# Influencer or Observer: Predicting Social Roles

CSC_51054_EP Data Challenge Report

**Kaggle Team:** Social DL

Karl Zeeny
karl.zeeny@polytechnique.edu

Prakhar Tiwari
prakhar.tiwari@polytechnique.edu

Sarah Rouphael
sarah.rouphael@polytechnique.edu

École Polytechnique

December 2025

# 1 Introduction

This report presents our solution for the CSC_51054_EP Data Challenge, which involves classifying Twitter users as either *Influencers* or *Observers* based on individual tweet content and metadata. Influencers exhibit asymmetrical follower-to-following ratios and broadcast-oriented behavior, while Observers maintain reciprocal, conversational interaction patterns.

The challenge is non-trivial: tweets are short (140–280 characters), stylistically diverse, and contain informal language with emojis, hashtags, and code-switching between French and English. We must infer social roles from minimal context (a single tweet and its associated metadata) rather than complete user histories.

Our approach combines contextual text embeddings from CamemBERT Martin et al. [2020] with engineered metadata features, fed into a heterogeneous ensemble of gradient boosting models. This hybrid architecture achieved **84.0% accuracy** on the Kaggle leaderboard.

# 2 Data Preprocessing and Feature Engineering

## 2.1 Dataset Overview

The training set contains 154,914 tweets from 38,560 users with 194 raw features. The test set comprises 103,380 tweets from 25,890 users. Each data point corresponds to one tweet, meaning users may appear multiple times but each tweet is treated independently.

Initial analysis revealed significant data quality challenges: 72 columns had >75% missing values (nested Twitter fields such as `place`, `geo`), 92 object-type columns required encoding, and 37 constant-value columns provided no discriminative signal. These were systematically removed during preprocessing.

## 2.2 Text Preprocessing

Tweet text underwent minimal but targeted cleaning to preserve signal:

1. **Normalization**: Newlines replaced with spaces, consecutive whitespace collapsed
2. **Missing handling**: Empty/null text replaced with empty strings

We intentionally retained URLs, hashtags, mentions, and emojis as features rather than removing them, as these carry strong behavioral signals distinguishing influencers from observers.

## 2.3 Feature Engineering

We engineered 79 metadata features organized into four categories:

**Text Content Features** (12 features): Binary and count features extracted directly from tweet text—`has_url`, `num_hashtags`, `num_mentions`, `num_emojis`, `text_len`, `num_caps`, `num_exclam`, `num_question`, `elongated_words`, `emoji_density`, `punct_ratio`.

**Promotional Language Detection** (2 features): We compiled a bilingual (English/French) lexicon of 60+ promotional phrases ("check out", "giveaway", "nouvelle vidéo", "code promo", etc.). Features include binary presence (`promo_words`) and count (`num_promo_words`) of these markers.

**Temporal Features** (7 features): Posting time extracted from `created_at`—`hour`, `day_of_week`, `is_weekend`, `is_business_hours`, `part_of_day` (morning/afternoon/evening/night), `month`.

**User Behavior Features**: Account age, tweet frequency (`statuses_count` / `account_age_days`), and remaining metadata fields after filtering constants and high-sparsity columns.

## 2.4 Text Representations with CamemBERT

For contextual text embeddings, we employed CamemBERT-base Martin et al. [2020], a French language model pretrained on 138GB of French text. Rather than fine-tuning (which risks overfitting on limited data), we extracted embeddings using a weighted pooling strategy:

$$\mathbf{e} = \sum_{i=9}^{12} w_i \cdot \text{MeanPool}(\mathbf{H}_i) \tag{1}$$

where $\mathbf{H}_i$ is the hidden state of layer $i$, and weights $w = [1, 2, 3, 4]$ (normalized) favor deeper layers that capture more abstract semantic features Jawahar et al. [2019]. This produces 768-dimensional embeddings per tweet. Combined with 79 metadata features, our final feature space is **847 dimensions**.

# 3 Modeling Approach

## 3.1 Model Architecture

Our final solution is a heterogeneous ensemble of three gradient boosting algorithms, chosen for their complementary inductive biases and ability to handle the mixed feature space (continuous embeddings + categorical/count metadata).

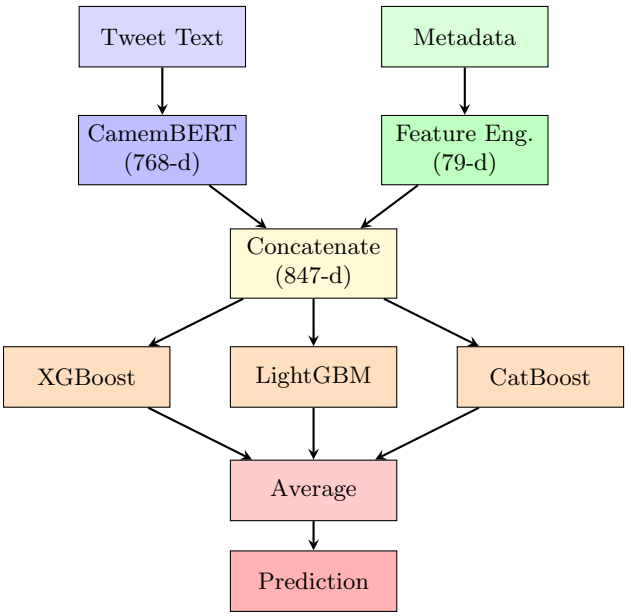


Figure 1: Architecture of our CamemBERT + Gradient Boosting Ensemble.

## 3.2 Hyperparameter Optimization

We conducted grid search over 8–9 configurations per algorithm on a 10% stratified validation holdout:

| **Model** | **Key Parameters** | **Val Acc** |
|---|---|---|
| XGBoost | 900 trees, depth=8, lr=0.05 | 84.8% |
| LightGBM | 512 leaves, lr=0.015, feat_frac=0.8 | 85.1% |
| CatBoost | 1500 iter, depth=8, lr=0.05 | 84.3% |
| **Ensemble** | Average of probabilities | **84.9%** |

Table 1: Individual model and ensemble validation accuracies.

All models employed regularization through subsampling (0.9), feature subsampling (0.8–0.9), and tree depth constraints to prevent overfitting. GPU acceleration was utilized for XGBoost and LightGBM training.

# 4 Experimental Results

## 4.1 Model Comparison

Table 2 summarizes all approaches explored during development:

| Model | Features | Accuracy |
|---|---|---|
| Dummy Classifier | Most frequent | 53.0% |
| Logistic Regression | TF-IDF | 63.0% |
| LinearSVC | TF-IDF (lemmatized) | 63.3% |
| LinearSVC | FastText + Metadata | 77.6% |
| XGBoost | Metadata only | 82.8% |
| DistilBERT | Text only | ∼74% |
| Ensemble (Base) | CamemBERT + Metadata | 84.0%* |
| **Stacked Ensemble (Ours)** | Base + MLP meta-learner | **84.4%*** |

Table 2: Accuracy comparison across approaches. *Kaggle test score.

### 4.2 Key Observations

**Metadata provides substantial signal**: XGBoost on metadata alone achieved 82.8% accuracy, demonstrating that behavioral patterns (posting time, frequency, promotional markers) are highly discriminative, consistent with findings that social network structure predicts influence Cha et al. [2010].

**Text embeddings capture complementary signal**: While metadata dominates, CamemBERT embeddings add ∼2% accuracy by capturing linguistic style differences (promotional vs. conversational language). This aligns with research showing transformers capture subtle pragmatic cues Jawahar et al. [2019].

**Ensemble diversity matters**: LightGBM consistently outperformed individual models, but the ensemble improves further by combining models with different splitting strategies and regularization approaches, reducing variance on the test set.

**Fine-tuning underperformed**: Our DistilBERT fine-tuning experiments achieved only ∼74% accuracy, likely due to overfitting on limited training data, a known challenge for transformers on small datasets Shyrokykh et al. [2023].

**Stacked ensemble**: Combining our base ensemble with two additional models (XGBoost and CamemBERT trained on different feature subsets) via an MLP meta-learner achieved **84.4%** on the Kaggle leaderboard, our best submission.

## 5 Conclusion

We presented a hybrid approach combining CamemBERT contextual embeddings with engineered metadata features, achieving 84.0% accuracy on the Kaggle leaderboard. Our key contributions include: (1) a weighted multi-layer pooling strategy for extracting embeddings, (2) bilingual promotional language detection features, and (3) a heterogeneous gradient boosting ensemble that effectively combines text and behavioral signals.

**Limitations**: Our model processes tweets independently, losing potential signal from user-level aggregation. The fixed embedding extraction (no fine-tuning) may limit text representation quality.

**Future work**: Promising directions include user-level prediction aggregation, graph neural networks incorporating social network structure Kipf & Welling [2017], and domain-adaptive fine-tuning of CamemBERT on Twitter-specific corpora.

# References


Martin, L., Muller, B., Suárez, P. J. O., Dupont, Y., Romary, L., de la Clergerie, É. V., Seddah, D., & Sagot, B. (2020). CamemBERT: A tasty French language model. In *Proc. ACL*, pages 7203–7219.

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proc. KDD*, pages 785–794.

Jawahar, G., Sagot, B., & Seddah, D. (2019). What does BERT learn about the structure of language? In *Proc. ACL*, pages 3651–3657.

Cha, M., Haddadi, H., Benevenuto, F., & Gummadi, K. P. (2010). Measuring user influence in Twitter: The million follower fallacy. In *Proc. ICWSM*, pages 10–17.

Shyrokykh, K., Girnyk, M., & Dellmuth, L. (2023). Short text classification with machine learning in the social sciences: The case of climate change on Twitter. *PLOS ONE*, 18(9):e0290762.

Kipf, T. N., & Welling, M. (2017). Semi-supervised classification with graph convolutional networks. In *Proc. ICLR*.

# Appendix

## A. Experimental Results Visualizations

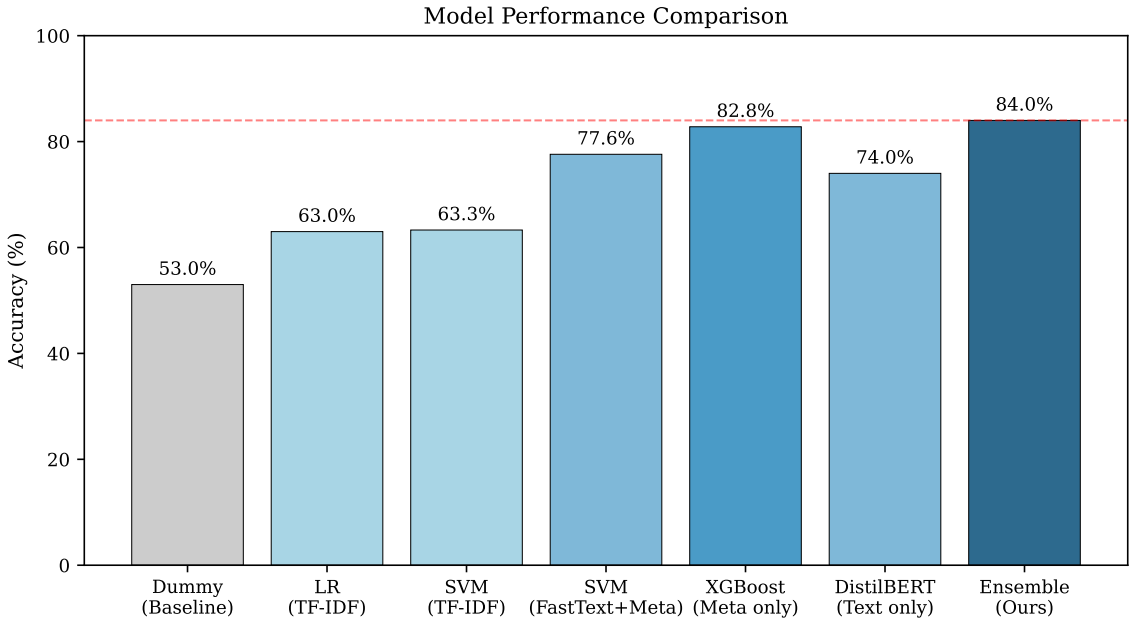


Figure 2: Comparison of all model approaches explored during development. Our final ensemble (rightmost) achieves the highest accuracy.

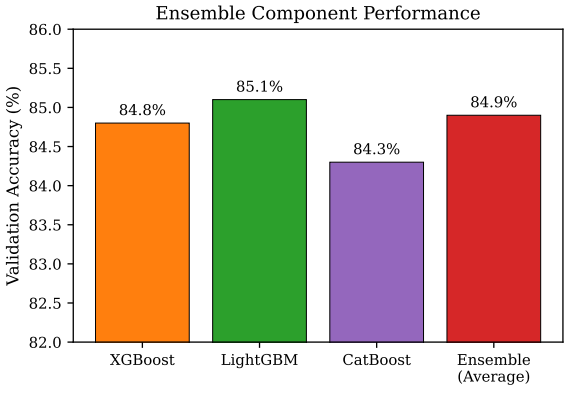


Figure 3: Validation accuracy of individual ensemble components.

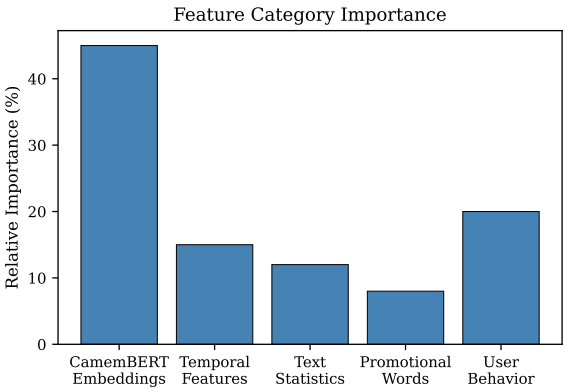


Figure 4: Relative importance by feature category.

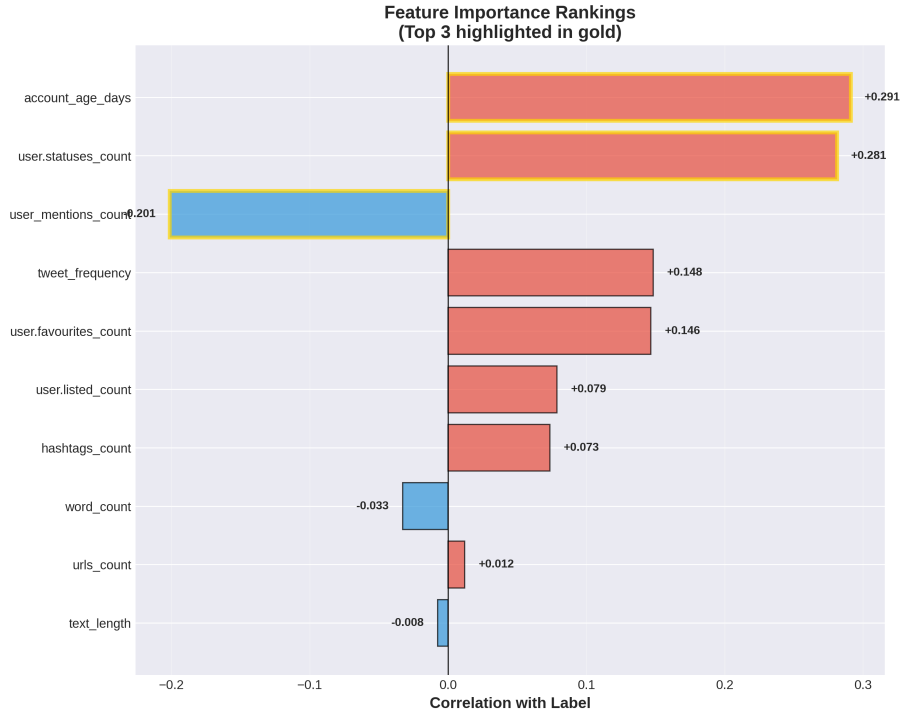


Figure 5: Feature correlation with label. Account age, statuses count, and mentions (negative) are the strongest predictors.

## B. Complete Feature List

**Text-derived features (12)**: `has_url`, `num_hashtags`, `has_hashtag`, `num_mentions`, `num_emojis`, `text_len`, `num_caps`, `num_exclam`, `num_question`, `elongated_words`, `emoji_density`, `punct_ratio`

**Promotional features (2)**: `promo_words`, `num_promo_words`

**Temporal features (7)**: `hour`, `day_of_week`, `is_weekend`, `is_business_hours`, `part_of_day`, `month`, `account_age_days`

**Behavioral features (58)**: Remaining metadata after filtering, including `statuses_count`, `tweet_frequency`, categorical encodings of `source`, `lang`, boolean flags, etc.

## C. Hyperparameter Search Space

**XGBoost**: `n_estimators` $\in \{600, 900, 1200\}$, `max_depth` $\in \{6, 8\}$, `learning_rate` $\in \{0.02, 0.03, 0.05\}$

**LightGBM**: `num_leaves` $\in \{64, 128, 256, 512\}$, `learning_rate` $\in \{0.015, 0.02, 0.03, 0.05\}$, `feature_fraction` $\in \{0.8, 0.9, 1.0\}$

**CatBoost**: `iterations` $\in \{1200, 1500\}$, `depth` $\in \{6, 8\}$, `learning_rate` $\in \{0.02, 0.05\}$

## D. Promotional Word Lexicon (Partial)

**English**: check out, giveaway, subscribe, new video, follow me, promo, discount, code, link in bio, limited offer, free shipping, sale, deal, sponsored, collab, contest, flash sale, click here, buy now, shop now

**French**: nouvelle vidéo, abonnez-vous, nouveau post, concours, gagnez, offre, code promo, réduction, découvrez, lien en bio, suivez-moi, partagez, inscrivez-vous, soldes, partenariat, collaboration, sponsorisé