NLP project report – Classification of Quality Groups of Hebrew Wikipedia Articles

Submitted by Matan Kichler
June 2024

Links

🔗**notebook on Google Colab** (for Hebrew Wikipedia)
🔗**notebook on Kaggle** (for English Wikipedia)
🔗**dataset on Kaggle**

Introduction

Wikipedia is a collaborative project managed by a community of volunteers from all walks of life. Consequently, some of the articles on Wikipedia are written by individuals who are not trained in encyclopedic writing or who cannot contribute useful content. This results in a situation where anyone can edit, change, and even maliciously vandalize articles.

The ratio of the ever-increasing number of daily edits to the limited number of experienced editors who can oversee the quality of these edits has led to a somewhat absurd situation in the Hebrew Wikipedia. There are thousands of articles of poor quality. Instances of poor edits or vandalism often go unnoticed by editors for many years.

There are some automated tools that alert patrollers (editors that follow changes) to problematic edits (usually curses), but as of now, there is no automated tool in Hebrew Wikipedia capable of analyzing the semantics and structure of entire articles to assess their quality. As a Wikipedian and ex-patroller myself, such a tool, if proven effective, would greatly assist the community of editors and improve the overall trust on Hebrew Wikipedia. While several papers (see related works) suggest leveraging machine learning algorithms and deep learning models for classifying the quality of Wikipedia articles – nothing was done for Hebrew Wikipedia.

In this report I suggest an improved method to the approach that was presented on [Wang et al., 2021], tested and compared on English Wikipedia. Later on, I applied the same approach on Hebrew Wikipedia with some adaptations to Hebrew.

Related works

Over the last two decades several papers proposed to automatically assess the quality of Wikipedia articles. Early works applied metric-based methods relied on handcrafted statistical features like the number of editors, views, edits, editor rankings ect. [Betancourt et al., 2016; Cozza et al., 2016; Dang & Ignat, 2016a, b, 2017; Zhang et al. 2018]. However, these metric-based methods showed relatively the lowest accuracies (<0.5 using 3 quality classes).

Some papers applied machine learning algorithms on these handcrafted features in order to assess some aspects of quality [Anderka, 2013; Arazy & Nov, 2010], the paper [Wang et al., 2013] achieved an accuracy of 0.543 (3 quality classes) – slight improvement compared to metric-based methods. Yet, the semantic content from the article text – which has the richest information about the quality of the article – was often ignored.

A few studies tried using basic word embeddings (without the attention mechanism) to extract semantic features, but failed to capture deeper semantics. As such, [Dang, 2016c] uses Doc2Vec encoder, with its embeddings being fed into DNN (accuracy of 0.55, see Table 2.). Most recent studies tried using advanced document embeddings, and feeding them either into deep learning models or into machine learning algorithms. [Moás & Lopes, 2023] compare recent studies using deep learning models (see Table 2.), and machine learning algorithms (see Table 1.).

Study	Best Method	Accuracy
Schmidt & Zangerle, 2019	Gradient Boosted Trees	0.73
Dang & Ignat, 2016a	Random Forest	0.64
Halfake, 2017	ORES (Gradient Boosting)	0.629
Halfaker & Geiger, 2020	Gradient Boosting	0.629
Narun et al., 2020	MLR	0.494

Table 1. Classical machine learning algorithms (accuracy of 3 quality classes)

Study	Best Method	Accuracy
Wang et al., 2021	Stacked Learning	**0.755**
Shiyue et al., 2018	RNN + LSTM	0.686
Aili et al., 2017	Bi-LSTM+	0.682
Dang & Ignat, 2017	RNN + LSTM	0.68
Edison et al., 2019	Bi-LSTM	0.666
Bhanu et al., 2020	BERT + GRU	0.632
Aili et al., 2020	BiLSTM	0.625
Aili et al., 2019	BiLSTM	0.594
Dang & Ignat, 2016c	DNN	0.55
Dang & Ignat, 2016b	DNN	0.55

Table 2. Deep Learning models (accuracy of 3 quality classes)

The previous state-of-the-art method leveraged powerful pre-trained language models like BERT and ELMo to extract semantic representations from the content. These semantic features are combined with handcrafted statistical features and fed into ensemble models containing multiple machine learning and deep learning models (“stacked learning” approach). The models were trained on English Wikipedia articles labeled into 3 “coarse” quality levels from 6 levels

based on human review; stub, start, B, GA, A, FA (increasing order of quality). stub and start articles labeled [‘Low’], B and GA labeled [‘Medium’], and, A and FA labeled [‘Good’].

Previous state-of-the-art results

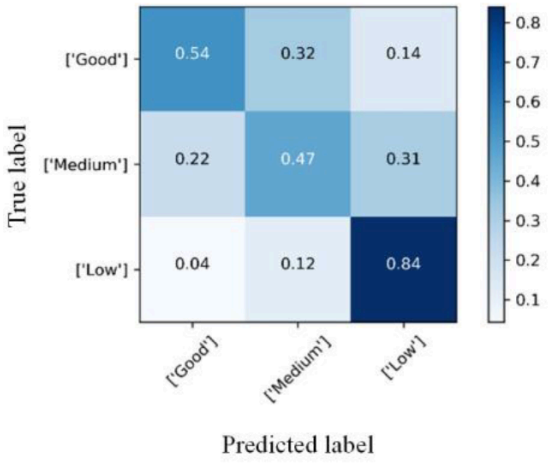

Table 3. Confusion matrix for BERT features

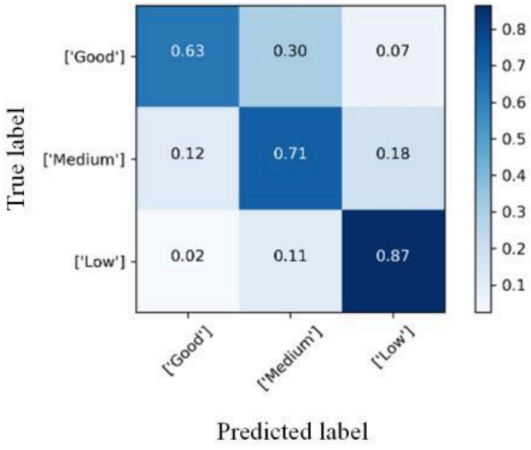

Table 4. Confusion matrix for all features

Feature sets	Precision	Recall	F1-score
All features	0.75	0.75	0.75
BERT extracted features	0.62	0.63	0.62
ELMo extracted features	0.69	0.68	0.68

Table 5. Features fusion leads to better results

Methods

My hypothesis is that the textual input data alone with the right embedding and classifier, can achieve the greatest results compared to previous methods that used meta-information about the articles (handcrafted features). While parameters like the number of edits or editors might cast some light upon the quality of the article – whenever an article is being discussed whether it has to be considered as a feature article, the discussion is primarily focused on the textual content, i.e. its readability, comprehensiveness, style etc.[1], almost ignoring handcrafted features, as opposed to the doctrine presented in previous works.

As far as I know, longformers [Beltagy et al., 2020] weren’t tested for classification of quality of wikipedia articles. While BERT [Devlin et al., 2019] captures the context of 512 tokens, roughly the first paragraph in each article, longformers capture the context of 4096 tokens, which is on average most of the article. Longformers do better on classification tasks involving long documents [Beltagy et al., 2020].

[1] Featured article criteria on English Wikipedia:
https://en.wikipedia.org/wiki/Wikipedia:Featured_article_criteria

The results in [Wang et al., 2021] raised the idea of concatenating the embeddings of more than one encoder model (so called features fusion) into one long vector, which is sent into the classification model. They have tested both BERT and ELMo embeddings, separately. Concatenating both archived much better results (See Table 5.).

Inspired by this idea, I suggest concatenating the embedding from pre-trained models LongHeRo [Shalumov & Haskey, 2023] with that of AlephBERT [Seker et al., 2021]. In this way, I capture both narrow context using BERT, and wide context using a longformer. In this report, my hypothesis is to get state-of-the-art results using the concatenating of both.

Each embedding is of dimension 768, after concatenating the dimension is 1536. It is then sent into a standartic DNN with fully connected input layer of size 1536, 2 hidden fully connected layers, first of size 512, and the second of size 256, and output layer with softmax of size 3, the number of classes. I used on each layer the Relu activation function, which is a decent choice for learning nonlinear functions.

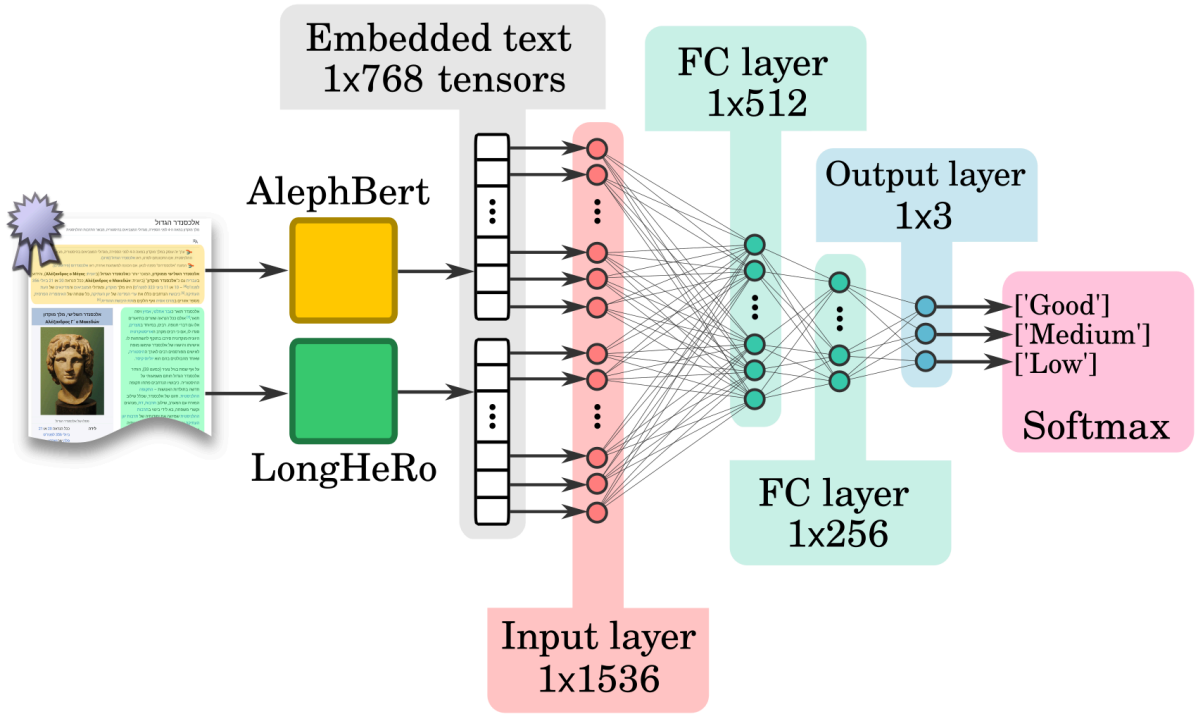

Fig 1. My proposed architecture scheme

In order to make a comparison with the results of the state-of-the-art method presented in [Wang et al., 2021], I run my proposed architecture on their dataset using the original BERT and longformer embedding. Apart from that, we focus on the Hebrew wikipedia articles classification in this report.

Since [Schmidt & Zangerle, 2019] achieved high accuracies using Gradient Boosted Trees (XGBoost) without the attention mechanism, I also test this algorithm on my embeddings instead of the proposed DNN for comparison.

Experimental results

Dataset

I have created 2 datasets[2], one for Hebrew Wikipedia and the other for English Wikipedia in order to compare my results with previous studies. Articles of both Wikipedias were gathered using Wikipedia API[3], which gives a textual string clean of Wiki source code or any "magic words" that indicate the quality of an article.

For Hebrew Wikipedia the dataset was divided into three categories: featured articles (see Figure 3.), medium-quality articles, and poor articles (with 'cleanup' template, see Figure 2.), denoted by ['Good'], ['Medium'] and ['Low'] respectively. Since the smallest group among these is the featured articles with only 727 articles (to this day), I gathered about the same number of articles from the other classes; 726 poor articles and 763 medium quality articles. I picked them randomly.

Label	Class	Criteria
0	['Low']	There are over 3000 articles labeled as such. These articles have a template at the top indicating that they need to be rewritten. This template is used to highlight significant content issues, which can include poor organization, non-encyclopedic writing style, extensive machine translation, lack of neutrality, inappropriate language level, excessive detail, and more.
1	['Medium']	These are articles without any quality-related templates. To ensure they are not poor-quality articles, I selected articles that have surpassed a certain threshold of views and edits
2	['Good']	There are 727 such articles. These are the best and highest-quality articles Hebrew Wikipedia has to offer. They emphasize both content and aesthetics, presenting topics clearly, accurately, and up-to-date.

Table 6. Hebrew Wikipedia Quality groups

For the English Wikipedia dataset, I followed [Wang et al., 2013] collecting 400 articles from FA, 400 articles from GA, 336 articles from A, 284 articles from B, 284 articles from Start, and 284 articles from Stub. With a total number of 1988 articles.

Metric Evaluation

For the evaluation, I conducted a 5-fold cross-validation for the proposed approach, consistent with previous studies. I randomly divided the dataset into five folds and repeated the evaluation five times, with each fold serving as the test dataset once. The results presented in this report (see Results and Analysis) are the averages of these five folds.

---

[2] Link to datasets: https://www.kaggle.com/datasets/matankic/quality-groups-of-hebrew-wikipedia-articles

[3] I replaced '*article*' with the title of a given article, each time, and scraped the 'extract' of the JSON https://en.wikipedia.org/w/api.php?format=json&action=query&prop=extracts&exlimit=max&explaintext&titles=*article*

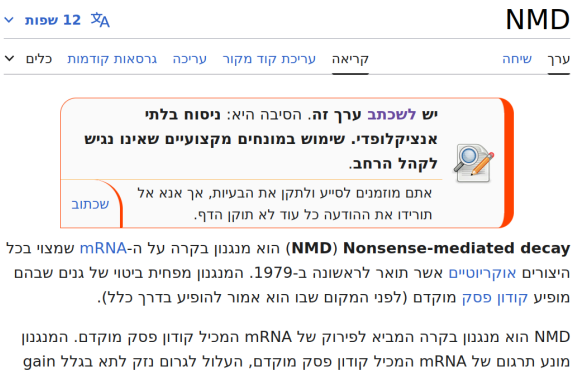

Fig 2. poor article with a ‘cleanup’ template

Fig 3. Featured article with a rosette icon

Regarding the evaluation metrics, I used accuracy, in line with previous studies. Additionally, I provided precision and F1-score for each of the quality groups. I compared the results of the English Wikipedia dataset with those of previous studies.

Results and Analysis

English wikipedia

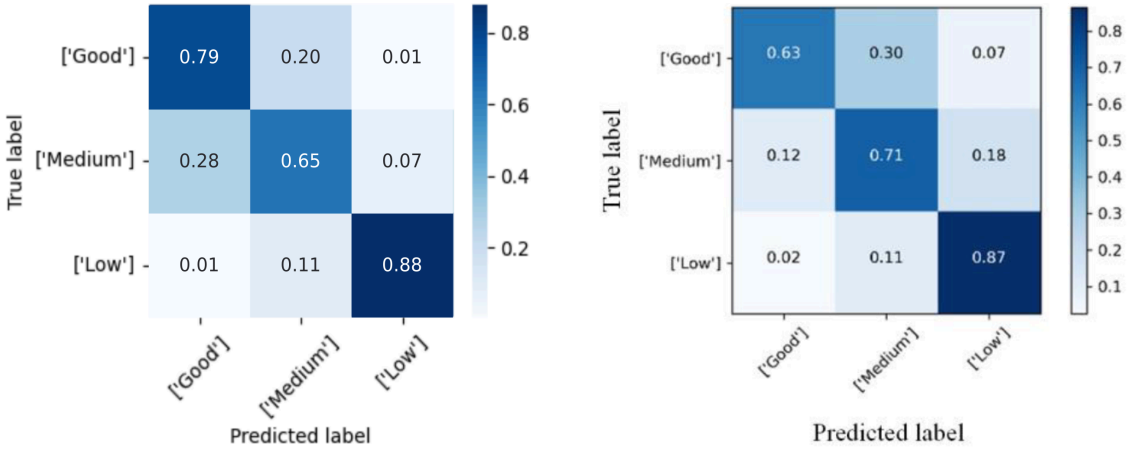

Fig 4. Confusion matrix for my proposed architectures (left) compare to previous SOTA (right)

Class	Precision	Recall	F1-score
['Low']	0.90	0.88	0.888
['Medium']	0.68	0.648	0.664
['Good']	0.746	0.79	0.766
Macro avg	0.778	0.776	0.772
Weighted avg	0.765	0.765	0.765

Table 7. Result of my proposed architectures. Accuracy of 0.765

I achieved the state-of-the-art accuracy of 0.765, compared to 0.755 of the previous state-of-the-art [Wang et al., 2021].

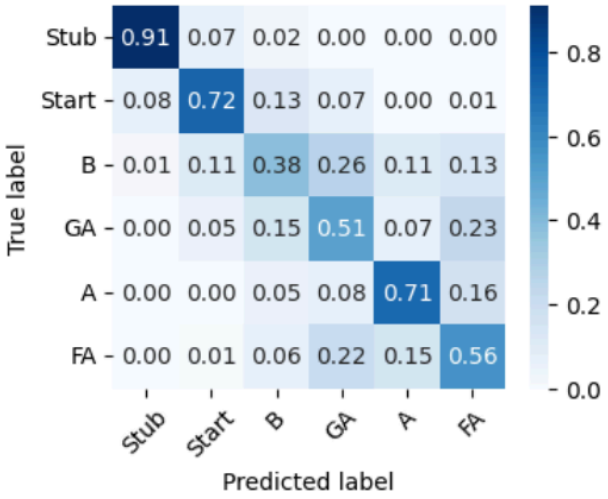

Table 8. Confusion matrix for my proposed architectures with 6-classes (before they were merged into 3 coarse classes)

Hebrew wikipedia

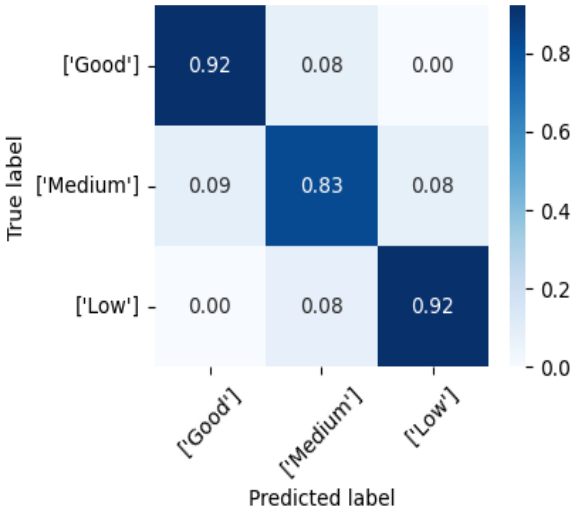

Table 9. AlephBERT + LongHeRo embedding feed into DNN confusion matrix

Class	Precision	Recall	F1-score
['Low']	0.89	0.92	0.90
['Medium']	0.83	0.76	0.79
['Good']	0.87	0.93	0.90
Macro avg	0.87	0.87	0.87
Weighted avg	0.87	0.87	0.86

Table 10. LongHeRo embedding feed into DNN

Class	Precision	Recall	F1-score
['Low']	0.92	0.92	0.92
['Medium']	0.85	0.83	0.84
['Good']	0.91	0.92	0.92
Macro avg	0.89	0.89	0.89
Weighted avg	0.89	0.89	0.89

Table 11. AlephBERT + LongHeRo embedding feed into DNN

The results on Hebrew Wikipedia are much higher compared to the accuracy on English Wikipedia. This might be due to the fact that the division of the three classes in Hebrew is much clearer. While on English Wikipedia, six fine classes were merged into three coarse classes, on Hebrew Wikipedia, I focused on more trichotomous classes. FA, A, and GA might be very similar, and the fact that an article is labeled GA ('[medium]' in our three-class results) might be due to technical issues unrelated to the text content itself. Also, the '[Low]' class on Hebrew Wikipedia is not equivalent to Start and Stub on English Wikipedia. On Hebrew Wikipedia, it deals with well-known issues ('cleanup' template, see Figure 2.), while Start and Stub articles might be well-written but short, for instance.

Features and classifier ablation

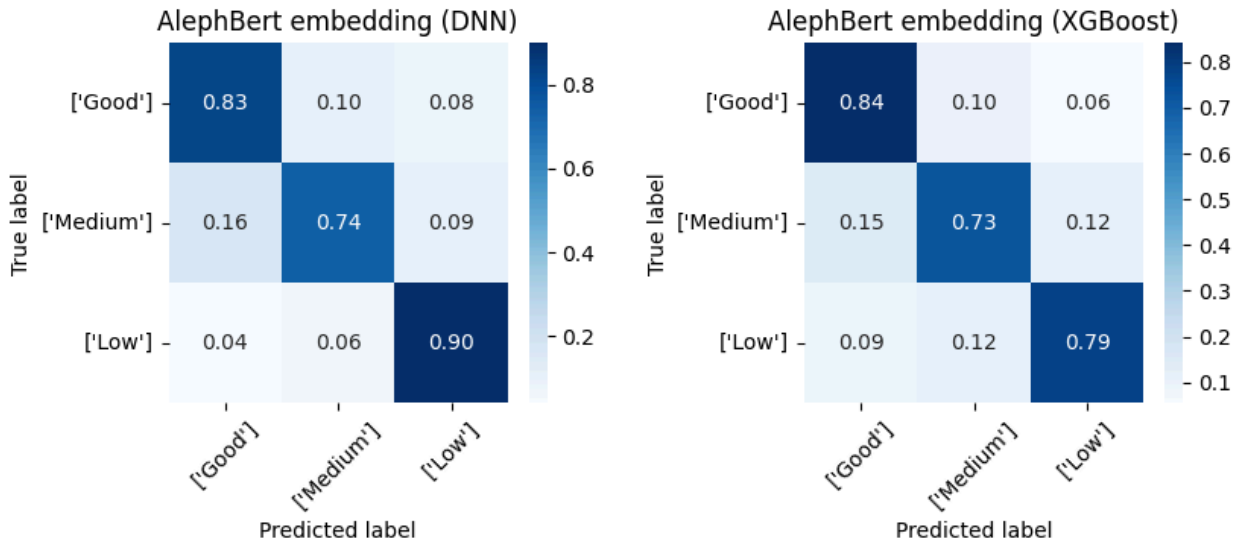

Fig 11. Ablation of LongHeRo embedding

Considering the ablation of the embedding method, we clearly see that LongHeRo gives much better results. As for the classifier, while XGBoost gives great results, my proposed DNN accuracies are better (see Fig 11., 12.)

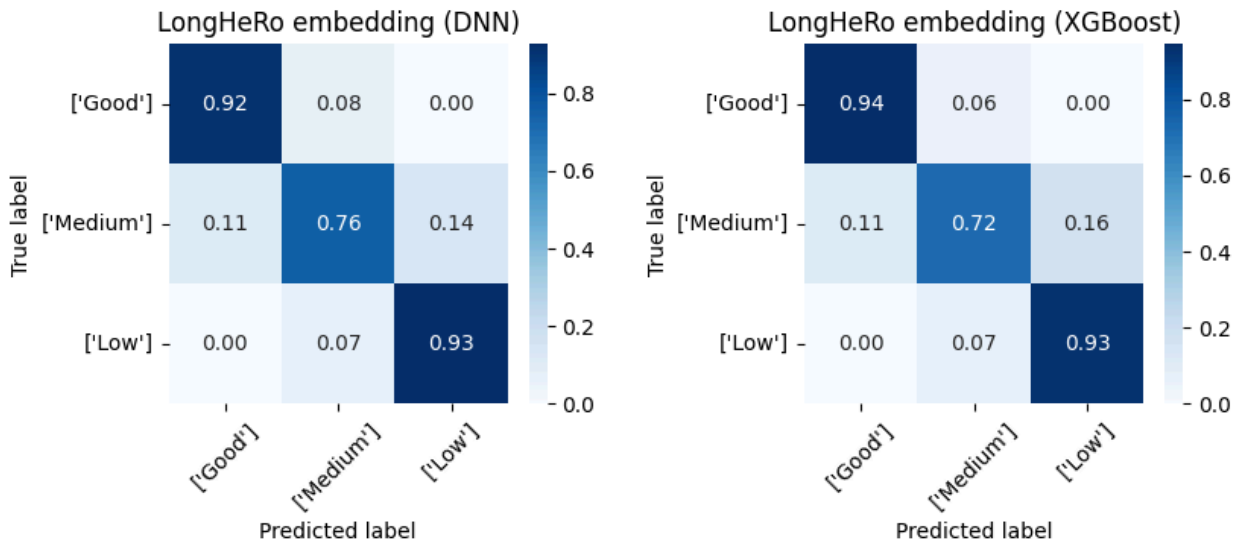

Fig 12. Ablation of HeBert embedding

Conclusion and Discussion

While my proposed model keeps being relatively simple, it succeeds in achieving the new state-of-the-art results. These results are primarily due to the wide context that LongHeRo are capable of achieving, compared to previous studies. Concatenating AlephBERT enables one to capture a narrow context that concentrates on the very first paragraph, which by itself is a fair indicator of the article's quality, since it gives the very first impression (attention) and has to follow very strict guidelines.

My hypothesis was true – Indeed, the textual input data alone achieved the greatest results compared to previous methods that used meta-information.

This architecture can be easily deployed and adapted on multiple Wikipedias of different languages, as I did in this report, using the Hebrew pre-trained equivalent to BERT and longformers.

For future work I suggest the use of powerful open-source large language models (LLM) such as Llama 3, and fine-tuning them with a few shots of different quality groups of articles. No work was done leveraging the powerful utilities of these kinds of LLMs on this subject.

References

Betancourt, G. G., Segnine, A., Trabuco, C., Rezgui, A., & Jullien, N. (2016). Mining team characteristics to predict Wikipedia article quality. In Proceedings of the 12th International Symposium on Open Collaboration, August 2016, Berlin, Germany, pp. 1-9.

Cozza, V., Petrocchi, M., & Spognardi, A. (2016). A matter of words: NLP for quality evaluation of wikipedia medical articles. In Web Engineering, June 2016, Lugano, Switzerland, pp. 448-456.

Dang, Q. V., & Ignat, C. L. (2016b). Quality assessment of wikipedia articles: A deep learning approach by Quang Vinh Dang and Claudia-Lavinia Ignat with Martin Vesely as coordinator. ACM SIGWEB Newsletter(Autumn), 5, 1-6.

Dang, Q., & Ignat, C. (2016c). Quality assessment of Wikipedia articles without feature engineering. In JCDL '16: ACM/IEEE Joint Conference on Digital Libraries. Association for Computing Machinery, New York Citys, 27–30.

Dang, Q. V., & Ignat, C. L. (2017). An end-to-end learning solution for assessing the quality of Wikipedia articles. In Proceedings of the 13th International Symposium on Open Collaboration, August 2017, Galway, Ireland, pp. 1-10.

Dang, Q., & Ignat, C. (2016a). Measuring quality of collaboratively edited documents: The case of wikipedia. In 2016 IEEE 2nd international conference on collaboration and internet computing (CIC), November 2016, Pittsburgh, PA, USA, pp. 266-275.

Zhang, S., Hu, Z., Zhang, C., & Yu, K. (2018). History-based article quality assessment on wikipedia. In 2018 IEEE international conference on big data and smart computing (BigComp), January 2018, Shanghai, China, pp. 1-8.

Anderka, M. (2013). Analyzing and predicting quality flaws in user-generated content: The case of wikipedia (Ph. D). Bauhaus-Universitaet Weimar Germany.

Arazy, O., & Nov, O. (2010). Determinants of wikipedia quality: The roles of global and local contribution inequality. In Proceedings of the 2010 ACM conference on Computer supported cooperative work, February 2010, Savannah, Georgia,USA, USA, pp. 233–236.

Warncke-Wang, M., Cosley, D., & Riedl, J. (2013). Tell me more: an actionable quality model for Wikipedia. In *Proceedings of the 9th International Symposium on Open Collaboration*. Association for Computing Machinery.

Moás, P., & Lopes, C. (2023). Automatic Quality Assessment of Wikipedia Articles—A Systematic Literature Review. *ACM Comput. Surv., 56(4)*.

Manuel Schmidt and Eva Zangerle. (2019). Article quality classification on Wikipedia: Introducing document embeddings and content features. In OpenSym '19: International Symposium on Open Collaboration. Association for Computing Machinery, New York City, 1–8.

Aaron L. Halfaker. (2017). Interpolating quality dynamics in Wikipedia and demonstrating the Keilana effect. In OpenSym '17: International Symposium on Open Collaboration. Association for Computing Machinery, New York City, 1–9.

Aaron L. 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 (2020), 1–37.

Narun K. Raman, Nathaniel Sauerberg, Jonah Fisher, and Sneha Narayan. (2020). Classifying Wikipedia article quality with revision history networks. In OpenSym '20: International Symposium on Open Collaboration. Association for Computing Machinery, New York City, 1–7.

Aili Shen, Jianzhong Qi, and Timothy Baldwin. (2017). A hybrid model for quality assessment of Wikipedia articles. In ALTA '17: Australasian Language Technology Association Workshop. ACL Anthology, Online, 43–52.

Bhanu Prakash Reddy Guda, Sasi Bhusan Seelaboyina, Soumya Sarkar, and Animesh Mukherjee. (2020). NwQM: A neural quality assessment framework for Wikipedia. In EMNLP '20: Conference on Empirical Methods in Natural Language Processing. ACL Anthology, Online, 8396–8406.

Aili Shen, Bahar Salehi, Jainzhong Qi, and Timothy Baldwin. (2020). A multimodal approach to assessing document quality. Journal of Artificial Intelligence Research 68 (2020), 607–632

Aili Shen, Bahar Salehi, Timothy Baldwin, and Jianzhong Qi. (2019). A joint model for multimodal document quality assessment. In JCDL '19: Joint conference on digital libraries. Association for Computing Machinery, New York City, 107–110.

Edison Marrese-Taylor, Pablo Loyola, and Yutaka Matsuo. (2019). An edit-centric approach for Wikipedia article quality assessment. In WNUT '19: Workshop on Noisy User-generated Text. ACL Anthology, Online, 381–386.

Shiyue Zhang, Zheng Hu, Chunhong Zhang, and Ke Yu. (2018). History-based article quality assessment on Wikipedia. In BIGCOMP '18: International Conference on Big Data and Smart Computing. Institute of Electrical and Electronic Engineers, New York City, 1–8.

Iz Beltagy, Matthew E. Peters, & Arman Cohan. (2020). Longformer: The Long-Document Transformer.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, & Kristina Toutanova. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.

Vitaly Shalumov, & Harel Haskey. (2023). HeRo: RoBERTa and Longformer Hebrew Language Models.

Amit Seker, Elron Bandel, Dan Bareket, Idan Brusilovsky, Refael Shaked Greenfeld, & Reut Tsarfaty. (2021). AlephBERT:A Hebrew Large Pre-Trained Language Model to Start-off your Hebrew NLP Application With.