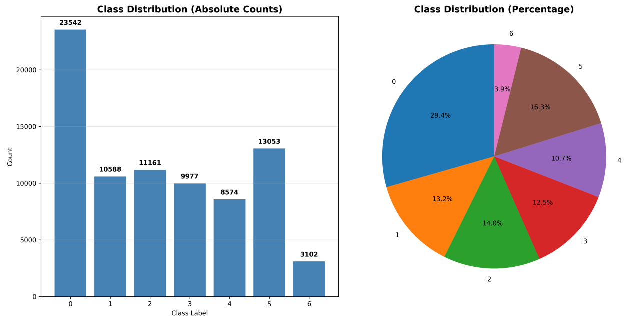

Fig. 1. Class imbalance

The development set exhibits a significant class imbalance that poses a direct risk of statistically biased predictions. As shown in the distribution analysis, Label 0 (International News) dominates the corpus, comprising nearly a third of all instances. In contrast, Label 6 (Health) represents the extreme minority. The resulting imbalance ratio of roughly $7.6 : 1$ between the majority and minority classes necessitates specific algorithmic adjustments to prevent the model from defaulting to the majority class in cases of high uncertainty.

The dataset reveals several critical integrity challenges that necessitate a specialized preprocessing strategy:

- **Complex Textual Noise:** Raw `title` and `article` fields are saturated with non-semantic elements, including HTML markup (e.g., `<p>`, `<a>`), raw URLs, and database artifacts such as `\\N`.
- **Embedded Semantic Loss:** Many structural tags encapsulate high-value information within attributes, such as `alt` image descriptions or keywords within `href` links that must be preserved during stripping to maintain context.
- **Metadata Inconsistency:** The `source` and `timestamp` attributes contain invalid placeholders (e.g., `0000-00-00 00:00:00`), requiring explicit encoding of missing information to facilitate consistent handling.
- **Label Ambiguity:** Duplicate records with identical content (`source`, `title`, and `article`) are occasionally mapped to conflicting categories, introducing deterministic noise into the training signal.

## II. Proposed Approach

### A. Preprocessing

Given the unstructured nature of the news articles and the presence of heterogeneous metadata, the procedure is divided into three main stages: data sanitation and flexible cleaning, feature engineering with strictly aggressive normalization, and metadata processing.

*1) Data Sanitation and Flexible Cleaning:* The preliminary stage of our pipeline focuses on structural integrity and discretionary text refinement. First, we addressed the `source` and `timestamp` fields, which contained numerous invalid entries and placeholder values, such as ”0000-00-00 00:00:00”. These inconsistencies were systematically converted to null representations (`NaN`) to facilitate consistent handling during subsequent imputation.

Concurrently, a specialized parsing procedure was applied to the `title` and `article` fields. At this stage, text cleaning is non-destructive and governed by hyperparameters, to decide which structural artifacts to preserve. This includes stripping HTML syntax while extracting meaningful text from `alt` and `title` attributes, and decomposing URLs into descriptive tokens rather than removing them. To address label ambiguity, we implemented a majority voting deduplication strategy for identical document groups, discarding tied records to eliminate deterministic noise from the training set.

*2) Feature Engineering and Strict Aggressive Normalization:* To provide the model with a statistical overview of the documents, we synthesized a `full_text` feature by concatenating the `title` and `article`. Before vectorization, we extracted structural meta-features:

- **Quantitative Descriptors:** Total character counts for the combined text, as well as independent lengths for `title` and `article`.
- **Digit Density:** The ratio of numerical digits to total document length, serving as a proxy for identifying data-heavy content.

Following this extraction, the text underwent a second, strictly aggressive normalization pass. Unlike the first pass, this step is not discretionary; it is functionally bound by the requirements of the TF-IDF vectorizer to maximize lexical consistency. We applied NFKD Unicode normalization to strip accents, removed all non-alphanumeric characters, and enforced a lowercase conversion. Finally, words shorter than three characters were filtered out to strictly minimize vocabulary sparsity and noise.

*3) Metadata Processing and Scaling:* The `source` and `timestamp` attributes were processed to capture categorical and temporal biases. `Source` was handled via a `OneHotEncoder` with a frequency threshold of 4, mapping infrequent or unknown publishers to a single category to prevent overfitting. The `timestamp` was decomposed into components like *year* and *weekday*.

All numerical features, including the low-cardinality `page_rank` (4 unique values) and extracted meta-features, were normalized using a `MinMaxScaler`. This maps all structural data to a $[0, 1]$ range, ensuring compatibility with the `MultinomialNB` estimator's non-negativity (see Model selection) and preventing high-magnitude character counts from numerically dominating the linear classifiers.

### *B. Model selection*

Given the high dimensionality and sparsity of the feature space resulting from the TF-IDF vectorization (approximately 50,000 features), we focused our model selection on algorithms known for their efficiency and robustness in text classification tasks. Instead of relying on a single estimator, we adopted an Ensemble Learning strategy to improve generalization performance and reduce the variance associated with individual models.

The Voting Ensemble Strategy aggregates the predictions of multiple base models to determine the final class. We opted for a "Hard Voting" mechanism, where the final class label is determined by the majority vote of the constituent classifiers.[1]

- **Linear Support Vector Machine (LinearSVC)**:
  Selected as the primary component due to its capability to maximize the decision margin in high-dimensional spaces. SVMs are mathematically well-suited for sparse data with a high number of features. LinearSVC focuses on the decision boundary seeking to find the hyperplane that maximizes the margin (distance) between the hyperplane and the nearest data points (support vectors) of distinct classes. Since in our case $\#samples < \#features$, the primal form is more efficient so we set `dual = False`.[2]
- **Logistic Regression**:
  This estimator provides a robust probabilistic baseline. For a multi-class problem, the implementation uses the Multinomial (Softmax) formulation, directly modeling the probability of class membership. We chose the Solver "saga" that has fast convergence even for high-dimensional data. The model minimizes the Cross-Entropy Loss function that is modified weighting the contribution of each sample. If "Health" articles are ten times rarer than "Sports", an error on a "Health" article generates a gradient ten times larger, forcing the optimizer to pay attention to the minority class. This is directly aligned with the Macro F1 objective, as it helps in balancing the decision boundary, particularly in cases where classes are not linearly separable with a wide margin.
- **Multinomial Naive Bayes**:
  Chosen for its computational efficiency and theoretical foundation in counting-based feature occurrences. Despite its strong independence assumptions, Multinomial NB is highly effective for document classification and acts as a regularizer within the ensemble, preventing overfitting to complex decision boundaries that might be learned by the SVM.[3]

### *C. Hyperparameters tuning*

Due to the high dimensionality of the feature space and the limited computational resources available, performing an exhaustive grid search over the entire pipeline was not feasible. Consequently, we adopted a decoupled optimization approach, operating under the assumption that the feature extraction configuration (mainly the TF-IDF parameters) influences model performance independently from the classifiers' regularization settings.

We fixed the TF-IDF vectorization parameters based on empirical heuristics while restricting the *Grid Search* scope to the estimators' core hyperparameters. The tuning of TF-IDF parameters was done afterwards. This search utilized a 3-fold cross-validation strategy to optimize separately:

- **Linear Support Vector Machine**: We tuned the inverse regularization strength $C$ within the range $[0.01, 1.0]$. This range tests varying levels : smaller values are investigated to prioritize a wider margin, which is a common strategy to prevent overfitting in sparse feature spaces where the number of features exceeds the number of samples.
- **Logistic Regression**: The inverse regularization strength was probed across the range $[0.1, 2.0]$. We tested this range to find the point where the model effectively calibrates class probabilities. The range allows us to

observe how different levels of penalty affect the model's ability to focus on minority classes.
- **Multinomial Naive Bayes**: We tested the Laplace smoothing factor $\alpha$ within the range $[0.01, 1.0]$. This range is necessary to investigate how much smoothing is required to mitigate zero-probability issues for rare feature occurrences without diluting the discriminative signal of the TF-IDF features.

The figures below report the mean values and the ranges of the evaluation metric across different hyperparameter configurations.

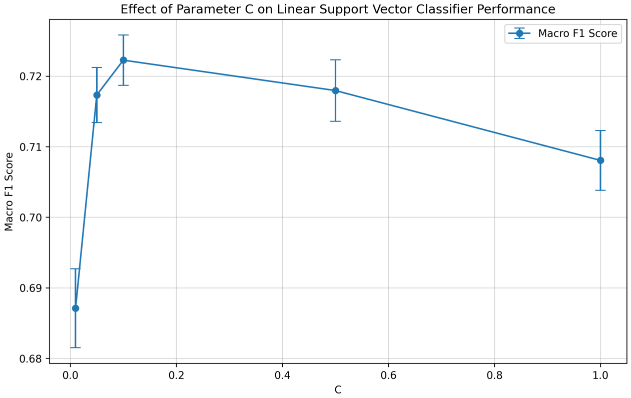

Fig. 2. Linear SVC: macro-f1 w.r.t. C

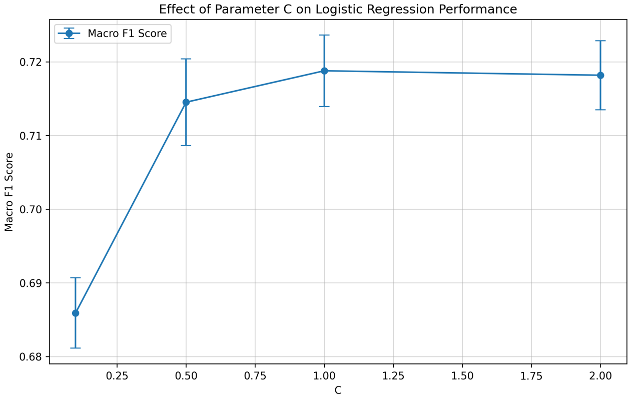

Fig. 3. Logistic Regression: macro-f1 w.r.t. C

## III. Results

The analysis of the performance trends yielded the following optimal values:

- Linear SVC: $C = 0.1$
- Logistic Regression: $C = 1.0$
- Multinomial Naive Bayes: $\alpha = 0.1$

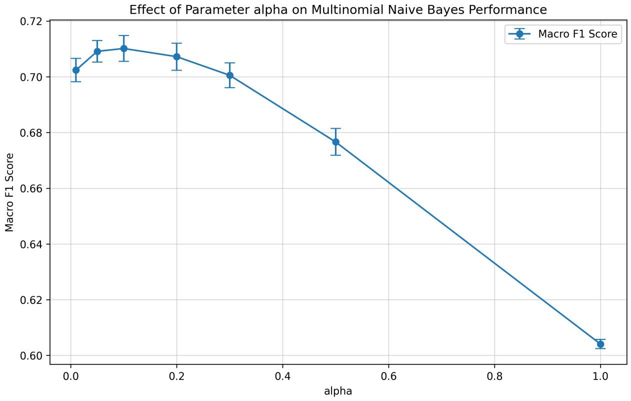

Fig. 4. Multinomial Naive Bayes: macro-f1 w.r.t. $\alpha$

The graphical analysis of the cross-validation results provided critical insights into the hyperparameter dynamics of each estimator.

The **Linear SVC** achieved peak performance at a relatively low regularization parameter ($C = 0.1$), validating the hypothesis that a wider margin is essential for managing outliers and preventing overfitting in high-dimensional, sparse feature spaces.

Conversely, **Logistic Regression** exhibited optimal stability at a higher value ($C = 1.0$), where it could effectively model class probabilities without excessive penalty.

Furthermore, the **Multinomial Naive Bayes** performance was highly sensitive to the additive smoothing parameter $\alpha$. The tuning curve identified an optimal peak at $\alpha = 0.1$, indicating a precise requirement for Laplace smoothing: while necessary to prevent zero-probability issues for out-of-vocabulary terms, excessive smoothing values were found to dilute the discriminative signal of the sparse TF-IDF features.

These findings guided the final calibration of the hard-voting ensemble, ensuring that each model contributed from its most stable numerical state.

Following the automated tuning of individual estimators, we performed a manual refinement of the `VotingClassifier` configuration. Unexpectedly, this heuristic adjustment yielded superior results compared to the combination of individually optimized "best" parameters. This phenomenon can be attributed to the complementary error distributions of the ensemble members. In a "hard" voting scheme, the objective is not to maximize the isolated accuracy of each model, but to ensure that their respective classification errors are uncorrelated. By slightly deviating from the individual optima, we achieved a more robust collective balance. The final selected configuration for the ensemble achieved an evaluation **macro F1-score of 0.732**, obtained with the following configuration:

- Linear SVC:
  `class_weight=balanced, dual=False, C=0.1`
- Logistic Regression:
  `class_weight=balanced, solver=saga,`

`C=1.6`

- Multinomial NB:
  `alpha=0.5`

## IV. Discussion

The evaluation metric indicates a promising performance, far greater than the baseline provided. A close inspection of the Confusion Matrix generated in internal evaluation reveals one specific area of residual ambiguity: the model exhibits a persistent misclassification pattern between Class 0 and Class 5. The overlapping feature space between these two categories tells that semantic content (keywords) alone and the metadata extracted are insufficient to fully discriminate between them, due to the shared vocabulary of such similar topics.

We propose extending the feature space beyond lexical content by incorporating readability and syntactic complexity metrics. Since the TF-IDF approach relies on vocabulary overlap, it struggles when two categories share similar keywords. However, we hypothesize that distinct classes may exhibit subtle but statistically significant differences in their writing style and structural complexity, which can be imperceptible to qualitative analysis. Metrics such as the Flesch-Kincaid Grade Level and the SMOG Index could be explored to quantify the reading difficulty, while Lexical Diversity (measured via Type-Token Ratio - TTR)[4] could assess the richness of the vocabulary.

## References

[1] Dietterich, TG. (2000). Ensemble methods in machine learning. Multiple Classifier Systems: First International Workshop, MCS 2000, Lecture Notes in Computer Science. 1-15.

[2] Joachims, Thorsten. (1998). Text Categorization with Support Vector Machines. Proc. European Conf. Machine Learning (ECML'98). 10.17877/DE290R-5097.

[3] F. Pedregosa et al., ”Scikit-learn: Machine Learning in Python,” Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011.

[4] Majumdar, Partha. (2025). Type-Token Ratio (TTR) - A Measure of Lexical Diversity. 10.5281/zenodo.15770103.