Towards a fair vandalism detection system for Wikipedia

Mykola Trokhymovych
Pompeu Fabra University

Muniza Aslam
Wikimedia Foundation

Ai-Jou Chou
Wikimedia Foundation

Diego Saez-Trumper
Wikimedia Foundation

Ricardo Baeza-Yates
EAI, Northeastern University

Abstract

This paper presents a novel design of the system aimed at supporting the Wikipedia community in preventing vandalism on the platform. To achieve this, we collected a massive dataset of 47 languages, and applied advanced filtering and feature engineering techniques, including multilingual masked language modeling, to build the training dataset from human-generated data. The performance of the system was evaluated through comparison with the current state-of-the-art ORES system. Our research results in a significant increase in the number of languages covered, making Wikipedia patrolling more efficient to a wider range of communities. Furthermore, our model outperforms the existing state-of-the-art model, ensuring that the results provided are not only accurate but also less biased against certain groups of contributors.

Keywords: knowledge integrity, vandalism detection, fairness, machine learning, Wikipedia

Introduction

Wikipedia has a central role in the Web ecosystem. Its content is frequently used for powering other websites and products, from educational purposes, such as incorporating verified facts into curricula, to artificial intelligence solutions, such as training large language models.

Wikipedia relies on the contributions of a large community of users across the globe. This collaborative approach allows a wide range of viewpoints to appear, so that information on the site is constantly updated. However, it also requires significant manual work to maintain the site's accuracy and integrity.

The collaborative nature of Wikipedia is one of its strengths. However, it also presents some challenges, such as content quality, factual correctness, point-of-view bias, and vandalism. To address these issues, Wikipedia has a group of dedicated volunteer editors known as patrollers, who work to ensure the accuracy and integrity of the information on the site.

However, their work is not easy, as they have to keep up with the fast pace of Wikipedia. To support the work of patrollers, Wikipedia has developed machine learning (ML) systems such as ORES (Objective Revision Evaluation Service) to assist in identifying potentially damaging changes (Halfaker and Geiger, 2020). In this work, we focus on the *damaging* model, which helps patrollers easily discover and prioritize reviewing potentially harmful contributions.

While ML systems can greatly assist patrollers, there are still open problems to be solved, such as limitations regarding language coverage, precision, and bias against anonymous users. The main contributions of this work are: *(i)* Introduction of an open-source,[1] multilingual model for content patrolling on Wikipedia, outperforming the state-of-the-art models; *(ii)* Significantly increasing the number of languages covered in more than 60%; *(iii)* Study the biases of different models and discuss the trade-offs between performance and fairness.

Methods

System architecture. The proposed system takes the Wikipedia revision as input and aims to define the probability of the revert event. The key idea of our solution is to analyze the changes in text content, which is the main component of the encyclopedia page.

We use two main groups of content features: *(i)* `mwedittypes` package standard output;[2] *(ii)* fine-tuned MLMs based features. Having the content changes features, we combine them with the revision and creator metadata to pass in the final classifier to get the revert probability score. The system inference logic schema is presented in Figure 1.

Data preparation. We use *mediawiki history* and *mediawiki wikitext* data from Wikimedia Data Lake to build the dataset. The data collection process can be divided into three stages: *(i)* aggregating edits history data; *(ii)* data filtering; *(iii)* merging text data, extracting content features.

The final dataset represents the history of changes from 2022-01-01 to 2022-07-01 for 47 most edited languages in Wikipedia. It contains only revisions for page content

[1] GitHub repository: `https://github.com/trokhymovych/KI_multilingual_training`

[2] mwedittypes python package: `https://github.com/geohci/edit-types`

changes. Also, in order to reduce noise and biases in data, the following types of revisions are filtered out: *(i)* created by bots, *(ii)* new pages creation revisions, *(iii)* "edit wars". The last refers to sequential revisions that revert to one another.

We present two types of datasets of all users and anonymous only users revisions. The final datasets contain revisions for various languages, their metadata, creator information, and features that correspond to revision text differences (inserts, removes, changes). Later text differences were used to build MLM-based features for the final classifier. Also, we have the independent testing set of the following week after the training data period. The time-based splitting procedure is needed to avoid time-related anomalies that can bias evaluation results.

Models training. We tune four independent MLMs to extract signals from text content like title, inserted, removed, and changed text. We use MLM that was initially trained on multilingual datasets - `bert-base-multilingual-cased` and tune it for the binary classification task. Revisions used for MLM tuning are not later used for final classification model training. We independently tune models on two different training sets: $\text{train}^{\text{anon}}$ and $\text{train}^{\text{all}}$. Later we use aggregated final models output before (scores) and after (probabilities) the softmax layer as features for the final classifier.

As for the final classifier model, we use the Catboost classification model. It is fitted with MLMs, `mweditypes` features, and revision metadata. Revision metadata includes the difference in bytes made by revision, users' information, and the language of the page.

Results

Metrics. We evaluate our model trained on data with different configurations: (i) using different feature sets; (ii) training only on anonymous revisions, or using both registered and anonymous users' revisions as training data. We use the precision at a recall level of 0.75 ($Pr@R0.75$), and AUC score on a full unbalanced holdout testing set as the main metrics for model comparison. The results of the models' evaluation are presented in Table 1 for all users and Table 2 for anonymous-only users. Only revisions for languages implemented in ORES are used in the presented comparison tables.

The final results show that the multilingual model trained on all users' revisions with users and MLMs features outperforms all competitors, including ORES, using the AUC metric. Precision-recall graphs in Figure 2 support our reasoning. We also observe a significant improvement in performance for anonymous users.

We tested the inference time on CPU-only instance and concluded that proposed system is comparable with ORES in single score prediction speed. We also compared the models using the AUC score for different languages (Figure 3). We concluded that the presented system performs better or equal to ORES for all presented languages.

Models Fairness. Although anonymous edits tend to contain more vandalism than the ones done by registered users, the big majority of Wikimedia communities keep supporting anonymous editors (a.k.a., IP Editors). Therefore, it is important that ML systems try to mitigate potential biases against this group of editors.

We are using Disparate Impact Ratio (DIR) and the Difference in AUC score for anonymous and registered users to analyze the bias against (unprivileged) anonymous users. The results are presented in Table 3.

We calculate $DIR^{base} = 7.93$, which shows the ratio of base rates. The results shows that the current model (ORES) has a DIR equal to 20.02, much bigger than the base one. It means ORES presents a significant bias against the unprivileged class (anonymous users). At the same time, the best-performing configuration of our solution shows a DIR value of 9.54, which is much closer to the base value. It means that we still introduce the bias against the anonymous user, but it is significantly lower.

Conclusions

This paper presents a new design for a system aimed to help in preventing vandalism on Wikipedia. The model was trained using a large dataset of 47 different languages that utilize sophisticated filtering and feature engineering techniques. The results showed that the system outperformed the current state-of-the-art ORES system in terms of both the number of languages covered, accuracy and fairness. Reducing bias against anonymous users aims to help extend the active editors pool, which is the core of the Wikipedia movement.

Nevertheless, the current study has several limitations that should be considered when interpreting the results. Our study focused on analyzing changes in the text content of pages when only 56% of revisions have at least some changes in the text. Analyzing other types of content can be a great extension of our work in the future. Also, we restricted our data to the most frequent languages. We make our analysis on the fixed time frame and don't consider time-related patterns. Also, we only tested one language model and did not explore other language models that could potentially provide different results. We leave this for future work.

References

[Halfaker and Geiger2020] Aaron Halfaker and R Stuart Geiger. 2020. Ores: Lowering barriers with participatory machine learning in wikipedia. *Proceedings of the ACM on Human-Computer Interaction*, 4(CSCW2):1–37.

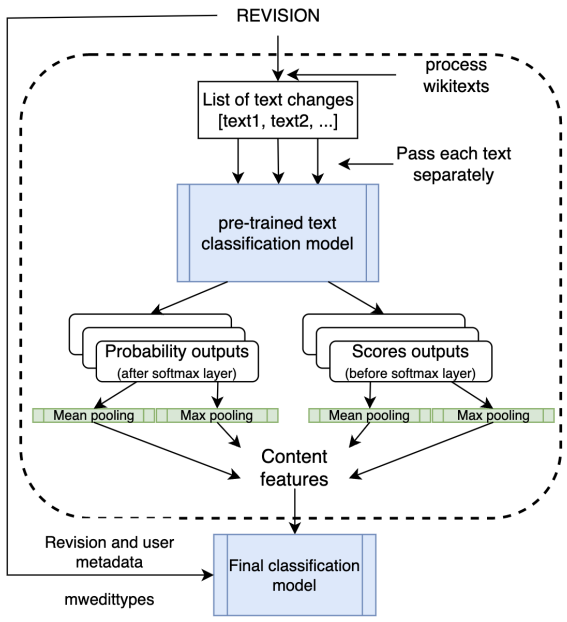

Figure 1: System inference logic schema.

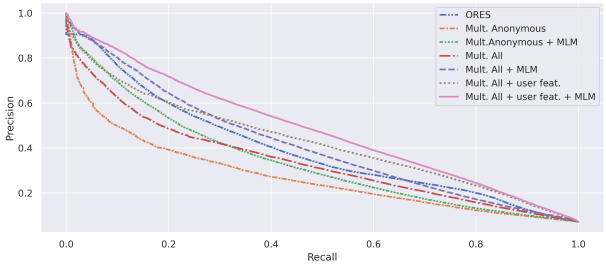

Figure 2: Precision-recall graph for all users.

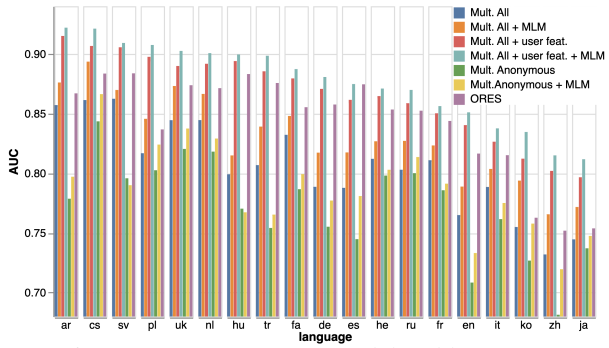

Figure 3: AUC score per model and language.

Table 1: System performance on testing set of all users.

Model	AUC	Pr@R0.75
Rule-based	0.75	0.07
ORES	0.84	0.22
Multilingual^{anon}	0.77	0.14
Multilingual^{anon} + MLM features	0.79	0.15
Multilingual^{all}	0.82	0.18
Multilingual^{all} + MLM features	0.84	0.20
Multilingual^{all} + user features	0.87	0.27
Multilingual^{all} + user features + MLM features	**0.88**	**0.28**

Table 2: System performance on testing set of anonymous users.

Model	AUC	Pr@R0.75
Rule-based	0.50	0.24
ORES	0.70	0.31
Multilingual^{anon}	0.77	0.40
Multilingual^{anon} + MLM features	**0.80**	**0.44**
Multilingual^{all}	0.75	0.38
Multilingual^{all} + MLM features	0.78	0.42
Multilingual^{all} + user features	0.76	0.39
Multilingual^{all} + user features + MLM features	0.79	0.43

Table 3: Fairness metrics evaluation.

Model	DIR	AUC diff
ORES	20.02	-0.043
Multilingual^{anon}	1.98	0.073
Multilingual^{anon} + MLM features	2.06	0.084
Multilingual^{all}	2.91	0.010
Multilingual^{all} + MLM features	3.08	0.017
Multilingual^{all} + user features	9.36	-0.035
Multilingual^{all} + user features + MLM features	9.54	-0.017