# Machine and Deep Learning Data Challenge
# Influencer or Observer: Predicting Social Roles

Team: Gameover

Avrile Floro
Institut Polytechnique de Paris
avrile.floro@polytechnique.edu

Falguny Barua Ema
Institut Polytechnique de Paris
falguny.barua-ema@polytechnique.edu

Saurabh Mishra
Institut Polytechnique de Paris
saurabh.mishra@polytechnique.edu

December 2025

### Abstract

We present a stacking ensemble approach for predicting Twitter user social roles (influencer versus observer), as part of our final report for the CSC_51054_EP's Deep Learning Data Challenge. Our method combines: (1) LightGBM for prediction on structured tabular features, (2) fine-tuned CamemBERTav2 for textual analysis using a *user card* representation, and finally (3) HistGradientBoosting as a meta-learner. Leveraging this multi-modal approach, we achieve 87.7% accuracy on out-of-fold validation. Both behavioral features and language understanding are important and complementary for the classification of social roles on social networks, such as Twitter.

# 1 Introduction

The distinction between influencers and observers on social media reflects different communication strategies: influencers create original content for large audiences, while observers engage in conversations and maintain balanced interactions. The classification task presents the following challenges: individual tweets provide limited signal, users exhibit mixed behaviors, and the dataset structure requires careful preprocessing.

Our methodology uses a pipeline composed of three stages:

1. LightGBM extracts patterns from 1,373 user-level features that were derived from aggregating 343 tweet-level features
2. CamemBERTav2 captures linguistic signatures through fine-tuned French language understanding
3. HistGradientBoosting combines base model predictions as a meta-learner

# 2 Data Preprocessing and Feature Engineering

## 2.1 User Reconstruction via `user.created_at`

The training dataset contains 154,914 tweets from 30,696 users, according to our assumptions about user identification. The distribution is highly concentrated, with 98.65% of users having exactly 5 tweets. A fundamental design decision was to operate at the user level rather than at the individual tweet level, as research demonstrates significant benefits from user-level aggregation for population-level predictions using Twitter data [Giorgi et al., 2018]. Although no explicit user ID is provided, we use the `user.created_at` timestamp as a unique identifier, as it remains constant across all tweets from the same user. This enables: (1) aggregation of behavioral features across multiple tweets per user, (2) construction of comprehensive user profiles, and (3) proper stratified cross-validation at the user level to prevent data leakage.

## 2.2 Transductive Feature Engineering

We partially addressed the missing engagement metrics (`followers_count`, `friends_count`) by bridging the train and test sets in a transductive manner. The key lies in the `quoted_status` metadata: by correlating timestamps with follower history, we could reconstruct missing attributes for users appearing across splits. This matching process effectively allows us to recover some high-value features.

## 2.3 Feature Engineering & User-Level Aggregation

Our pipeline creates 343 features per tweet, covering temporal activity, content structure and profile metadata. To lower signal volatility, we consolidated these metrics at the user level (computing mean, max, min and standard deviation) which results in 1,373 tabular features (see Table 2 for the breakdown).

# 3 Model Architecture

## 3.1 Stage 1: LightGBM for Structured Features

LightGBM was selected for its ability to model non-linear interactions in high-dimensional tabular data while being computationally efficient [Ke et al., 2017]. Trained on the 1,373 aggregated user-level features, it provides a strong first-stage predictor that captures the behavioral patterns of influencers and observers. The model is configured with 1,500 trees, a learning rate of 0.03, depth 7, 31 leaves, L1/L2 regularization (0.1/1.0), and row/column subsampling of 0.8, in order to help control overfitting.

To avoid user-level leakage, we perform 5-fold stratified cross-validation using the user identifier as the grouping variable. We are thus sure that all tweets from the same user appear in the same fold. Under this setup, LightGBM achieves an out-of-fold accuracy of 87.1%. Final predictions are obtained by averaging probabilities across folds and assigning the user-level score to all tweets associated with that same user.

## 3.2 Stage 2: Textual Analysis Strategy with CamemBERTav2

### 3.2.1 User Profile Construction via Multi-View Cards

The role classification of users on social networks is difficult because of data sparsity: tweets, considered individually, are short texts, noisy, and often present limited lexical signal [Hong and Davison, 2010]. To overcome this challenge, we adopted a user-centric approach instead of staying at the tweet level. Aggregating textual content enables us to build a dense representation of the users [Rangel et al., 2013]. This concatenation helps in smoothing the stylistic and thematic variations encountered at the tweet level. The model is trained with a contextual window that is more representative of the user's personality as a whole. It mirrors the way humans form their opinions about other users' profiles.

Our user cards approach enriches raw text with profile and behavioral metadata that are directly integrated into the prompt when available, leveraging the advantage of learning from both structured and unstructured data [Yin et al., 2020]. Information such as biography, account age, followers count, friends count and interaction ratios (e.g., original content versus replies; see Table 5) are highly discriminative in social network classification tasks [Benevenuto et al., 2010]. Because Transformer models like BERT are limited to 512 tokens [Devlin et al., 2018], we use a multi-card generation procedure: users with rich histories (>5 tweets) are represented through two cards sampling different subsets of tweets. This acts as a light data augmentation mechanism [Wei and Zou, 2019] and enables inference-time ensemble averaging across cards, making predictions more robust and with lower variance.

### 3.2.2 Technical Implementation (CamemBERTav2)

For French tweet content, we used CamemBERTa-v2 [Antoun et al., 2024], a state-of-the-art French language model pretrained on 275B tokens drawn from a wide range of web sources, including social networks. We first compared eight French and multilingual encoders (Figure 1) and found that CamemBERTa-v2-base offered the best user-level accuracy at epoch 2. We then ran a hyperparameter search with Optuna (30 trials; Table 3). This search was carried out before later improvements to the model, however the corresponding hyperparameters were kept. We therefore fixed the learning rate to $2.32 \times 10^{-5}$, set the batch size to 48, and used a weight decay of 0.019. The model was trained for 2 epochs with label smoothing (0.005), a linear scheduler, and a 10% warmup ratio. With this setup, we achieved an out-of-fold accuracy of 87.5%.

## 3.3 Stage 3: Meta-Model Stacking

We use the HistGradientBoostingClassifier (HGBoost) from scikit-learn [Pedregosa et al., 2011] as the meta-learner to integrate the multi-modal signals generated in the previous stages. This stacking architecture follows established ensemble learning principles, where a meta-learner is trained on the outputs of the base models and improves generalization beyond what each model can achieve individually [Wolpert, 1992, Breiman, 1996].

The model is trained on features combining:

- base model probabilities, i.e. user-level predictions from LightGBM and CamemBERTav2
- the 1,373 aggregated user-level structural features used in Stage 1
- additional meta-features such as model disagreement metrics and per-user probability statistics (mean, max, standard deviation)

Other gradient boosting models such as CatBoost or LightGBM achieved lower or similar accuracy. We selected HistGradientBoosting for its robustness and stability. We use a learning rate of 0.03, a maximum depth of 6, L2 regularization of 0.5, and up to 800 iterations with early stopping. Adding CamemBERTa-v2 embeddings on top of these features did not improve accuracy and was therefore omitted. The same validation strategy was applied using `StratifiedGroupKFold` with `user_key` as the grouping variable to ensure that no user appears in both training and validation splits.

## 4 Results and Discussion

Table 1: Model Performance Comparison

| **Model** | **OOF Accuracy** | **Kaggle Score (public LB)** |
|---|---|---|
| Baseline (Logistic Regression) | — | 63% |
| Baseline (Dummy Classifier) | — | 53% |
| LightGBM (user-level) | 87.1% | 87.3% |
| CamemBERTav2 (user-level) | 87.5% | 87.7% |
| Meta-Model (HGB stacking) | 87.7% | 88.0% |

These results show that the stacked model surpasses both its constituent components. The performance gap shows synergy: the two models capture distinct and mutually reinforcing signals.

## 5 Conclusion and Future Work

Our results show that user-level aggregation is crucial for the classification of users on Twitter as observers or influencers, confirming prior observations on the superiority of modeling users rather than isolated messages [Giorgi et al., 2018]. Behavioral metadata captured by LightGBM and linguistic cues extracted by CamemBERTav2 prove highly complementary, and HGBoost, as the meta-model, effectively integrates both modalities. Behavioral patterns, such as account age, original-content ratios and reply dynamics (Figures 3–2; Table 4), have discriminative power that can be combined with the advantages of French-specific language models [Martin et al., 2020, Antoun et al., 2024]. The multi-card representation further improves robustness by capturing diverse facets of the same user's behavior. This methodology extends naturally to other user-classification problems on social platforms, and future work may incorporate graph-based features to better encode interaction structures.

# A Appendices

## A.1 Complete Feature Breakdown

Table 2 provides the complete distribution of 343 engineered features by category.

Table 2: Feature distribution across categories

| Category | Count | Examples |
|---|---|---|
| Temporal | 8 | account age (days, log), hour, weekday, weekend flag |
| Text Content | 4 | length, #hashtags, @mentions, uppercase ratio |
| User Engagement | 12 | followers, friends, listed, statuses (raw + log) |
| Transductive | 2 | mapped followers/friends from quoted users |
| Profile Metadata | 23 | presence flags (3), RGB colors (18), verified |
| Entity Counts | 25 | hashtags, mentions, URLs, symbols in entities |
| Frequency Encodings | 10 | source, language, location, URL domain |
| Geographic | 8 | withheld flags/counts, coordinates presence |
| Quoted Status | 22 | quoted user metrics, engagement, temporal lag |
| Generic Extractions | 229 | presence, str_len, bool, freq, n_items |
| **Total** | **343** | → 1,373 after aggregation (×4 stats + tweets count) |

## A.2 Model Selection Results

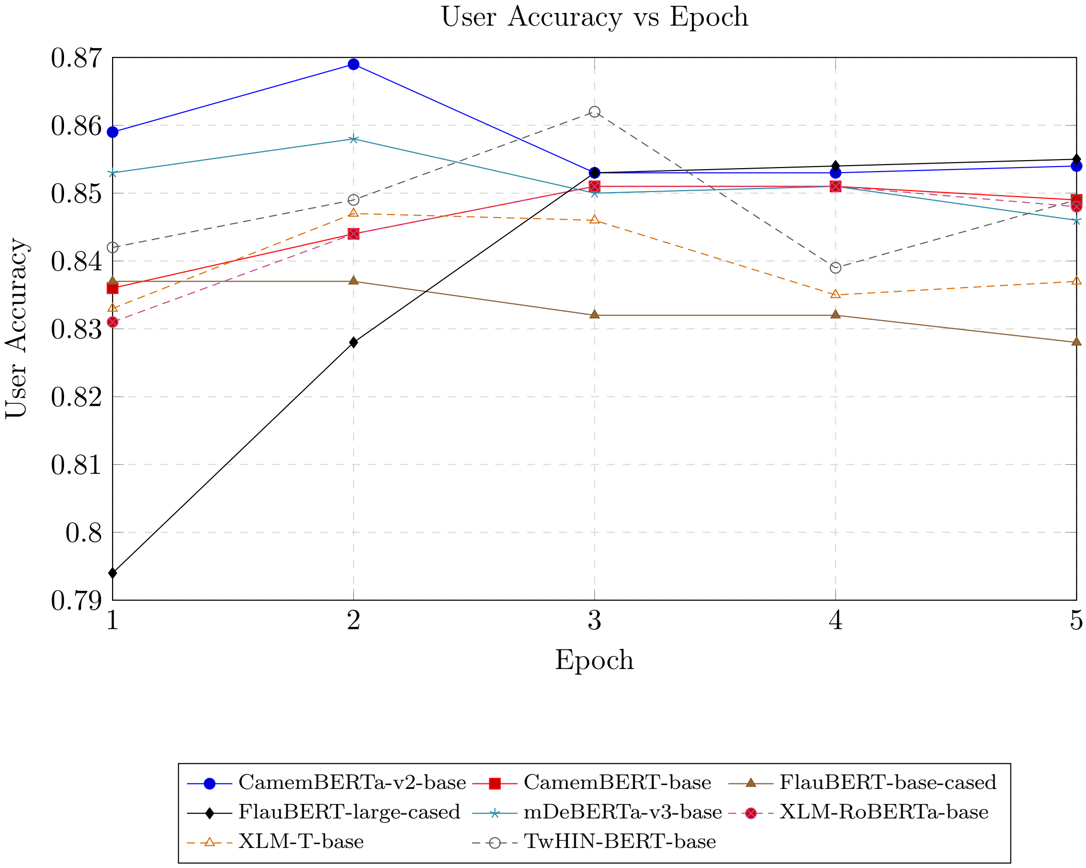

Figure 1: User-level accuracy across epochs for different encoder models.

## A.3 Hyperparameter Optimization

Table 3: Hyperparameter Optimization Results: CamemBERTa-v2

| Accuracy | Epoch | F1 | LR | Batch | W. Decay |
|---|---|---|---|---|---|
| 0.855 | 2 | 0.853 | 2.322e-05 | 48 | 0.019 |
| 0.854 | 2 | 0.853 | 1.681e-05 | 16 | 0.004 |
| 0.854 | 2 | 0.852 | 1.029e-05 | 16 | 0.004 |
| 0.853 | 2 | 0.851 | 4.328e-06 | 8 | 0.047 |
| 0.853 | 2 | 0.851 | 1.053e-05 | 8 | 0.059 |
| 0.853 | 2 | 0.851 | 1.015e-05 | 16 | 0.020 |
| 0.852 | 2 | 0.851 | 1.502e-05 | 16 | 0.183 |
| 0.852 | 2 | 0.851 | 1.180e-05 | 32 | 0.016 |
| 0.852 | 2 | 0.850 | 1.311e-05 | 16 | 0.065 |
| 0.852 | 2 | 0.850 | 6.645e-06 | 8 | 0.004 |
| 0.851 | 2 | 0.849 | 1.184e-05 | 48 | 0.006 |
| 0.851 | 2 | 0.849 | 2.335e-05 | 48 | 0.010 |
| 0.851 | 2 | 0.849 | 1.216e-05 | 16 | 0.002 |
| 0.851 | 2 | 0.849 | 2.363e-05 | 32 | 0.132 |
| 0.850 | 2 | 0.849 | 2.917e-05 | 48 | 0.014 |
| 0.848 | 2 | 0.843 | 4.038e-06 | 16 | 0.241 |
| 0.847 | 2 | 0.846 | 2.962e-05 | 16 | 0.291 |
| 0.847 | 2 | 0.847 | 1.714e-05 | 32 | 0.006 |
| 0.843 | 1 | 0.842 | 3.309e-05 | 16 | 0.007 |
| 0.842 | 2 | 0.841 | 3.835e-05 | 32 | 0.012 |
| 0.840 | 2 | 0.822 | 2.061e-06 | 8 | 0.268 |
| 0.831 | 2 | 0.828 | 2.037e-06 | 32 | 0.130 |
| 0.830 | 2 | 0.819 | 1.144e-06 | 8 | 0.094 |
| 0.826 | 2 | 0.824 | 1.726e-06 | 32 | 0.236 |
| 0.823 | 2 | 0.812 | 1.414e-06 | 32 | 0.117 |

## A.4 Behavioral Analysis: User Metrics Distribution

Table 4 presents summary statistics for key user engagement metrics by class.

Table 4: User Engagement Metrics: Summary Statistics by Class

| Metric | Class | N | Mean | Median | Std | Q10 | Q90 |
|---|---|---|---|---|---|---|---|
| `favourites_count` | Influencer | 14,306 | 21,056 | 6,254 | 39,764 | 96 | 58,291 |
| | Observer | 16,390 | 11,931 | 3,808 | 24,679 | 286 | 30,219 |
| `listed_count` | Influencer | 14,306 | 136.8 | 27 | 1,178 | 1 | 274 |
| | Observer | 16,390 | 7.1 | 1 | 30 | 0 | 14 |
| `statuses_count` | Influencer | 14,306 | 45,069 | 20,081 | 77,574 | 2,211 | 114,295 |
| | Observer | 16,390 | 11,568 | 4,120 | 30,123 | 663 | 26,698 |

*Note:* Q10 and Q90 represent the 10th and 90th percentiles respectively.

## A.5 Reply Behavior Analysis

Table 5 presents the distribution of reply percentage by class, a key discriminative feature.

Table 5: Reply Behavior Statistics by Class

| Class | N | Mean (%) | Median (%) | Std | Q10 (%) | Q90 (%) |
|---|---|---|---|---|---|---|
| Influencer | 14,306 | 20.1 | 0.0 | 27.5 | 0.0 | 60.0 |
| Observer | 16,390 | 41.0 | 40.0 | 37.9 | 0.0 | 100.0 |

*Note:* `pct_reply` represents the percentage of user tweets that are replies. A median of 0% for Influencers indicates that at least half of Influencers post no replies in their sampled tweets.

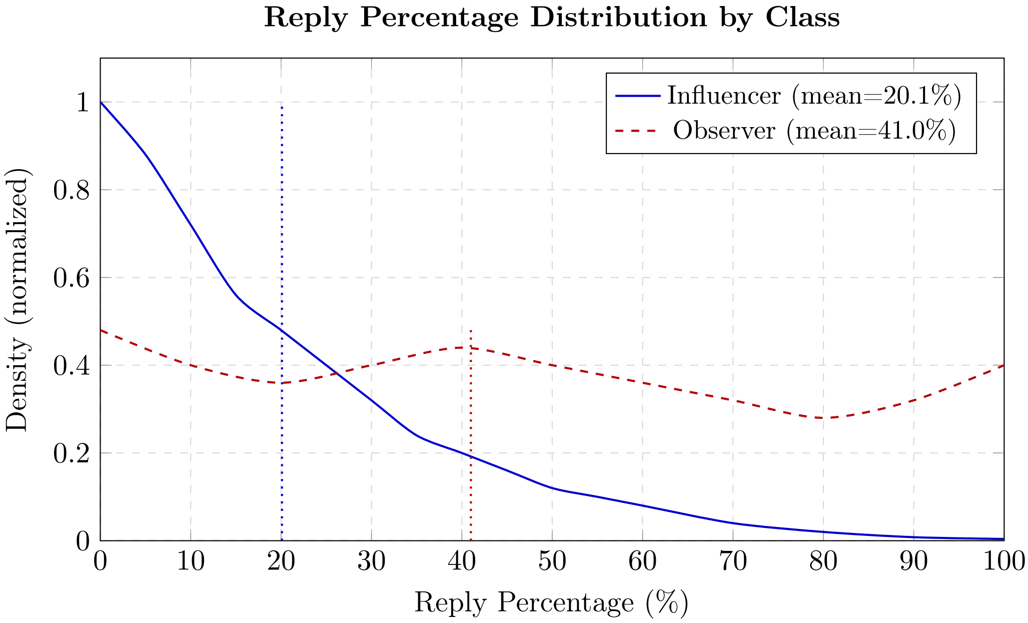

Figure 2: Schematic density distribution of reply percentage (normalized). Influencers are heavily skewed toward low reply rates (content creators), while Observers show a more uniform distribution with higher reply engagement.

## A.6 Account Age Distribution

Figure 3 presents the account age statistics by class.

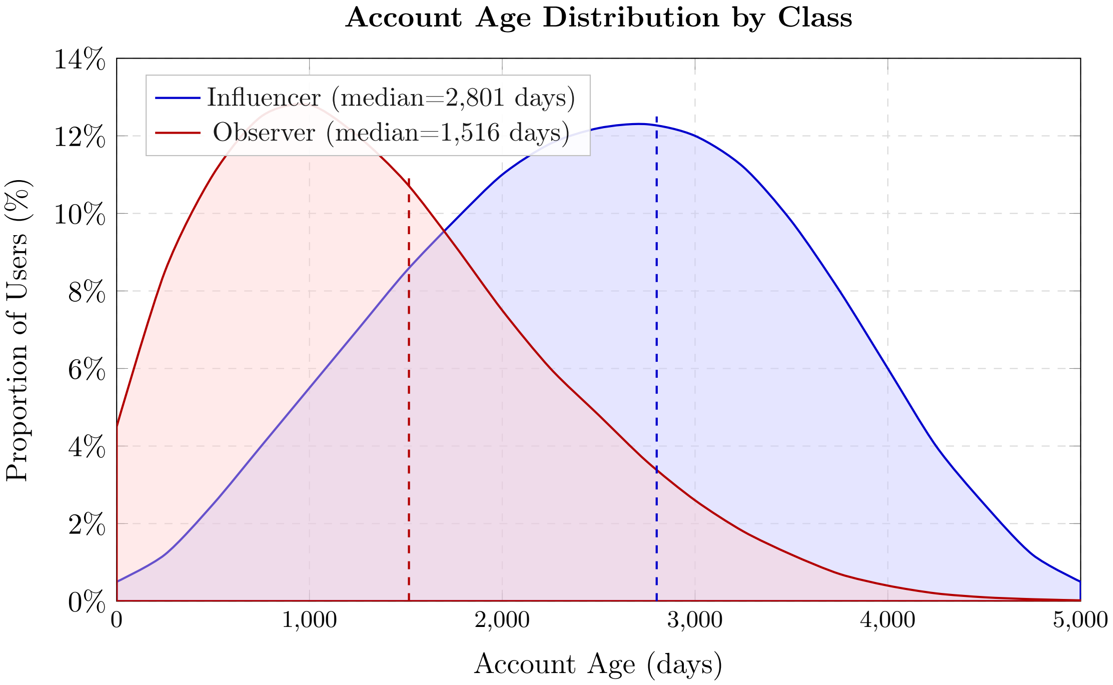

Figure 3: Influencers have a median presence of 2,801 days (≈ 7.7 years), almost doubling the 1,516 days (≈ 4.2 years) observed for Observers. This disparity signals that sustained longevity is a likely precondition for achieving influencer status.

## References

Wissam Antoun, Francis Kulumba, Rian Touchent, Éric de la Clergerie, Benoît Sagot, and Djamé Seddah. Camembert 2.0: A smarter french language model aged to perfection. *arXiv preprint arXiv:2411.08868*, 2024. URL https://arxiv.org/abs/2411.08868.

Fabrício Benevenuto, Gabriel Magno, Tiago Rodrigues, and Virgilio Almeida. Detecting spammers on twitter. In *Collaboration, electronic messaging, anti-abuse and spam conference (CEAS)*, volume 6, page 12, 2010. URL https://homepages.dcc.ufmg.br/~fabricio/download/ceas10.pdf.

Leo Breiman. Stacked regressions. *Machine Learning*, 24(1):49–64, 1996. URL https://doi.org/10.1007/BF00117832.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. URL https://arxiv.org/abs/1810.04805.

Salvatore Giorgi, Daniel Preotiuc-Pietro, Johannes C Eichstaedt, Margaret L Kern, Lyle H Ungar, and H Andrew Schwartz. The remarkable benefit of user-level aggregation for lexical-based population-level predictions. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1167–1172, 2018. URL https://aclanthology.org/D18-1148/.

Liangjie Hong and Brian D Davison. Empirical study of topic modeling in twitter. In *Proceedings of the first workshop on social media analytics*, pages 80–88, 2010. URL https://snap.stanford.edu/soma2010/papers/soma2010_12.pdf.

Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems (NIPS 2017)*, pages 3146–3154, 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/6449f44a102fde848669bdd9eb6b76fa-Paper.pdf.

Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric de la Clergerie, Djamé Seddah, and Benoît Sagot. Camembert: a tasty french language model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7203–7219, 2020. URL https://aclanthology.org/2020.acl-main.645/.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12:2825–2830, 2011. URL https://www.jmlr.org/papers/volume12/pedregosa11a/pedregosa11a.pdf.

Francisco Rangel, Paolo Rosso, Moshe Koppel, Efstathios Stamatatos, and Giacomo Inches. Overview of the pan 2013 author profiling task. In *CLEF Conference on Labs and Evaluation Fora*, pages 352–365, 2013. URL https://ceur-ws.org/Vol-1179/CLEF2013wn-PAN-RangelEt2013.pdf.

Jason Wei and Kai Zou. Eda: Easy data augmentation techniques for boosting performance on text classification tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, pages 6382–6388, 2019. URL https://aclanthology.org/D19-1670/.

David H Wolpert. Stacked generalization. *Neural Networks*, 5(2):241–259, 1992. URL https://doi.org/10.1016/S0893-6080(05)80023-1.

Pengcheng Yin, Graham Neubig, Wen-tau Yih, and Sebastian Riedel. Tabert: Pretraining for joint understanding of textual and tabular data. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8413–8426, 2020. URL https://aclanthology.org/2020.acl-main.745/.