

Wikipedia is Not a Dictionary, Delete! Text Classification as a Proxy for Analysing Wiki Deletion Discussions

Hsuvas Borkakoty¹ and Luis Espinosa-Anke^{1,2}

¹Cardiff NLP, School of Computer Science and Informatics, Cardiff University, UK

²AMPLYFI, UK

{borkakotyh, espinosa-ankel}@cardiff.ac.uk

Abstract

Automated content moderation for collaborative knowledge hubs like Wikipedia or Wikidata is an important yet challenging task due to multiple factors. In this paper, we construct a database of discussions happening around *articles marked for deletion* in several Wikis and in three languages, which we then use to evaluate a range of LMs on different tasks (from predicting the outcome of the discussion to identifying the implicit policy an individual comment might be pointing to). Our results reveal, among others, that discussions leading to deletion are easier to predict, and that, surprisingly, self-produced tags (keep, delete or redirect) don't always help guiding the classifiers, presumably because of users' hesitation or deliberation within comments¹.

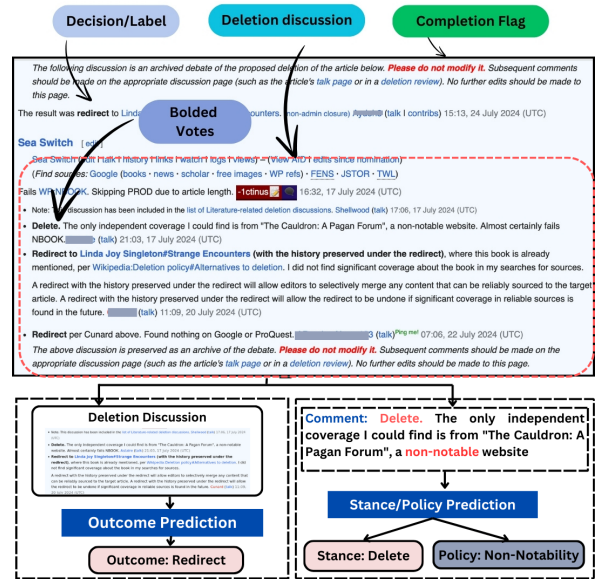


Figure 1: Example of a Deletion Discussion in English Wikipedia

1 Introduction

Wikipedia and its sister Wikis play an indispensable role as a collaborative knowledge source, and are widely used by students (Selwyn and Gorard, 2016) and the general public (Singer et al., 2017; Lemmerich et al., 2019) alike. They fulfill use cases that range from core knowledge *go-tos*, as well as “free” supporting documentation for content providers and search engines (e.g., Google² or YouTube³). However, due to their size and, most importantly, their collaborative nature, ensuring high quality in these platforms is challenging, especially given the need to “map” content to existing policies at least semi-automatically (Ribeiro et al., 2022). This is particularly relevant in the GenAI era, as AI-generated content has proliferated throughout the Internet (Brooks et al., 2024).

¹Dataset available at: <https://huggingface.co/datasets/hsuvasborkakoty/wider>.

²https://en.wikipedia.org/wiki/Relationship_between_Google_and_Wikipedia

³<https://support.google.com/youtube/answer/7630512?hl=en>

More generally, content moderation in online platforms is often the outcome of group coordination and communication (Chidambaram and Tung, 2005; Jensen and Scacchi, 2005; Butler et al., 2008). Unsurprisingly, NLP plays an important role in automating this process. For example, Singhal et al. (2023) defined a framework for social media moderation as a function of community guidelines, policy enforcement and violation detection. Some prominent examples of works connected with such a framework are: policy based content moderation in Facebook (Sablosky, 2021), rule-breaking behavior analysis on Reddit (Chandrasekharan et al., 2018), and topic based content moderation discourse on X (Alizadeh et al., 2022). In the case of Wikis, both the guidelines and the rules that govern the quality of their content are maintained by contributions from the community (Seering, 2020), where discussion-based approaches towards content moderation are the norm.

Task	Lang.	Platform	Label Set
Outcome Prediction	En	Wikipedia	delete, keep, redirect, no-consensus, merge, speedy keep, speedy delete, withdrawn
		Wikidata-ent.	delete, keep, merge, redirect, no-consensus, comment
		Wikidata-pr.	delete, keep, no-consensus
		Wikiquote	delete, keep, redirect, merge, no-consensus
		Wikinews	delete, keep, speedy delete, comment
	Es	Wikipedia	borrar (delete), mantener (keep), fusionar (merge), otros (others)
Stance Detection	Gr	Wikipedia	Διαγραφή (delete), Διατήρηση (keep), Δεν υπάρχει συναίνεση (no-consensus)
	En	Wikipedia	delete, keep, merge, comment
Policy Prediction	En	Wikipedia	Wikipedia:Notability, Wikipedia:What Wikipedia is not, Wikipedia:No original research, Wikipedia:Verifiability, Wikipedia:Arguments to avoid in deletion discussions, Wikipedia:Biographies of living persons, Wikipedia:Criteria for speedy deletion, Wikipedia:Articles for deletion, Wikipedia:Wikipedia is not a dictionary, Wikipedia:Deletion policy

Table 1: Label set for the different tasks and datasets we consider in this paper (Outcome, Stance, and Policy), for three languages (English: En, Spanish: Es, and Greek: Gr) and five (4+1) platforms (Wikipedia, Wikidata-Entity (ent.) and Property (pr.), Wikinews, and Wikiquote).

The way this generally works is that, given an article flagged by a community member, users justify their stance towards it and its adherence to the policies, and then editors act. However, manual efforts to clear the backlog are insufficient. Therefore, NLP techniques for predicting the outcome of a deletion discussion, or for capturing a user’s stance towards a specific article are critical (Mayfield and Black, 2019b; Kaffee et al., 2023). Despite this need, there is a surprising lack of work beyond Wikipedia, and there is almost no published work that looks at non-English languages (with the exception of Kaffee et al. (2023)). Moreover, in-depth comparative analyses of parameter-efficient techniques have also been so far largely unexplored.

We therefore aim to address all of the above with an analysis on a novel collection of deletion discussions, in three languages and four platforms. These discussions come, if resolved, alongside the discussion outcomes (generally speaking, suggesting to keep or delete the article, although the actual outcome tags are more fine grained than this), with individual comments having their own stance and referring to specific policies Kaffee et al. (2023). Our classification experiments set strong baseline results for the community to build upon, and provide insights into these community-led activities.

2 Tasks and dataset Construction

We consider three tasks, namely (1) **Outcome prediction** - given a full discussion around an article

Language	Platform	Total
en	Wikipedia	18,528
	Wikidata-entities	355,428
	Wikidata-properties	498
	Wikinews	91
	Wikiquote	695
es	Wikipedia	3,274
gr	Wikipedia	392

Table 2: Overall number of deletion discussions per language and platform in the outcome prediction dataset.

marked for deletion, predict the final decision; (2) **Stance detection**, i.e. given an individual comment, determine its stance towards the decision to be made for that article; and (3) **Policy prediction**, where again, given one single comment, we want to determine the policy that comment is most likely be referring to (Figure 1 shows an example). We build a novel dataset for outcome prediction, while for the other two tasks we largely rely on the dataset from Kaffee et al. (2023) (although with some important modifications to enable the goal of this paper, namely an in-depth analysis). We provide more detail about these datasets in the following subsections.

2.1 Outcome Prediction

We retrieve and clean deletion discussions programmatically⁴ for three different languages and four

⁴We use the WIDE-ANALYSIS toolkit: <https://pypi.org/project/wide-analysis/> (Borkakoty and

Platform	Language	Example title	Discussion (truncated)	Outcome
Wikipedia	en	Beast Poetry	Editor 1: Keep in one form or another. Editor 2: One option could be to re-frame the article to be about the book.	Keep
Wikidata-Ent	en	Q28090948	Editor 1: no description : Vandalism.	Delete
Wikidata-Prop	en	JMdict sequence number (P11700)	Editor 1: Deleted - () Support I assume no comments have been made because this is a clear case to delete...	Delete
Wikinews	en	Mugalkhod Jeedga Mutta organizes mass marriage in Belgaum, India	Editor 1: There is no further meaningful work on the article. Editor 2: All advice ignored, kill it with cleansing fire and stop wasting time.	Speedy delete
Wikiquote	en	3rd Rock From The Sun	Editor 1: Two reasons to delete this: - It is a copy of about half of quotes on IMDB - 3rd Rock from the Sun is a different article. Editor 2: Merge, perhaps with some trimming.	Merge
Wikipedia-Es	es	Héroes: El legado de la Evolución	Editor 1: Bórrese Irrelevante enciclopédico. Editor 2: Bórrese Irrelevante.	Borrar
Wikipedia-gr	gr	Μουσείο Μιχάλη Τσαρτσίδη	Editor 1: Σχόλιο Διαφωνώ έντονα με την λογική/φράση «Απλώς ένα από τα πολλά ανά την Ελλάδα μουσε... Editor 2: Ο μόνος λόγος διαγραφής μπορεί να είναι η παραβίαση πνευματικών.	διαγραφή

Table 3: Examples for different platforms and languages, alongside outcome labels.

platforms (with Wikidata being split in two: properties and entities, as their discussions happen separately). In terms of coverage, for small platforms with less activity such as Wikinews or Wikiquote⁵, we consider all data available in their website at the time of scraping, whereas for larger and more active platforms like Wikidata, we consider the last 4 full years (from 2021 to 2024, both inclusive). Label sets per task are provided in Table 1, whereas statistics in terms of raw size can be found in Table 2.

2.2 Dataset for Stance Detection and Policy Prediction

For stance detection and policy prediction we use the existing WIKI-STANCE dataset (Kaffee et al., 2023). We keep the stance detection dataset as-is, including their original label set. However, for policy prediction, we consider a reduced label set in order to perform error analysis, and so keep only the top 10 most frequent labels (as opposed to the 92 contained in the original dataset). This change, however, has a small effect on the overall dataset size, as retaining only these labels still results in roughly 80% of the original dataset. As an example, policy labels contained in the original dataset

Espinosa-Anke, 2024).

⁵According to Wikimedia Statistics, in 2024 Wikipedia and Wikidata received 130 Billion and 3 Billion pageviews, whereas Wikinews and Wikiquote received 77 Million and 179 Million respectively.

like Wikipedia: Userfication, Wikipedia: Record charts or Wikipedia: Attack page account for only 106, 105 and 102 instances, respectively (out of 437,770, which means a negligible percentage, around 0.02%).

3 Experiments

With these datasets in place, we proceed to run classification experiments.

3.1 Outcome Prediction

While previous works (Mayfield and Black, 2019a) cast outcome prediction as binary classification (Delete and Keep), we follow Wikipedia’s official guidelines⁶ and propose a more nuanced scheme (again, c.f. Table 1) and re-cast it as a multi-class classification. Following Mayfield and Black (2019b), we have two set ups: *Masked*, where labels are redacted from the comments, and *Full-text* where classifiers see the full dataset, including the self-assigned labels (which in theory act as extremely informative features about the stance of each comment and therefore good predictors of the final outcome of the discussion - however, as we will see, this is not always the case). We evaluate BERT (Devlin et al., 2018), RoBERTa (Liu et al., 2019), DistilBERT (Sanh et al., 2020) (all in Base and Large), and Twitter-RoBERTa-Base (Barbieri

⁶https://en.wikipedia.org/wiki/Wikipedia:Guide_to_deletion

Input Type	Model	Wikip.	Wikid.-ent	Wikid.-pr	Wikin.	Wikiq.
Fulltext	RoBERTa-B	0.56	0.61	0.56	0.4	0.71
	RoBERTa-L	0.58	0.63	0.62	0.47	0.76
	BERT-B	0.56	0.57	0.54	0.4	0.7
	BERT-L	0.58	0.62	0.61	0.47	0.73
	DistilBERT	0.55	0.56	0.5	0.39	0.57
	Tw.-RoBERTa-B	0.49	0.6	0.55	0.4	0.7
Masked	RoBERTa-B	0.49	0.51	0.56	0.36	0.62
	RoBERTa-L	0.52	0.61	0.56	0.42	0.65
	BERT-B	0.49	0.57	0.56	0.33	0.7
	BERT-L	0.5	0.62	0.6	0.36	0.72
	DistilBERT	0.43	0.56	0.5	0.27	0.32
	Tw-RoBERTa-B	0.46	0.52	0.42	0.3	0.38

Table 4: F1 Scores for fine-tuned models in Wikipedia (Wikip.), Wikid.-end (Wikidata, entities subset), Wikid.-pr (Wikidata, entities subset), and Wikiq. (Wikiquote), both for (a) full text and (b) masked text inputs. Models are identified by their versions: Tw (Twitter), B (Base) and L (Large).

et al., 2020) (in order to explore the effect of models tailored to user-generated content). Information about training, validation and tests splits, and implementation details, are provided in Appendices A (Table 10) and B, respectively. Furthermore, we divide our set of experiments in three scenarios: in-platform, cross-platform and multilingual.

3.1.1 In-platform

We train each model with platform-specific training sets under both masked and full text settings, and report F1 results in Table 4. As expected, we can see that hiding the self-reported tags generally causes a drop in performance across the board, most noticeable in the X (Twitter)-specific model. We can also see that RoBERTa large is always the best model in full text, while this is more inconsistent in the masked setup. Further per-label analysis of the difference between the full text and masked settings for RoBERTa large (the best performing model) is provided in Figure 2, which shows confusion matrices for the Wikipedia-en dataset. The performance drop for ‘keep’, ‘merge’ and ‘withdrawn’ suggests that editors are more decisive about deletion of the article than keeping it. It also shows that full text is almost always useful, but interestingly, merge decisions benefit less from seeing these tags, likely because merging discussions are often less explicit and drift more between deletion and keep. Another interesting finding (which is consistent across both platforms) is that the withdrawn outcome often gets confused with keep, again reinforcing this idea of more ambiguity when the decision is *not* lead-

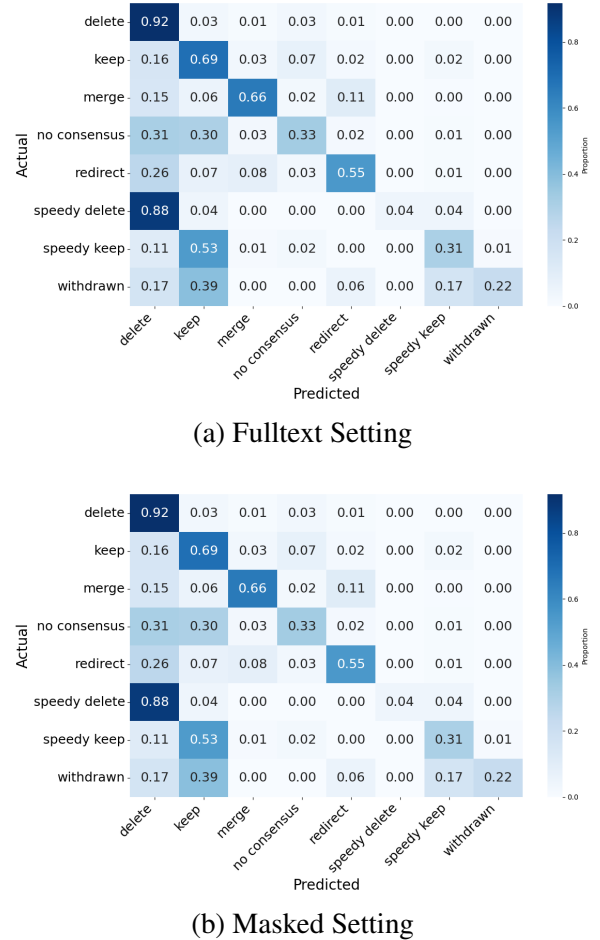


Figure 2: Confusion Matrix for RoBERTa-Large model in Outcome Prediction Task.

ing towards deletion. Next, no consensus outputs seem very hard to predict, with an almost even split

between predictions spread among the correct class (no consensus), delete and keep. And finally, a striking result is the massive confusion between speedy delete and delete in the masked setting. This suggests that, in practice, there virtually no difference in how editors *talk about* deleting articles, but they are however implicitly opinionated about how urgently the decisions needs to happen.

Another interesting perspective on this experiment is the option to explore more efficient approaches than simply fine-tuning an arguably large model, especially given that the size of the datasets is very varied. To test our hypothesis that simpler approaches could be beneficial, we evaluate a SetFit (Sentence-transformer Fine-tuning) model (Tunstall et al., 2022). SetFit is a simple yet powerful technique that fine-tunes a sentence transformers model⁷ by artificially sampling training pairs for a contrastive learning stage, and uses the fine-tuned embeddings as feature vectors for a logistic regression classifier. Note that, in SetFit, in the embedding fine-tuning stage a large number of document pairs can be generated, specifically $K(K - 1)/2$, where K is the number of labeled examples (i.e., the original training set). Therefore, we subsample the original training sets into a smaller stratified training set of 100 labeled examples. We found a striking boost in performance with this model, especially for smaller datasets (like Wikinews), which suggestst that, in production environments, SetFit could be an efficient and highly performing option. We show a SetFit vs best model comparison in Table 5.

3.1.2 Cross-platform

In previous training experiments, we observe that models struggle to perform well in smaller datasets, likely due to the lack of training data (like Wikinews, where the performance was lowest on average for all models, with some cases up to 30% drop - as in the case of the Twitter-specialized model). This motivates us to explore the potential of a cross-platform training regime, under the hypothesis that many features in deletion discussions might be similar across Wiki-platforms. We therefore perform an experiment where, for each model and training set, we evaluate on the test set of all the Wiki-platforms. Due to the variation in label sets, we simplify this experiment and map all the

⁷We used BAAI/bge-base-en-v1.5 (Xiao et al., 2023), a model roughly 3 times smaller than our best performing fine-tuned models.

Platform	Setting	Model	F1
Wikipedia	Fulltext	RoBERTa-L	0.60
	Masked	RoBERTa-L	0.52
	Fulltext	SetFit	0.65
	Masked	SetFit	0.57
Wikidata-ent	Fulltext	RoBERTa-L	0.63
	Masked	RoBERTa-L	0.61
	Fulltext	SetFit	0.88
	Masked	SetFit	0.87
Wikidata-pr	Fulltext	RoBERTa-L	0.62
	Masked	BERT-L	0.60
	Fulltext	SetFit	0.61
	Masked	SetFit	0.70
Wikinews	Fulltext	RoBERTa-L	0.47
	Masked	RoBERTa-L	0.42
	Fulltext	SetFit	0.57
	Masked	SetFit	0.44
Wikiquote	Fulltext	RoBERTa-L	0.76
	Masked	BERT-L	0.72
	Fulltext	SetFit	0.87
	Masked	SetFit	0.44

Table 5: Best model vs. SetFit results.

labels of each dataset to only keep and delete, the two common labels in all the datasets. We test the models in the fulltext setting.

The expectation from the results listed in Table 6 would be to have a bold diagonal, i.e., a model trained on dataset X would be expected to be the best on the test set for X. While this is primarily the case, we find a comparable performance in other datasets, indicating a subtle but prominent generalization of the models across the platforms. The outlier in this trend is Wikinews, where we see both Wikipedia and Wikidata Entity-derived models performing better than the in-domain model. This can be attributed to the size of Wikinews dataset, which may not be enough for the models learn platform specific patterns. In fact, for this case, training on the most general dataset (i.e., Wikipedia) yields the best performance, specifically a non-negligible 11% increase in F-1. The performances of Wikidata-entity and property across all other platform is also quite similar, despite of the large difference in data instances between them, further signaling the similarity in contents between the two important components of the same platform. However, there is an important difference, it seems that Wikidata properties transfers well into Wikidata entities (with only a 2% drop in F1), however this

	Wikip.	Wikid.-ent	Wikid.-pr	Wikin.	Wikiq.
Wikip.	0.76	0.63	0.67	0.55	0.14
Wikid.-ent	0.43	0.89	0.87	0.33	0.63
Wikid.-pr	0.61	0.71	0.83	0.04	0.1
Wikin.	0.44	0.42	0.39	0.35	0.04
Wikiq.	0.07	0.04	0.05	0.01	0.94

Table 6: Results of F1 scores of model performance on different test sets (columns represent the data models were trained on and rows represent the data model is tested on.).

Language	Model	F1 (FT)	F1 (M)
gr	XLM-R-Base	0.47	0.38
	XLM-R-Large	0.59	0.49
	MBERT	0.59	0.40
	Tw.-XLM-R	0.44	0.40
	SetFit	0.81	0.60
es	XLM-R-Base	0.66	0.47
	XLM-R-Large	0.88	0.85
	MBERT	0.70	0.67
	Tw.-XLM-R	0.56	0.46
	SetFit	0.90	0.61

Table 7: F1 score for Fulltext (FT) and Masked (M) settings for multilingual models in Spanish (es) and Greek (gr).

is not the case vice versa, as a Wikidata entities-trained model falls short by 12% vs the in-domain model (trained on Wikidata properties).

Amongst all the similar performances from the models, an obvious outlier is Wikiquote, which fails to perform well on the other datasets, and all of the other models also fail to perform on the Wikiquote dataset, clearly showing the distinctive nature of Wikiquote discussions. However, it should be noted that compared to other popular platforms like Wikipedia, Wikiquote has a smaller editor base⁸, which could cause a lack of diversity, consistency and overall quality, such as unmoderated discussions or inconsistencies between outcomes.

3.1.3 Multilingual

For multilingual datasets (Wikipedia-es and Wikipedia-gr), we experiment with XLM-R (Conneau, 2019) (Base and Large, XLM-R-Base and XLM-R-Large), Multilingual BERT (MBERT) (Devlin et al., 2018), and Twitter-XLM-R (Tw.-XLM-R) (Barbieri et al., 2020) (to explore if a model specialized on Twitter, another instance of user-generated content, could give advantages). We also

⁸According to Wikiquote’s official Wikipedia page, it only has 474 active editors as compared to Wikipedia’s 126,324 (in other words, Wikiquote has only 0.004% of the editors of Wikipedia).

introduced SetFit in this experiment, on top of a multilingual embedding model⁹. Due to the large data difference between the languages we consider (c.f. Table 2), as well as them having a different set of outcome labels, a cross language comparison is perhaps not appropriate, and therefore we discuss classification results on both languages separately.

Table 7 shows results following a similar performance pattern as the English experiment, with a significant difference between fulltext and masked setups. However, it is worth noting that XLM-R-Large and MBERT trained and tested on masked data in Spanish were able to come very close to the fulltext variant. This clearly points to Spanish editors using more explicit language when discussing whether an article should be deleted, giving a stronger signal to a classifier even after masking these self-assigned labels. This can be further verified in confusion matrices (on the fulltext setting, Figure 4), where the Greek model struggles to distinguish between delete and no consensus, which is certainly not the case in Spanish. Concerning the SetFit results, these are, again, surprisingly good, being the best model on the same test sets over fully fine-tuned models, with the exception only of the masked experiment, where it is outperformed by XLM-R-Large and MBERT. We attribute this to a potential mismatch of the style/topics/theme of the subsampled dataset and the training set. We leave for future work performing multiple runs to evaluate the robustness of SetFit (or other approaches based on synthetic data generation) when datasets are varied.

Finally, we asked ourselves the question of why a strong multilingual model (XLM-R) further specialized on multilingual data from social media (Tw.-XLM-R) model would perform so poorly on another instance of user-generated texts which, as we can see from the examples in Tables 3 and 8, are not so different from well-formed tweets (note that in the original sampling from Barbieri et al. (2020) a few heuristics were put in place to filter out pure noise like all-emoji tweets). One approach to gain further insights is by computing pseudo (log) likelihood (Salazar et al., 2019) over sequences in the dataset from different models by taking a sample of the data, masking the actual (sub)word one at a time, and compute the loss of the models, and averaging over the whole sequence. Higher likelihood typically points to a model with a “good grasp” of

⁹Specifically, paraphrase-multilingual-MiniLM-L12-v2.

Policy	Example	Instances
Wikipedia:Notability	[WP:N] Fails and with only routine coverage.	232,422 (70.22%)
Wikipedia:What Wikipedia is not	If there are sources linking this to the attack.	34,559 (10.44%)
Wikipedia:No original research	"WP:OR, . Mishmash of random trivia.	13,583 (4.10%)
Wikipedia:Verifiability	I'm leaning delete per as this is completely unverified.	12,531 (3.78%)
Wikipedia:Arguments to avoid in deletion discussions	Her own work is admittedly ordinary, even run-of-the-mill.	8,105 (2.44%)
Wikipedia:Biographies of living persons	Because of BLP requirements, it needs to be rewritten.	7,346 (2.21%)
Wikipedia:Criteria for speedy deletion	In regards to the above comment about speedy deletion.	5,833 (1.77%)
Wikipedia:Articles for deletion	What searches did you do to establish notability?	5,758 (1.75%)
Wikipedia:Wikipedia is not a dictionary	Clearly a lexical entry and in violation of policies.	5,474 (1.65%)
Wikipedia:Deletion policy	In this case, I don't see the point of a redirect.	5,332 (1.61%)

Table 8: Top 10 Policies of policy prediction task dataset with examples and number of instances (with percentage of the complete dataset).

the presented data (domain, style, themes, etc), as shown, e.g., in the context of temporal adaptation (Loureiro et al., 2022). To this end, we computed pseudo log likelihood over a sample of 1,000 Greek Wikipedia articles. We find the distribution in Figure 3, which, instead of a curve-shaped distribution (which would be the ideal), has two clear spikes, which suggests that this particular Twitter model might struggle to generalize (very low likelihood scores). Further analysis into the role of tokenization is left for future work.

3.2 Stance and Policy prediction

Stance detection classifies a moderator’s opinion towards the article using stance labels (keep, delete, merge, comment), while policy prediction identifies an explicitly or implicitly mentioned Wikimedia policy (refer to Table 8 for illustrative examples). Both are comment level tasks. We experiment with the same base models from the Outcome Prediction task. We report our results in Table 9, which shows results for both tasks according

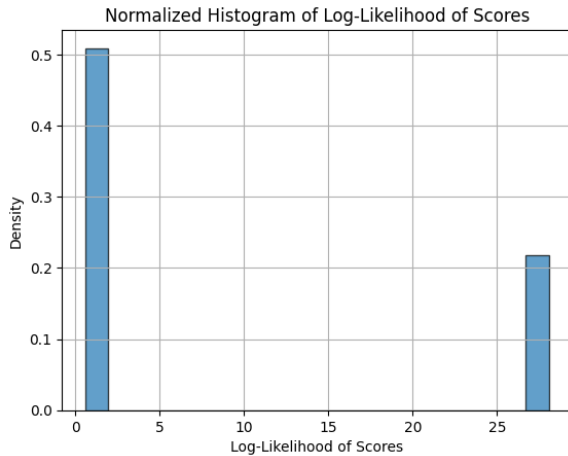
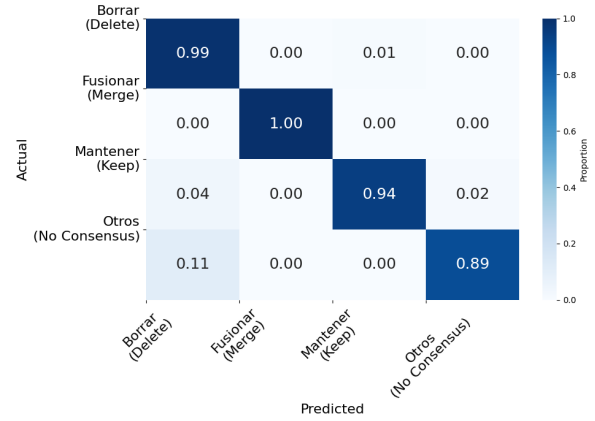
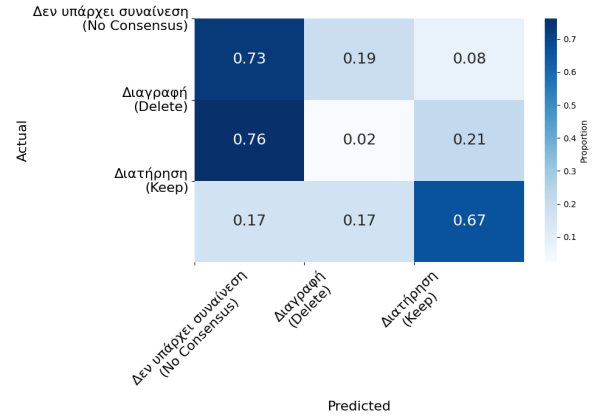


Figure 3: Normalized perplexity distribution for Twitter-XLM-Roberta in a sample of the Greek Wikipedia.



(a) Wikipedia-Es



(b) Wikipedia-Gr

Figure 4: Confusion Matrix for XLM-RoBERTa-Large model in Outcome Prediction Task for Spanish and Greek Wikipedia.

to weighted F1-score. We are interested primarily in this metric to understand the benefit of such models for the platform as a whole, rather than investigating nuances in individual categories which account, in practice, for a very small proportion of the dataset. We still perform per-label analysis, but we believe weighted F1 in this scenario sends

	Stance				Policy			
	Acc.	Prec.	Rec.	F1	Acc.	Prec.	Rec.	F1
RoBERTa-B	0.90	0.83	0.78	0.81	0.83	0.70	0.56	0.61
RoBERTa-L	0.94	0.85	0.81	0.83	0.86	0.74	0.62	0.67
BERT-B	0.89	0.81	0.80	0.80	0.81	0.65	0.49	0.55
BERT-L	0.91	0.84	0.82	0.83	0.84	0.71	0.59	0.63
DistilBERT-B	0.90	0.83	0.74	0.78	0.80	0.69	0.46	0.50
Tw.-RoBERTa-B	0.88	0.80	0.66	0.70	0.81	0.68	0.53	0.58
SetFit	0.83	0.81	0.82	0.82	0.66	0.68	0.67	0.67

Table 9: Stance and policy prediction results (with Weighted-F1 scores).

a clearer takeaway message to those interested in automating content moderation in these platforms in production environments.

3.2.1 Stance Detection

Similar to Kaffee et al. (2023) we pose our stance detection task as a 4-class classification with the labels delete, keep, merge and comment, where the first three labels carry the same meaning as the outcome prediction task, and comment means the discussion goes on. Our stance detection results are comparable to the ones reported in Kaffee et al. (2023), which in fact points to their strong and robust model, since their reported results were in macro F1, and are only slightly lower than ours (80% macro F1, vs our 83% weighted F1). In terms of analysis, we do not find any major differences between RoBERTa-Large and BERT-Large, although as is the norm in this paper, Tw.-RoBERTa-B does not perform well. Following from our previous experiments, we also test the ability of SetFit in these tasks. In this case, we also downsample the training and validation sets, in this case, from the original to 1,000 (train) and 300 (validation) stratified samples, with the test set staying the same for enabling a comparison. While not clearly outperforming the other models, as it was the case in previous experiments, it turned out to be an extremely competitive option, rivaling fully fine-tuned PLMs.

3.2.2 Policy Prediction

We modify the policy prediction task into a 10-label setup and follow a similar experimental setup as in previous sections. Our results show that with this task formulation works quite well, with 0.67 F1 for the best model (RoBERTa-Large) as shown in Table 9. In terms of comparison with previous works, Kaffee et al. (2023) reported Accuracy figures about 0.75 on the original dataset (90+ labels), whereas we achieve about 10 points more in a trimmed down version. This suggests that the very long tail of about 80 infrequent labels and the likely under-performance on them does not im-

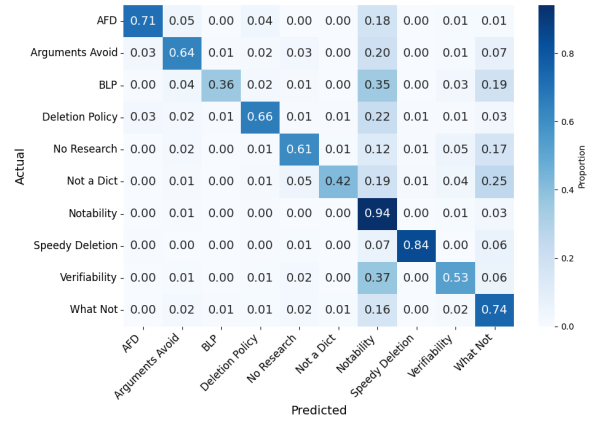


Figure 5: Confusion Matrix for RoBERTa-Large in Policy Prediction.

pact the overall picture, and we can conclude that both their models and ours would behave similarly if deployed. The most interesting part of this experiment, however, is again looking at sources of confusion in the test set. In Figure 5 we see, for RoBERTa-Large, a well formed diagonal showing the correlation between actual and predicted labels. However, two major confounding sources emerge: ‘Notability’ and ‘What Not’ (shortened for ‘What Wikipedia is Not’). Notability is less surprising as this is the most frequent category, but ‘What Not’ seems to be an overly generic category acting as a superset of other finer grained policies like ‘Not a Dict’ (Wikipedia is Not a Dictionary). It would be interesting to explore the actual differences in comments pointing to these arguably interrelated policies, which could perhaps lead to merging them or further splitting ‘What Not’ into others.

4 Conclusion

Automated Content moderation is a challenging yet important part of maintaining healthy content in community driven Wiki-platforms. Through this work, we analyze four different Wiki-platforms and three languages to give an all-round understanding of automated content moderation scenarios in these Wikis. Our analysis shows that these community based platforms can highly benefit from the usage of PLM based content moderation strategies, and to that end, we contribute a dataset and a range of strong baseline results from different PLMs for the community to build on.

Limitations

Our work does not extensively explore all deletion discussions obtainable from Wikipedia (throughout the years), even though it can be obtained using our package. We also do not explore any other LMs except BERT-family of models, and due to lack of domain data for sentiment analysis and offensive language detection, we do not train our own models for those tasks. Finally, the tool we propose here can be made better with integration of more analytical tasks and capabilities of model based activities, such as fine-tuning.

Ethics statement

We believe that enhancing quality control for Wikipedia and other sibling Wiki platforms, which is the most popular online encyclopedia through content moderation is always of utmost importance. There is importance of Wikipedia as a viable knowledge source for users, and a data source for today's NLP research is undeniable. This calls for the necessity of tools that enable automated content moderation, so that the discussions that happens behind the curtain of Wikipedia articles regarding its reliability should maintain its standard, while providing resolution for the disputed ones.

References

- Meysam Alizadeh, Fabrizio Gilardi, Emma Hoes, K Jonathan Klüser, Mael Kubli, and Nahema Marchal. 2022. Content moderation as a political issue: The twitter discourse around trump's ban. *Journal of Quantitative Description: Digital Media*, 2.
- Francesco Barbieri, Jose Camacho-Collados, Luis Espinosa Anke, and Leonardo Neves. 2020. Tweeteval: Unified benchmark and comparative evaluation for tweet classification. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1644–1650.
- Hsuvas Borkakoty and Luis Espinosa-Anke. 2024. Wide-analysis: Enabling one-click content moderation analysis on wikipedia's articles for deletion. *arXiv preprint arXiv:2408.05655*.
- Creston Brooks, Samuel Eggert, and Denis Peskoff. 2024. The rise of ai-generated content in wikipedia. *arXiv preprint arXiv:2410.08044*.
- Brian Butler, Elisabeth Joyce, and Jacqueline Pike. 2008. Don't look now, but we've created a bureaucracy: the nature and roles of policies and rules in wikipedia. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 1101–1110.
- Eshwar Chandrasekharan, Mattia Samory, Shagun Jhaver, Hunter Charvat, Amy Bruckman, Cliff Lampe, Jacob Eisenstein, and Eric Gilbert. 2018. *The internet's hidden rules: An empirical study of reddit norm violations at micro, meso, and macro scales*. *Proc. ACM Hum.-Comput. Interact.*, 2(CSCW).
- Laku Chidambaram and Lai Lai Tung. 2005. Is out of sight, out of mind? an empirical study of social loafing in technology-supported groups. *Information systems research*, 16(2):149–168.
- A Conneau. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
- Chris Jensen and Walt Scacchi. 2005. Collaboration, leadership, control, and conflict negotiation and the netbeans. org open source software development community. In *Proceedings of the 38th Annual Hawaii International Conference on System Sciences*, pages 196b–196b. IEEE.
- Lucie-Aimée Kaffee, Arnav Arora, and Isabelle Augenstein. 2023. Why should this article be deleted? transparent stance detection in multilingual wikipedia editor discussions. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5891–5909.
- Florian Lemmerich, Diego Sáez-Trumper, Robert West, and Leila Zia. 2019. Why the world reads wikipedia: Beyond english speakers. In *Proceedings of the twelfth ACM international conference on web search and data mining*, pages 618–626.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. *Roberta: A robustly optimized bert pretraining approach*. *Preprint*, arXiv:1907.11692.
- Daniel Loureiro, Francesco Barbieri, Leonardo Neves, Luis Espinosa Anke, and Jose Camacho-Collados. 2022. Timelms: Diachronic language models from twitter. *arXiv preprint arXiv:2202.03829*.
- Elijah Mayfield and Alan W Black. 2019a. Analyzing wikipedia deletion debates with a group decision-making forecast model. *Proceedings of the ACM on Human-Computer Interaction*, 3(CSCW):1–26.
- Elijah Mayfield and Alan W Black. 2019b. Stance classification, outcome prediction, and impact assessment: Nlp tasks for studying group decision-making. In *Proceedings of the Third Workshop on Natural Language Processing and Computational Social Science*, pages 65–77.

Manoel Horta Ribeiro, Justin Cheng, and Robert West. 2022. Automated content moderation increases adherence to community guidelines. *arXiv preprint arXiv:2210.10454*.

Jeffrey Sablosky. 2021. Dangerous organizations: Facebook’s content moderation decisions and ethnic visibility in myanmar. *Media, culture & society*, 43(6):1017–1042.

Julian Salazar, Davis Liang, Toan Q Nguyen, and Katrin Kirchhoff. 2019. Masked language model scoring. *arXiv preprint arXiv:1910.14659*.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2020. [Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter](#). *Preprint*, arXiv:1910.01108.

Joseph Seering. 2020. Reconsidering self-moderation: the role of research in supporting community-based models for online content moderation. *Proceedings of the ACM on Human-Computer Interaction*, 4(CSCW2):1–28.

Neil Selwyn and Stephen Gorard. 2016. Students’ use of wikipedia as an academic resource—patterns of use and perceptions of usefulness. *The Internet and Higher Education*, 28:28–34.

Philipp Singer, Florian Lemmerich, Robert West, Leila Zia, Ellery Wulczyn, Markus Strohmaier, and Jure Leskovec. 2017. Why we read wikipedia. In *Proceedings of the 26th international conference on world wide web*, pages 1591–1600.

Mohit Singhal, Chen Ling, Pujan Paudel, Poojitha Thota, Nihal Kumarwamy, Gianluca Stringhini, and Shirin Nilizadeh. 2023. Sok: Content moderation in social media, from guidelines to enforcement, and research to practice. In *2023 IEEE 8th European Symposium on Security and Privacy (EuroS&P)*, pages 868–895. IEEE.

Lewis Tunstall, Nils Reimers, Unso Eun Seo Jo, Luke Bates, Daniel Korat, Moshe Wasserblat, and Oren Pereg. 2022. Efficient few-shot learning without prompts. *arXiv preprint arXiv:2209.11055*.

Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. 2023. [C-pack: Packaged resources to advance general chinese embedding](#). *Preprint*, arXiv:2309.07597.

A Train/Validation/Test Splits

The data splits with number of instances used in this paper are described in Table 10.

B Training details and results for Outcome Prediction

Following are the details of different hyperparameters we use in our experiments, which as can be seen vary across datasets due mostly to dataset size.

Lang.	Platform	Data	Rows	Total
en	Wikipedia	Train	12,963	18,528
		Val	1,856	
		Test	3,709	
	Wikidata-ent	Train	248,871	355,428
		Val	35,558	
		Test	70,999	
	Wikidata-pr	Train	349	498
		Val	52	
		Test	97	
	Wikinews	Train	63	91
		Val	9	
		Test	19	
	Wikiquote	Train	484	695
		Val	69	
		Test	142	
es	Wikipedia	Train	2,291	3,274
		Val	294	
		Test	689	
gr	Wikipedia	Train	274	392
		Val	35	
		Test	83	
en	Stance	Train	372,033	437,770
		Val	21,961	
		Test	43,776	
en	Policy	Train	274,867	341,337
		Val	30,540	
		Test	35,930	

Table 10: Data distribution statistics, divided in 3 blocks: English Outcome Prediction (top), Multilingual Outcome Prediction (middle), and (English) Stance and Policy Prediction (bottom).

- Number of epochs: 20 (Outcome Prediction)/ 5 (Stance/Policy detection)
- Learning rate: 1e-5 (Outcome Prediction)/ 2e-6 (Stance/Policy detection)
- Batch size: 4
- Optimizer: Adam
- Resource used: NVIDIA RTX 4090