Quality Flaws Prediction in Wikipedia by Using Deep Learning Approaches

Gianfranco Capodici[1], Gerónimo Bazán Pereyra[1], Rodolfo Bonnin[1], and Edgardo Ferretti[1,2]

[1] Universidad Nacional de San Luis (UNSL), San Luis - Argentina
[2] Laboratorio de Investigación y Desarrollo en Inteligencia Computacional (UNSL)
e-mail: ferretti@unsl.edu.ar

Abstract. Quality flaws prediction in Wikipedia is an ongoing research trend. In particular, in this work we tackle the problem of automatically predicting four out of the ten most frequent quality flaws; namely: *No footnotes*, *Notability*, *Primary Sources* and *Refimprove*. Different deep learning state-of-the-art approaches were evaluated on the test corpus from the 1st *International Competition on Quality Flaw Prediction in Wikipedia*; a well-known uniform evaluation corpus from this research field. Particularly, the results show that *TabNet* reachs or improves the existing benchmarks for the *Notability* and *Refimprove* flaws, and performs in a very competitive way for the other two remaining flaws.

Keywords: Wikipedia, Information Quality, Quality Flaws Prediction, Deep Learning

1 Introduction

The evaluation of the information quality (IQ) on the Web has become a crucial task today, since entities from different areas make decisions on the information available on this source. In turn, the amount of information has increased exponentially, and in part, this is due to the growing popularity of websites that allow ordinary users to generate content very easily. The latter has driven the need to automate the evaluation of the quality of information on the Web. Wikipedia is one of the best examples we have from these sites. It is a free content encyclopedia, generated from the contributions of millions of registered and anonymous users. These users write, correct and edit articles; and they are heterogeneous in aspects such as: their education level, age, culture, writing skills and specialization area. This fact makes this encyclopedia one of the 20 most visited sites in the world, but at the same time, it generates the challenge of finding a way to automatically improve the IQ of its articles; viz. a multi-dimensional concept which combines criteria such as accuracy, reliability and relevance.

A widely accepted interpretation of IQ is the "fitness for use in a practical application", i.e. the assessment of IQ requires the consideration of context and use case. Particularly, in Wikipedia the context is well-defined by the encyclopedic genre, that forms the ground for Wikipedia's IQ ideal, within the so-called

featured article criteria.[3] Having a formal definition of what constitutes a high-quality article, i.e. a featured article (FA), is a key issue; however, as indicated in [1], in 2012 less than 0.1% of the English Wikipedia articles were labeled as featured. At present, this ratio still remains, since there are 6 114 featured articles out of 6 525 174 articles on the English Wikipedia.[4]

In the literature, a variety of approaches have been proposed to automatically assess different quality aspects in Wikipedia, such as: featured articles identification; development of quality measurement metrics; vandalism detection, among others. In particular, in this paper we will concentrate on the quality flaws prediction research trend [2–10], since this approach provides concrete hints for human editors about what has to be fixed in order to improve articles' quality. The detection of quality flaws is based on user-defined cleanup tags, which are commonly used in the Wikipedia community to tag content that has some shortcomings. Thus, the tagged articles serve as human-labeled data that is exploited by a machine learning (ML) approach to predict flaws in untagged articles.

This paper extends [7] by doing a deeper study on the Deep Neural Networks (DNN) and Stacked-LSTM models previouslty evaluated on this work and by also exploring *TabNet* [11], a novel high-performance and interpretable canonical deep tabular data learning architecture, that to the best of our knowledge, has not been previously studied in the Wikipedia domain of quality flaws prediction.

The rest of the article is organized as follows. Section 2 introduces the context of the problem faced in this work. Then, in Sect. 3, we present the formal problem statement and the different prediction approaches evaluated are briefly described. Also, the document model used to represent the articles is discussed. Section 4 reports on the experimental setting carried out and the obtained results. Finally, Sect. 5 offers the conclusions.

2 Related Work

In 2012, the first exhaustive study of quality flaws for the English Wikipedia [1] gave rise to the generation of a well-formed data set (for its use in IQ research by the scientific community related to Wikipedia), in the context of the 1st *International Competition on Quality Flaw Prediction in Wikipedia* [12]. That same year, in the international competition "ImageNet Large-Scale Visual Recognition Challenge (ILSVRC)", AlexNet [13] –a system based on Deep Convolutional Neural Networks (DCNN)– emerged as the broad winner. In this way, since 2012, deep learning approaches are consolidated as the state of the art in the field of visual recognition and then spread their supremacy to other ML fields as well.

In this respect, according to our literature review, we can observe that from 2012 to middle 2021, the state of the art regarding IQ in Wikipedia has been mostly determined by research works that use classical approaches ([3–8, 10]). The differences between these works are mainly found in the applied classification

[3] `http://en.wikipedia.org/wiki/Wikipedia:Featured_article_criteria`

[4] `https://en.wikipedia.org/wiki/Wikipedia:Featured_articles` (accessed June 2022)

algorithms (semi-supervised or supervised), the underlying document representation model (number of features, their complexity and conceptualization made of each flaw, among others). Having this great diversity, it makes difficult to establish a conceptual comparison on which approach is the state of the art to be improved.

For example, [3–7, 10] have followed working methodologies close to the original one proposed by Anderka et al. [2]. In [3], the quality flaw prediction task was faced as a one-class classificacion problem and in [4], the same document model used in [3] was evaluated on the corpus from the "1st International Competition on Quality Flaw Prediction in Wikipedia", where a modified version of the PU-learning winning approach was proposed. The obtained results showed an improvement of 18.31%, averaged over the ten flaws. From among the ten flaws of the competition, the so-called *Refimprove* flaw –which alerts that the tagged article needs additional citations for verification–, has been particularly studied in [5–7]. It is worth mentioning that this information quality flaw, ranks among the five most frequent flaws and represents 12.4% of the flawed articles in the English Wikipedia [3].

In particular, [6] and [7] use the same document model proposed by Anderka [3] and these works were also evaluated on the corpus from the 1st international competition mentioned above. In [6], three different state-of-the-art binary approaches were used with the aim of handling the existing imbalances between the number of articles' tagged as flawed content, and the remaining untagged documents that exist in Wikipedia. These approaches were under-bagged decision trees, biased-SVM and centroid-based balanced SVM. The results showed that under-bagged decision trees with the *min* rule as aggregation method, perform best achieving an F_1 score of 0.96 for the *Refimprove* flaw. In addition, [7] extends the work performed in [6] by incorporating deep neural methods to the study (DNN and Stacked-LSTM) and tackles other quality flaws as well. Stacked-LSTM performed well and reached the existing benchmark for the *Refimprove* flaw. For the other flaws (*No footnotes*, *Notability*, *Primary Sources* and *Wikify*), under-bagged decision trees with different aggregation rules perform best.

Finally, regarding the aforementioned original methodology proposed by Anderka et al., we found that [10] studies different ML approaches, both traditional and deep learning methods; all using as learning experience manually constructed document models and/or automatically extracted features. From among the 12 studied classifiers, the deep approach Bi GRU (bidirectional gated recurrent unit) is the one that achieved the best classification performance: $F_1 = 0.99$ for *Notability*, *No Footnotes* and *Refimprove* flaws; and $F_1 = 0.98$ for *Primary Sources*. It is important to note that the classification approach addressed in this work, is the so-called optimistic approach by Anderka et al. [2], which uses FAs as negative class, while the approach of the competition is the so-called pessimistic, and therefore more challenging.

To be best of our knowledge, Anderka's document model [3] and the one proposed in [14], are the most comprehensive document models built so far based on a features engineering approach. In particular, Bassani and Viviani document

model [14] is composed of 264 features and in principle it seems to contain the 95 features from Anderka's document model. They evaluated their model with the aim of building a suitable ground truth for a (single-label) multi-class classification task, where each article is assigned exactly to one of the seven classes from the quality grading scheme that Wikipedia employed at the time of that paper writing.[5] They evaluated eight state-of-the-art classifiers and Gradient Boosting performed best achieving and accuracy of 90% in some experiments.

A similar classification problem to that reported in [14] was evaluated by Zhang et al. [15], since that a 6-class classification task was performed –considering the Wikipedia quality grading scheme mentioned above–, but where `AC` was skipped on the grounds that is not a real quality class and it overlaps with `FA` and `GA` classes. The proposed history-based article quality assessment model combines feature engineering with learned features by a Recurrent Neural Network (RNN); and it only contains 16 features. Zhang et al. argue that this can be one of the reasons why the best-achieved accuracy value rounds 69%.

In [16], the same 6-class classification task performed in [15] was tackled but with a different document model that relies on explicitly defined features. Moreover, as classification method it was used XGBoost. Furthermore, a deep learning-based baseline was used for assessing the performance of XGBoost given the same feature set. In this respect, the accuracy achieved by XGBoost was 73% against 67% of the deep learning-based baseline. Additionally, XGBoost was also compared against the RNN-LSTM evaluated by Dang and Ignat in [17], where the classification of Wikipedia articles in English, French, and Russian languages in different quality grading schemes was promising without the need of a feature extraction phase. In particular, for the English dataset, XGBoost outperformed the RNN-LSTM by 5%; i.e. RNN-LSTM achieved an accuracy of 68%.

Finally, in [18], following a feature engineering approach to build articles' document models –composed of 68 features–, Wang and Li present a comparative study of state-of-the-art deep-learning approaches by distinguishing high quality articles from low quality. With this aim, a 6-class classification problem on the Wikipedia quality grading scheme mentioned above, was reduced to a binary classification problem where the high-quality class includes `FA`, `AC` and `GA`; and the low-quality class includes `BC`, `SC` and `SB`. Stacked-LSTM networks achieved the best performance ($F_1 = 0.8$). Also, the influence of different features and feature sets on the proposed models were extensively investigated.

3 Problem Statement and Flaw Prediction Approaches

We start with a formal definition of the problem faced in this paper, namely the algorithmic prediction of quality flaws in Wikipedia (Section 3.1). We then provide the theoretical background of the flaw prediction approaches used in our work (Section 3.2) and finally, we introduce the document model used to represent articles (Section 3.3).

[5] At present, this quality grading scheme has been refined; cf. `https://en.wikipedia.org/wiki/Template:Grading_scheme`

3.1 Problem Statement

Following [3], quality flaw prediction is treated here as a classification problem. Let D be the set of English Wikipedia articles and let f_i be the specific quality flaw that may occur in an article $d \in D$. Let $\mathbf{d}$ be the feature vector representing article d, called document model, and let $\mathbf{D}$ denote the set of document models for D. Hence, for flaw f_i, a specific classifier c_i is learned to decide whether an article d suffers from f_i or not; that is, $c_i : \mathbf{D} \rightarrow \{1, 0\}$. For flaw f_i a set $D_i^+ \subset D$ is available, which contains articles that have been tagged to contain f_i (so-called *labeled* articles). However, no information is available about the remaining articles in $D \setminus D_i^+$—these articles are either flawless or have not yet been evaluated with respect to f_i (so-called *unlabeled* articles).

As originally proposed (see e.g. [2, 3]) c_i is modeled as a one-class classifier, which is trained solely on the set D_i^+ of labeled articles. However, in the Wikipedia setting, the large number of available unlabeled articles may provide additional knowledge that can be used to improve classifiers training. Thus, addressing the problem of exploiting unlabeled articles to improve the performance of c_i lead us to cast the problem as a binary classification task.

3.2 Flaw Prediction Approaches

Despite its theoretical one-class nature, quality flaw prediction has been tackled in prior studies as a binary classification task –which relates to the realm of supervised learning– and the results achieved in practice have been quite competitive [5–7]. Supervised learning deals with the situation where training examples are available for all classes that can occur at prediction time. In *binary classification*, the classification $c_i(\mathbf{d})$ of an article $d \in D$ with respect to a quality flaw f_i is defined as follows: given a sample $P \subseteq D_i^+$ of articles containing f_i and a sample $N \subseteq (D \setminus D_i^+)$ of articles not containing f_i, decide whether d belongs to P or to N. The binary classification approach tries to learn a class-separating decision boundary to discriminate between P and a particular N. In order to obtain a sound flaw predictor, the choice of N is essential. N should be a representative sample of Wikipedia articles that are flawless regarding f_i.

ANN An Artificial Neural Network (ANN) is just a collection of units (mathematical model that it simply "fires" when a linear combination of its inputs exceeds some hard or soft threshold; that is, it implements a linear classifier) connected together; the properties of the network are determined by its topology and the properties of the "neurons". In this work, we will refer as an ANN, a feed-forward network; that is, every unit receives inputs from "upstream" units and delivers output to "downstream" units; there are no loops —like in the case of Recurrent Neural Networks [19]. A feed-forward network represents a non-linear function of its current input; thus, it has no other internal state than the weights themselves.

DNN As stated in [20], the quintessential example of a deep learning model is the feedforward deep network (DNN), or multilayer perceptron; that is an ANN with more than one hidden layer. The input of the model is presented to the so-called "input layer", because it contains the variables that we are able to observe. Then a series of hidden layers extracts increasingly abstract features from the input. These layers are called "hidden" because their values are not given in the data; instead the model must determine which concepts are useful for explaining the relationships in the observed data.

Stacked-LSTM Long short-term memory (LSTM) [21] are a modification of the original Recurrent Neural Networks, which includes three types of gates: the forget gate, the input gate, and the output gate. The original LSTM model is comprised of a single hidden LSTM layer followed by a standard feedforward output layer. Stacked-LSTM model [22] extends the reach of this type of network, to the realm of deep neural architecture, in that it has multiple hidden LSTM layers where each layer contains multiple memory cells. Every LSTM in the stack obtains all the information from the preceding layer only.

TabNet It is a recently proposed canonical DNN architecture for tabular data [11]. It inputs raw tabular data without any preprocessing and is trained using gradient descent-based optimization, enabling flexible integration into end-to-end learning. TabNet uses sequential attention to choose which features to reason from at each decision step, enabling interpretability and better learning as the learning capacity is used for the most salient features. That is, feature selection is instance-wise, given that it can be different for each input.

3.3 Document Model

To model the articles, we used the document model proposed in [3], one of the most comprehensive document model proposed so far for quality flaw prediction in Wikipedia—it comprises 95 article features. Formally, given a set $D = \{d_1, d_2, \ldots, d_n\}$ of n articles, each article is represented by 95 features $F = \{f_1, f_2, \ldots, f_{95}\}$. A vector representation for each article d_i in D is defined as $d_i = (v_1, v_2, \ldots, v_{95})$, where v_j is the value of feature f_j. A feature generally describes some quality indicator associated with an article.

In [3] four such subsets were identified by organizing the features along the dimensions *content*, *structure*, *network* and *edit history*. Content features are computed based on the plain text representation of an article and mainly address aspects like writing style and readability. Structure features rely on an article's wiki markup and are intended to quantify the usage of structural elements like sections, templates, tables, among others. Network features quantify an article's connectivity by means of internal and external links. Edit history features rely on an article's revision history and model article evolution based on the frequency and the timing of edits as well as on the community of editors. In [3], a detailed description for each feature is provided including implementation details. Due to space constraints, these features are not explicitly described in this paper.

Table 1. Four out of the top ten quality flaws of English Wikipedia articles that are comprised in the PAN-WQF-12 corpus.

Flaw name	Flaw description	Training corpus		Test corpus	
		tagged articles	untagged articles	tagged articles	untagged articles
No footnotes	The article's sources are unclear because of its inline citations.	6 068	–	1 000	1 000
Notability	The article does not meet the general notability guideline.	3 150	–	1 000	1 000
Primary sources	The article relies on references to primary sources.	3 682	–	1 000	1 000
Refimprove	The article needs additional citations for verification.	23 144	–	999	999
Additional random (untagged) articles		–	50 000	–	–

4 Experiments and Results

To perform our experiments, we have used the corpus available in the above-mentioned Competition on Quality Flaw Prediction in Wikipedia [12], which has been released as a part of PAN-WQF-12,[6] a more comprehensive corpus related to the ten most important article flaws in the English Wikipedia, as pointed out in [1]. The training corpus of the competition contains 154 116 tagged articles (not equally distributed) for the ten quality flaws, plus additional 50 000 untagged articles. The test corpus (19 010 articles) contains a balanced number of tagged articles and untagged articles for each of the ten quality flaws, and it is ensured that 10% of the untagged articles are FAs. Table 1 introduces a brief description for each flaw evaluated in our work. Moreover, for each flaw, the numbers of tagged and untagged articles in the training and test corpus of the 2012-competition is specified. The training corpus does not contain untagged articles for the individual flaws, but it comprises 50 000 additional randomly selected untagged articles.

4.1 Experimental Setting

As mentioned in Sect. 1, in this paper we extend [7], where an initial study on deep learning approaches applied to quality flaws prediction in the Wikipedia domain was carried out. In that work, also classical (non-neural) approaches were evaluated and were in fact, the ones which reported in general the best

[6] The corpus is available at `https://webis.de/data/pan-wqf-12.html`

performing measures, except for Stacked-LSTM that reached the existing benchmark for the *Refimprove* flaw of $F_1 = 0.96$ from [6]. In [7], only *random search* (RS) over the different variables that influence each model was evaluated. In our current work, we have also tried two other search strategies, viz. *HyperBand* (HB) and Bayesian optimization (BO) for the DNN and Stacked-LSTM models, maintaining the same number of epochs (10) for training. However (cf. Table 2), only for *Notability* flaw BO or HB performed better than RS for all the models. We conjecture that this may be due to the low number of epochs –given the computational cost of each trial–, this number is not large enough to allow these more sophisticated methods to show some more advantageous parametric configurations.

Moreover, we also extend [7] by evaluating TabNet, a deep-learning achitecture specially suited for tabular data, as it is our case. We also evaluated an ANN as a baseline. Due to resource and time-execution constraints, in the validation stage we used a split of 80%-20% of the dataset. All the networks (ANN, DNN and Stacked-LSTM) consist of an input layer of 95 units and a sigmoid layer output. All the neurons in the hidden layers use ReLU activation functions. For the case of the ANN, different hidden layer widths were tried (from 512 to 2018, in 512 units steps) and values 0.001 and 0.005 were evaluated as Adam's learning rate. For the DNN, a variable number of hidden layers (up to three) were evaluated with optional dropout layers. Similarly, for the Stacked-LSTM, a variable number of LSTM layers (up to five) was tried. The width of each hidden / LSTM layer was set from 128 units up to 2048, in 128 units steps and the learning rate was varied from 0.0001 to 0.005. Finally, the *Wikify* flaw is not addressed in our study as it was in [7], given that on August 2021 its associated cleanup tag was revised in the template index and replaced for more detailed tags indicating more specifically which layout aspects must be corrected.

4.2 Results

The state-of-the-art F_1 score of 0.96 for the *Refimprove* flaw on the test set of the 1st *International Competition on Quality Flaw Prediction in Wikipedia* was achieved in [6] by using under-bagged decision trees with the min rule as aggregation method. Besides, it was also achieved by a Stacked-LSTM deep approach in [7]. As we can see in Table 2, only two models have surpassed this value; a new configuration of a Stacked-LSTM ($F_1 = 0.97$) and TabNet ($F_1 = 0.98$). It may seem small improving the state-of-the-art result by 1% and 2.1%, respectively; but it is worth considering than the benchmark is high and increasing by 2.1% the current F_1 score, reduces notably the gap to the optimum score. Moreover, our results are directly comparable to the values found in [4, 6, 7], since we have used the same data set and document model for representing the articles. In this respect, we also reached the benchmark of $F_1 = 0.99$ for the *Notability* flaw and remain 0.01 below from the benchmark of $F_1 = 0.99$ for the *No footnotes* (from [7]) and *Primary Sources* (from [4] and [7]) flaws.

As expected, the values reported on the test set correspond to the configurations which achieved the best values on the validations sets. Due to space con-

straints, we only report next the configurations of combinations which achieved the best values –highlighted in bold in Table 2– viz. ANN-HB (learning rate 0.005, 2048 neurons in the hidden layer), DNN-BO (learning rate 0.001, [1536, 2048, 2048][7]) and DNN-HB (learning rate 0.001, [1024, 512]) for *Notability* flaw, and Stacked-LSTM-HB (learning rate 0.001, [384, 512, 512, 256, 128]) for *Primary Sources* flaw. We evaluated TabNet with its default parameters.[8]

Table 2. F_1 values on the test set of the 1st International Competition on Quality Flaw Prediction in Wikipedia for all the evaluated models.

Flaws / Models	**ANN**			**DNN**			**Stacked LSTM**			**TabNet**
	RS	**BO**	**HB**	**RS**	**BO**	**HB**	**RS**	**BO**	**HB**	
No Footnotes	0.79	0.72	0.67	0.75	0.68	0.79	0.97	0.95	0.93	**0.98**
Notability	0.98	0.93	**0.99**	0.98	**0.99**	**0.99**	0.95	0.93	0.98	**0.99**
Primary Sources	0.86	0.76	0.86	0.92	0.89	0.87	0.76	0.93	**0.97**	**0.97**
Refimprove	0.64	0.64	0.63	0.63	0.63	0.64	0.97	0.93	0.81	**0.98**

5 Conclusions

In this work, we carried out a comparative study of three deep state-of-the-art approaches to automatically assess information quality; in particular, to identify four out of the ten quality flaws most frequent in Wikipedia, and the task was carried out by binary classification. The results obtained showed that the new benchmark of $F_1 = 0.98$ for the *Refimprove* flaw prediction was achieved by using the default configuration of TabNet architecture. Moreover, for the remaining flaws, very competitive results were obtained.

Acknowledgments

This work has been partially founded by PROICO 03-0620, UNSL, Argentina.

References

1. Anderka, M., Stein, B.: A breakdown of quality flaws in Wikipedia. In: 2nd joint WICOW/AIRWeb workshop on Web quality (WebQuality'12), ACM (2012) 11–18
2. Anderka, M., Stein, B., Lipka, N.: Detection of text quality flaws as a one-class classification problem. In: Proceedings of the CIKM'11, ACM (2011) 2313–2316
3. Anderka, M.: Analyzing and Predicting Quality Flaws in User-generated Content: The Case of Wikipedia. PhD thesis, Bauhaus-Universität Weimar (June 2013)

[7] This list notation should be understood as: 3 layers, 1536 neurons in the first layer, and 2048 in second and third layers.

[8] `https://pypi.org/project/pytorch-tabnet/`

4. Ferretti, E., Errecalde, M., Anderka, M., Stein, B.: On the use of reliable-negatives selection strategies in the pu learning approach for quality flaws prediction in wikipedia. In: 11th Intl. Workshop on Text-based Information Retrieval. (2014)
5. Ferretti, E., Cagnina, L., Paiz, V., Donne, S.D., Zacagnini, R., Errecalde, M.: Quality flaw prediction in spanish wikipedia: A case of study with verifiability flaws. Information Processing & Management **54**(6) (2018) 1169–1181
6. Bazán-Pereyra, G., Cuello, C., Capodici, G., Jofré, V., Ferretti, E., Errecalde, M.: Automatically assessing the need of additional citations for information quality verification in Wikipedia articles. In: Actas del XXV Congreso Argentino de Ciencias de la Computación (CACIC). (2019) 42–51 ISBN: 978-987-688-377-1.
7. Bazán Pereyra, G., Cuello, C., Capodici, G., Jofré, V., Ferretti, E., Bonnin, R., Errecalde, M.: Predicting information quality flaws in wikipedia by using classical and deep learning approaches. In Pesado, P., Arroyo, M., eds.: Computer Science – CACIC 2019, Cham, Springer International Publishing (2020) 3–18
8. Herrera, J., Funes, A., Ferretti, E., Cagnina, L.: Selección de Características para Clasificación de Clase Única de Fallas de calidad de Información en Wikipedia. In: Actas del XIII CoNaIISI. (2020)
9. Guda, B.P.R., Seelaboyina, S.B., Sarkar, S., Mukherjee, A.: NwQM: A neural quality assessment framework for Wikipedia. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, ACL (2020) 8396–8406
10. Wang, P., Li, M., Li, X., Zhou, H., Hou, J.: A hybrid approach to classifying wikipedia article quality flaws with feature fusion framework. Expert Systems with Applications **181** (2021)
11. Arik, S.Ö., Pfister, T.: TabNet: Attentive interpretable tabular learning. In: Proceedings of the AAAI Conference on Artificial Intelligence. Volume 35. (2021)
12. Anderka, M., Stein, B.: Overview of the 1st International Competition on Quality Flaw Prediction in Wikipedia. In Forner, P., Karlgren, J., Womser-Hacker, C., eds.: Working Notes Papers of the CLEF 2012 Evaluation Labs. (2012)
13. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In Pereira, F., Burges, C.J.C., Bottou, L., Weinberger, K.Q., eds.: Advances in Neural Information Processing Systems 25. (2012)
14. Bassani, E., Viviani, M.: Quality of Wikipedia articles: Analyzing features and building a ground truth for supervised classification. In: 11th International Joint Conference, IC3K. (2019) 338–346
15. Zhang, S., Hu, Z., Zhang, C., Yu, K.: History-based article quality assessment on Wikipedia. In: IEEE 5th Intl. Conference BigComp. (2018) 1–8
16. Schmidt, M., Zangerle, E.: Article quality classification on Wikipedia: introducing document embeddings and content features. In: 15th Intl. OpenSym. (2019)
17. Dang, Q.V., Ignat, C.L.: An end-to-end learning solution for assessing the quality of wikipedia articles. In: 13th Intl. Symposium on Open Collaboration. (2017) 1–10
18. Wang, P., Li, X.: Assessing the quality of information on Wikipedia: A deep-learning approach. Journal of the Association for Information Science and Technology **71**(1) (2020) 16–28
19. Rumelhart, D., Hinton, G., Williams, R.: Learning representations by back-propagating errors. Nature **323** (1986)
20. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press (2016)
21. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Computation **9**(8) (1997)
22. Graves, A., Mohamed, A., Hinton, G.E.: Speech recognition with deep recurrent neural networks. CoRR **abs/1303.5778** (2013)