

An Edit-centric Approach for Wikipedia Article Quality Assessment

Edison Marrese-Taylor¹, Pablo Loyola², Yutaka Matsuo¹
Graduate School of Engineering, The University of Tokyo, Japan¹
{emarrese, matsuo}@weblab.t-utokyo.ac.jp
IBM Research, Tokyo, Japan²
e57095@jp.ibm.com

Abstract

We propose an edit-centric approach to assess Wikipedia article quality as a complementary alternative to current full document-based techniques. Our model consists of a main classifier equipped with an auxiliary generative module which, for a given edit, jointly provides an estimation of its quality and generates a description in natural language. We performed an empirical study to assess the feasibility of the proposed model and its cost-effectiveness in terms of data and quality requirements.

1 Introduction

Wikipedia is arguably the world’s most famous example of crowd-sourcing involving natural language. Given its open-edit nature, article often end up containing passages that can be regarded as noisy. These may be the indirect result of benign edits that do not meet certain standards, or a more direct consequence of vandalism attacks. In this context, assessing the quality of the large and heterogeneous stream of contributions is critical for maintaining Wikipedia’s reputation and credibility. To that end, the Wikimedia Foundation has deployed a tool named ORES (Hal-faker and Taraborelli, 2015) to help monitor article quality, which treats quality assessment as a supervised multi-class classification problem. This tool is static, works at the document-level, and is based on a set of predefined hand-crafted features (Warncke-Wang et al., 2015).

While the ORES approach seems to work effectively, considering the whole document could have negative repercussions. As seen on Figure 1, article length naturally increases over time (see Appendix A.1 for additional examples), which could lead to scalability issues and harm predictive performance, as compressing a large amount

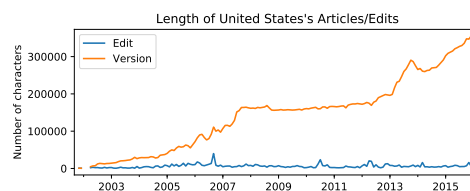


Figure 1: Monthly average of the article/edits length (in number of characters) for the United States article.

of content into hand crafted features could diminish their discriminative power. We conducted an exploratory analysis using a state-of-the-art (Dang and Ignat, 2016) document-level approach, finding that there is a clear negative relationship between document length and model accuracy (see details on Appendix A.2).

In light of this, we are interested in exploring a complementary alternative for assessing article quality in Wikipedia. We propose a model that receives as input only the edit, computed as the difference between two consecutive article versions associated to a contribution, and returns a measure of article quality. As seen on Figure 1, edit lengths exhibit a more stable distribution over time.

Moreover, as edits are usually accompanied by a short description which clarifies their purpose and helps with the reviewing process (Guzman et al., 2014), we explore whether this information could help improve quality assessment by also proposing a model that jointly predicts the quality of a given edit and generates a description of it in natural language. Our hypothesis is that while both tasks may not be completely aligned, the quality aspect could be benefited by accounting for the dual nature of the edit representation.

We performed an empirical study on a set of Wikipedia pages and their edit history, evaluating the feasibility of the approach. Our results show that our edit-level model offers competitive re-

sults, benefiting from the proposed auxiliary task. In addition to requiring less content as input, we believe our model offers a more natural approach by focusing on the actual parts of the documents that were modified, ultimately allowing us to transition from a static, document-based approach, to an edit-based approach for quality assessment. Our code and data are available on GitHub¹.

2 Related Work

In terms of quality assessment, the pioneer work of Hu et al. (2007) used the interaction between articles and their editors to estimate quality, proposed as a classification task. Later, Kittur and Kraut (2008) studied how the number of editors and their coordination affects article quality, while Blumenstock (2008) proposed to use word count for measuring article quality.

Warncke-Wang et al. (2013, 2015) took the classification approach and characterized an article version with several hand-crafted features, training a SVM-based model whose updated version was deployed into the ORES system. More recent work has experimented with models based on representation learning, such as Dang and Ignat (2016) who used a doc2vec-based approach, and Shen et al. (2017) who trained an RNN to encode the article content. While all these models are inherently static, as they model the content of a version, the work of Zhang et al. (2018) is, to the best of our knowledge, the only one to propose a history-based approach.

On the other hand, Su and Liu (2015) tackled the quality problem by using a psycho-lexical resource, while Kiesel et al. (2017) aimed at automatically detecting vandalism utilizing change information as a primary input. Gandon et al. (2016) also validated the importance of the editing history of Wikipedia pages as a source of information.

In addition to quality assessment, our work is also related to generative modeling on Wikipedia. Recent work includes approaches based on autoencoders, such as Chisholm et al. (2017), who generate short biographies, and Yin et al. (2019) who directly learn diff representations. Other works include the approach by Zhang et al. (2017) which summarizes the discussion surrounding a change in the content, and by Boyd (2018) who utilizes Wikipedia edits to augment a small dataset for grammatical error correction in German.

¹github.com/epochx/wikigen

3 Proposed Approach

Our goal is to model the quality assessment task on Wikipedia articles from a dynamic perspective. Let v_1, \dots, v_T be the sequence of the time-sorted T revisions of a given article in Wikipedia. Given a pair of consecutive revisions (v_{t-1}, v_t) , an *edit* $e_t = \Delta_{t-1}^t(v)$ is the result of applying the Unix *diff* tool over the *wikitext*² contents of the revision pair, allowing us to recover the added and deleted lines on each edition.

Due to the line-based approach of the Unix *diff* tool, small changes in *wikitext* may lead to big chunks (or hunks) of differences in the resulting *diff* file. Moreover, as changes usually occur at the sentence level, these chunks can contain a considerable amount of duplicated information. To more accurately isolate the introduced change, we segment the added and removed lines on each hunk into sentences, and eliminate the ones appearing both in the added and removed lines. Whenever multiple sentences have been modified, we use string matching techniques to identify the before-after pairs. After this process, e_t can be characterized with a set of before-after sentence pairs (s_{ti}^-, s_{ti}^+) , where s_{ti}^+ is an empty string in case of full deletion, and vice-versa.

Similarly to Yin et al. (2019), to obtain a fine-grained characterization of the *edit*, we tokenize each sentence and then use a standard *diff* algorithm to compare each sequence pair. We thus obtain an alignment for each sentence pair, which in turn allows us to identify the tokens that have been added, removed, or remained unchanged. For each case, we build an *edit-sentence* based on the alignment, containing added, deleted and unchanged tokens, where the nature of each is characterized with the token-level labels $+$, $-$ and $=$, respectively.

For a given *edit* e_t we generate an *edit* representation based on the contents of the associated *diff*, and then use it to predict the quality of the article in that time. We follow previous work and treat quality assessment as a multi-class classification task, with labels $\text{Stub} \leq \text{Start} \leq \text{C} \leq \text{B} \leq \text{GA} \leq \text{FA}$. We consider a training corpus with T edits e_t , $1 \leq t \leq T$.

Our quality assessment model encodes the input *edit-sentence* using a BiLSTM. Concretely, we use a token embedding matrix E_T to represent the input tokens, and another embedding matrix E_L

²mediawiki.org/wiki/Wikitext

to represent the token-level labels. For a given embedded token sequence X_t and embedded label sequence L_t , we concatenate the vectors for each position and feed them into the BiLSTM to capture context. We later use a max pooling layer to obtain a fixed-length edit representation, which is fed to the classifier module.

3.1 Incorporating Edit Message Information

When a user submits an *edit*, she can add a short message describing or summarizing it. We are interested in studying how these messages can be used as an additional source to support quality assessment task. A natural, straightforward way to incorporate the message into our proposal is to encode it into a feature vector using another BiLSTM with pooling, and combine this with the features learned from the *edit*.

Furthermore, we note that the availability of an edit message actually converts an *edit* into a dual-nature entity. In that sense, we would like to study whether the messages are representative constructs of the actual *edits*, and how this relation, if it exists, could impact the quality assessment task. One way to achieve this is by learning a mapping between *edits* and their messages.

Therefore, we propose to incorporate the edit messages by adding an auxiliary task that consists of generating a natural language description of a given *edit*. The idea is to jointly train the classification and the auxiliary task to see if the performance on quality assessment improves. Our hypothesis is that while both tasks are not naturally aligned, the quality aspect could benefit by accounting for the dual nature of the *edit* representation.

Our proposed generative auxiliary task is modeled using a sequence-to-sequence (Sutskever et al., 2014; Cho et al., 2014) with global attention (Bahdanau et al., 2015; Luong et al., 2015) approach, sharing the encoder with the classifier. During inference, we use beam search and let the decoder run for a fixed maximum number of steps, or until the special end-of-sentence token is generated. This task is combined with our main classification task using a linear combination of their losses, where parameter λ weights the importance of the classification loss. Figure 2 shows how the model with the auxiliary task looks like.

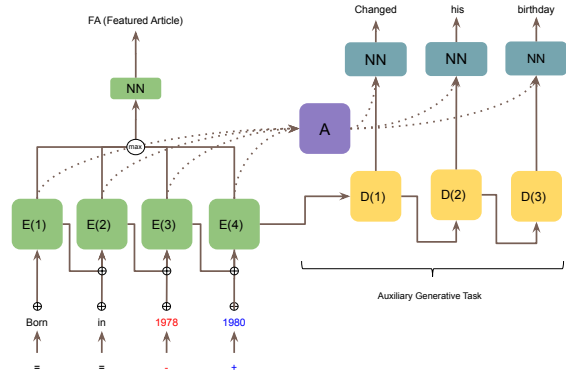


Figure 2: Proposed model with the addition of the module for the auxiliary generative task.

4 Empirical Study

We collected historical dumps from Wikipedia, choosing some of the most edited articles for both the English and German languages. Wikipedia dumps contain every version of a given page in *wikitext*, along with metadata for every *edit*. To obtain the content associated to each $\Delta_{t-1}^t(v)$, we sorted the extracted *edits* chronologically and computed the *diff* of each pair of consecutive versions using the Unix diff tool. We ignore *edits* with no accompanying message. For English sentence splitting we used the automatic approach by Kiss and Strunk (2006), and Somajo (Proisl and Uhrig, 2016) for German. The quality labels are obtained using the ORES API, the official platform of the Wikimedia Foundation to perform quality assessment. This platform is built upon a random forest classifier with 100 trees (Warncke-Wang et al., 2015), which when trained on a small corpus of 2,272 random articles obtained an F1-Score of 0.425. The API gives us a probability distribution over the quality labels for each revision, which in this work we use as a silver standard. We randomize and then split each dataset using a 70/10/20 ratio.

For comparison, we also consider the *Wikiclass* dataset built by Warncke-Wang et al. (2015), which consists of 30K revisions of random Wikipedia articles paired with their manually-annotated quality classes. To use this dataset with our models, we identified and downloaded the page revision immediately preceding each example using the Wikipedia API, to later apply the Unix *diff* tool and obtain the *edits*. We use the train/test splits provided and 20% of the training set as a validation. Other similar datasets are not

suitable for us as they do not include the revision ids which we require in order to obtain the edits.

4.1 Experimental Setting

On our collected datasets, the classification models are trained using the Kullback-Leibler divergence as the loss function—which in our preliminary experiments worked better than using the derived hard labels with cross entropy—while for the *Wikiclass* dataset we used the cross entropy with the gold standard. In both cases we used accuracy on the validation set for hyper-parameter tuning and evaluation, and also measured macro-averaged F1-Score. For the models with the auxiliary task, we also evaluate our generated descriptions with sentence-level BLEU-4 (Papineni et al., 2002). All our models are trained with a maximum input/output length of 300 tokens, a batch size of 64 and a learning rate of 0.001 with Adam.

4.2 Results

We firstly conducted an ablation study to identify the model components that have greater impact on the performance. We compare our *edit-sentence* encoder with a regular encoding mechanism, where the tokens from s_{ti}^- and s_{ti}^+ are concatenated (separated with a special marker token), and with a version that ignores the token-level label embeddings.

Model	F1	Acc	BLEU
Regular	0.47	0.74	-
+ <i>edit-sentence</i>	0.56	0.80	-
+ <i>diff</i> tags	0.62	0.78	-
+ Generation $\lambda = 0.2$	0.28	0.61	0.25
+ Generation $\lambda = 0.5$	0.33	0.68	0.24
+ Generation $\lambda = 0.8$	0.41	0.77	0.25
+ Generation $\lambda = 0.9$	0.65	0.77	0.22
Only Generation ($\lambda = 0$)	-	-	0.23

Table 1: Impact of the parameters on validation performance for the WWII article history.

As seen on Table 1, when compared against the regular encoder, utilizing our *edit-sentence* approach with token-level labels leads to a higher F1-Score and accuracy, showing the effectiveness of our proposed *edit* encoder. These results also shed some light on the trade-off between tasks for different values of λ . We see that although a higher value tends to give better classification performance both in terms of F1-Score and accuracy, it is also possible to see that there is a sweet-spot that allows the classification to benefit from learning an *edit*-message mapping, sup-

porting our hypothesis. Moreover, this comes at a negligible variation in terms of BLEU scores, as seen when we compare against a pure message-generation task (Only Generation on the table).

On the other hand, when we tested the alternative mechanism to combine the edit and message information simply combining their representations and feeding them to the classifier, we obtained no performance improvements. This again supports our choice to model the *edit*-message mapping for the benefit of quality assessment.

Since we discarded *edits* that were not accompanied by messages during pre-processing, it is difficult to assess the impact that the absence of these messages may have on quality assessment. In those cases, we believe our model with the auxiliary generative task could be used as a drop-in replacement and thus help content reviewers.

Dataset	Model	Validation			Test		
		F1	Acc	BL	F1	Acc	BL
Barack Obama	C	0.50	0.92	-	0.62	0.91	-
	C+G	0.57	0.89	0.21	0.66	0.88	0.20
Donald Trump	C	0.69	0.79	-	0.47	0.78	-
	C+G	0.69	0.76	0.22	0.47	0.77	0.20
Guns n' Roses	C	0.28	0.86	-	0.18	0.84	-
	C+G	0.31	0.79	0.23	0.30	0.81	-
Xbox 360	C	0.22	0.60	-	0.30	0.61	-
	C+G	0.37	0.62	0.34	0.32	0.63	0.31
Chicago	C	0.36	0.71	-	0.38	0.72	-
	C+G	0.44	0.70	0.30	0.39	0.71	0.29
Pink Floyd	C	0.43	0.79	-	0.35	0.80	-
	C+G	0.46	0.80	0.34	0.37	0.80	0.35
Manchester United F.	C	0.15	0.24	-	0.17	0.72	-
	C+G	0.29	0.79	0.43	0.39	0.77	0.43
Deutschland	C	0.19	0.31	-	0.12	0.31	-
	C+G	0.22	0.38	0.31	0.17	0.33	0.36
Zweiter Weltkrieg	C	0.26	0.29	-	0.15	0.30	-
	C+G	0.29	0.32	0.35	0.18	0.30	0.28

Table 2: Summary of our results. C indicates models that only perform classification, and C+G models with the auxiliary generative task. BL is short for BLEU-4.

Table 2 summarizes our best results on each selected article. We see how the addition of the generative task can improve the classification performance for both considered languages. In terms of the task trade-off, controlled with parameter λ , we empirically found that higher values tend to work better for datasets with more examples.

Regarding the *Wikiclass* dataset, we compared our model against a state-of-the-art document-level approach (Dang and Ignat, 2016; Shen et al., 2017) based on on doc2vec (Le and Mikolov, 2014). In this scenario, our model obtains an ac-

curacy of 0.40 on the test set, while the document level approach reaches 0.42. While the document level approach performed slightly better, our model is able to obtain a reasonable performance in a more efficient manner as it requires an input that averages only 2K characters (the edits), which contrasts to the average 12K characters in the documents. It is worth mentioning that the performance of the document-level approach reported by [Dang and Ignat \(2016\)](#) significantly differs from the value reported here. By looking at their implementation³ we note that this value is obtained when also using the test documents to train.

5 Conclusion and Future work

In this work we proposed a new perspective to the problem of quality assessment in Wikipedia articles, taking as central element the dynamic nature of the *edits*. Our results support our hypothesis and show the feasibility of the approach. We believe the temporal view on the problem that the proposed approach provides could open the door to incorporating behavioral aspects into the quality estimation, such as user traces and reverting activity, which are also critical to limit the amount of noise and ensure the reliability of Wikipedia.

We think our results could represent a concrete contribution in improving our understanding of the evolution knowledge bases, in terms of both software and scientific documentation, from a linguistic perspective. We envision this as a tool that could be useful for supporting documentation and quality-related tasks in collaborative environments, where human supervision is insufficient or not always available.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. [Neural machine translation by jointly learning to align and translate](#). In *Proceedings of the 2015 International Conference on Learning Representations*, San Diego, California.

Joshua E Blumenstock. 2008. Size matters: word count as a measure of quality on wikipedia. In *Proceedings of the 17th international conference on World Wide Web*, pages 1095–1096. ACM.

Adriane Boyd. 2018. [Using Wikipedia Edits in Low Resource Grammatical Error Correction](#). In *Proceedings of the 2018 EMNLP Workshop W-NUT*:

³github.com/vinhqgdang/doc2vec_dnn_wikipedia

The 4th Workshop on Noisy User-generated Text, pages 79–84, Brussels, Belgium. Association for Computational Linguistics.

- Andrew Chisholm, Will Radford, and Ben Hachey. 2017. [Learning to generate one-sentence biographies from wikidata](#). In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 633–642, Valencia, Spain. Association for Computational Linguistics.
- Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. [Learning Phrase Representations using RNN EncoderDecoder for Statistical Machine Translation](#). In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar. Association for Computational Linguistics.
- Q. Dang and C. Ignat. 2016. Quality assessment of Wikipedia articles without feature engineering. In *2016 IEEE/ACM Joint Conference on Digital Libraries (JCDL)*, pages 27–30.
- F. Gandon, R. Boyer, O. Corby, and A. Monnin. 2016. [Wikipedia editing history in dbpedia: Extracting and publishing the encyclopedia editing activity as linked data](#). In *2016 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, pages 479–482.
- Emitza Guzman, David Azócar, and Yang Li. 2014. [Sentiment analysis of commit comments in github: An empirical study](#). In *Proceedings of the 11th Working Conference on Mining Software Repositories, MSR 2014*, pages 352–355, New York, NY, USA. ACM.
- Aaron Halfaker and Dario Taraborelli. 2015. Artificial intelligence service ores gives wikipedians x-ray specs to see through bad edits.
- Meiqun Hu, Ee-Peng Lim, Aixin Sun, Hady Wirawan Lauw, and Ba-Quy Vuong. 2007. Measuring article quality in wikipedia: models and evaluation. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 243–252. ACM.
- Johannes Kiesel, Martin Potthast, Matthias Hagen, and Benno Stein. 2017. [Spatio-temporal analysis of reverted wikipedia edits](#).
- Tibor Kiss and Jan Strunk. 2006. [Unsupervised multi-lingual sentence boundary detection](#). *Comput. Linguist.*, 32(4):485–525.
- Aniket Kittur and Robert E Kraut. 2008. Harnessing the wisdom of crowds in wikipedia: quality through coordination. In *Proceedings of the 2008 ACM conference on Computer supported cooperative work*, pages 37–46. ACM.

- Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *International Conference on Machine Learning*, pages 1188–1196.
- Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. [Effective approaches to attention-based neural machine translation](#). In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, Lisbon, Portugal. Association for Computational Linguistics.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. [Bleu: a method for automatic evaluation of machine translation](#). In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.
- Thomas Proisl and Peter Uhrig. 2016. [Somajo: State-of-the-art tokenization for german web and social media texts](#). In *Proceedings of the 9th Web as Corpus Workshop (WaC-X) and the EmpiriST Shared Task*, pages 57–62, Berlin, Germany. Association for Computational Linguistics.
- Aili Shen, Jianzhong Qi, and Timothy Baldwin. 2017. [A Hybrid Model for Quality Assessment of Wikipedia Articles](#). In *Proceedings of the Australasian Language Technology Association Workshop 2017*, pages 43–52. Event-place: Brisbane, Australia.
- Q. Su and P. Liu. 2015. [A psycho-lexical approach to the assessment of information quality on wikipedia](#). In *2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, volume 3, pages 184–187.
- Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.
- Morten Warncke-Wang, Vladislav R. Ayukaev, Brent Hecht, and Loren G. Terveen. 2015. [The Success and Failure of Quality Improvement Projects in Peer Production Communities](#). In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing, CSCW '15*, pages 743–756, New York, NY, USA. ACM.
- Morten Warncke-Wang, Dan Cosley, and John Riedl. 2013. [Tell Me More: An Actionable Quality Model for Wikipedia](#). In *Proceedings of the 9th International Symposium on Open Collaboration, WikiSym '13*, pages 8:1–8:10, New York, NY, USA. ACM.
- Pengcheng Yin, Graham Neubig, Miltiadis Allamanis, Marc Brockschmidt, and Alexander L Gaunt. 2019. Learning to represent edits. In *Proceedings of the 2019 International Conference on Learning Representations*.
- Amy X. Zhang, Lea Verou, and David Karger. 2017. [Wikum: Bridging discussion forums and wikis using recursive summarization](#). pages 2082–2096.
- S. Zhang, Z. Hu, C. Zhang, and K. Yu. 2018. [History-Based Article Quality Assessment on Wikipedia](#). In *2018 IEEE International Conference on Big Data and Smart Computing (BigComp)*, pages 1–8.