Detection of Text Quality Flaws as a One-class Classification Problem

Maik Anderka Benno Stein Nedim Lipka

Bauhaus-Universität Weimar
99421 Weimar, Germany
<first name>.<last name>@uni-weimar.de

ABSTRACT

For Web applications that are based on user generated content the detection of text quality flaws is a key concern. Our research contributes to *automatic quality flaw detection*. In particular, we propose to cast the detection of text quality flaws as a one-class classification problem: we are given only positive examples (= texts containing a particular quality flaw) and decide whether or not an unseen text suffers from this flaw. We argue that common binary or multiclass classification approaches are ineffective in here, and we underpin our approach by a real-world application: we employ a dedicated one-class learning approach to determine whether a given Wikipedia article suffers from certain quality flaws. Since in the Wikipedia setting the acquisition of sensible test data is quite intricate, we analyze the effects of a biased sample selection. In addition, we illustrate the classifier effectiveness as a function of the flaw distribution in order to cope with the unknown (real-world) flaw-specific class imbalances. Altogether, provided test data with little noise, four from ten important quality flaws in Wikipedia can be detected with a precision close to 1.

Categories and Subject Descriptors: H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval; H.5.3 [Information Interfaces and Presentation]: Group and Organization Interfaces— *Evaluation/methodology*
General Terms: Measurement, Algorithms, Experimentation

1. INTRODUCTION

The machine-based assessment of text quality is becoming a topic of enormous interest. This fact is rooted, among others, in the increasing popularity of user generated Web content [3] and the (unavoidable) divergence of the delivered content's quality. Most of the relevant literature on automatic text quality assessment deals with the classification of texts in predefined abstract quality schemes, see for instance [2, 6, 11]. Only a few approaches focus on quality flaw *detection* and try to give precise indications in which respects a text needs improvement [1, 7]. A general finding of our literature review (from which we can show only a tiny excerpt here) is the fact that the detection of text quality flaws in general has not yet been operationalized.

CIKM'11, October 24–28, 2011, Glasgow, Scotland, UK.

The paper in hand focuses on detection issues, whereas our contributions are as follows: Firstly, in the remainder of this section, we argue that the detection of text quality flaws is essentially a one-class classification problem and give a respective problem definition. Secondly, we employ a one-class machine learning approach to detect quality flaws in Wikipedia articles (Section 2). Thirdly, we perform comprehensive analyses to assess the effectiveness of our approach (Section 3).

Problem Definition Let D be a set of text documents and let F be a set of text quality flaws. A document $d \in D$ can contain up to $|F|$ flaws, where, without loss of generality, the flaws in F are considered as being uncorrelated. A classifier c hence has to solve the following multi-labeling problem:[1]

$$c : \mathbf{D} \rightarrow 2^F,$$

where 2^F denotes the power set of F. A document d is represented by a feature vector $\mathbf{d}$, called document model, where $\mathbf{D}$ denotes the set of document models for D.

Basically, there are two strategies to tackle multi-labeling problems: (1) by multiclass classification, where a single classifier is learned on the power set of all classes, and (2) by multiple binary classification, where a specific classifier $c_i : \mathbf{D} \rightarrow \{1, 0\}$ is learned for each class $f_i \in F$. Since the high number of classes under a multiclass classification strategy entails a very large number of training examples, the second strategy is favorable.

In most classification problems training data is available for all classes that can occur at prediction time, and hence it is appropriate to train a classifier c_i with (positive) examples of the target class f_i and (negative) examples from the classes $F \setminus f_i$. However, in the case of detecting text quality flaws an unseen document can either belong to the target class f_i or to some unknown class that was not available during training. I.e., the standard discrimination-based classification approaches (binary or multiclass) are not applicable to learn a class-separating decision boundary: given a flaw f_i, its target class is formed by those documents that contain (among others) flaw f_i—but it is impossible to model the "co-class" with documents *not* containing f_i. Even if many counterexamples were available, they could not be exploited to properly characterize the universe of possible counterexamples. As a consequence, we model the classification $c_i(\mathbf{d})$ of an document $d \in D$ with respect to a text quality flaw f_i as the following one-class classification problem: Decide whether or not d contains f_i, whereas a sample of documents containing f_i is given.

The following example may serve as an additional illustration: Wikipedia articles should be written in a formal tone[2], and hence

[1] Possibly existing correlations among the flaws in F will not affect the nature of the multi-labeling problem.
[2] http://en.wikipedia.org/wiki/Wikipedia:Tone#Tone

"inappropriate tone" is a text quality flaw in this particular context. An even large sample of articles that suffer from this flaw can be compiled without problems (consider articles containing slang, jargon, etc.). However, it is impossible to compile a representative sample of articles that are written in a formal tone. Though there definitely exist outstanding articles written in a formal tone, they cannot be considered as a representative sample.

For an in-depth discussion of one-class classification and a survey of respective methodologies see [14, 9]. Typical one-class problems in the information retrieval domain include typist recognition [8], authorship verification [10], plagiarism analysis [12], and anomaly detection [5].

2. QUALITY FLAWS IN WIKIPEDIA

In previous research we analyzed cleanup template messages in the English Wikipedia and compiled a set of quality flaws of Wikipedia articles that have been tagged by the community [1]. This analysis is restricted to a specific subset of cleanup template messages. By applying the same approach without these restrictions to a more recent Wikipedia snapshot from January 2011 we extracted 388 quality flaws, which form the set F; the 3 557 468 articles of the snapshot form the set D. We distinguish different subsets of D, see Figure 1: The set D^- comprises 979 299 articles that have been tagged with at least one of the flaws from F, which corresponds to 27.5% of D. Notice that we have no knowledge about the articles in $D \setminus D_i^-$; these articles either do not contain the flaw f_i or have not yet been evaluated with respect to f_i.[3]

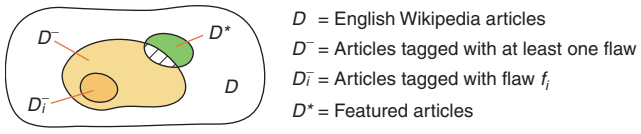

Figure 1: The sets of Wikipedia articles which we distinguish in this paper. Without loss of generality we assume in our experiments that the hashed area $D^- \cap D^*$ is empty, i.e., featured articles are flawless.

We make two assumptions in order to estimate the actual frequency of a flaw f_i: (1) each article in D^- is tagged completely, i.e. with all flaws that it contains (Closed World Assumption), and (2) the distribution of f_i in D^- is identical to the distribution of f_i in D. Based on these assumptions we estimate the actual frequency of a flaw f_i by the ratio of articles in D_i^- and articles in D^-.

We model the quality flaws of an article $d \in D$ by a feature vector $\mathbf{d}$, where each dimension in $\mathbf{d}$ quantifies a quality-specific characteristics of d. Our document model employs state-of-the-art features that have been proposed in the relevant literature [6, 11, 13] as well as new quality flaw predictors that quantify the usage of in-links, templates, lists, and special words, among others.

We employ a one-class classification approach as proposed by [8], which combines density estimation with class probability estimation. The idea is to use a reference distribution to model the probability $P(\mathbf{d} \mid f_i')$ of an artificial class f_i', and to generate (artificial) data governed by the distribution characteristic of f_i'. For a flaw f_i let $P(f_i)$ and $P(f_i \mid \mathbf{d})$ be the a-priori probability and the class probability function respectively. According to Bayes' theorem the class-conditional probability for f_i is given as follows:

$$P(\mathbf{d} \mid f_i) = \frac{(1 - P(f_i)) \cdot P(f_i \mid \mathbf{d})}{P(f_i) \cdot (1 - P(f_i \mid \mathbf{d}))} \cdot P(\mathbf{d} \mid f_i')$$

$P(f_i \mid \mathbf{d})$ is estimated by a class probability estimator (a classifier whose output is interpreted as probability). Since we are in a one-class situation we have to rely on the face value of $P(\mathbf{d} \mid f_i)$; more specifically, $P(\mathbf{d} \mid f_i)$ cannot be used to determine a maximum a-posterior (MAP) hypothesis among the $f_i \in F$. As a consequence, given $P(\mathbf{d} \mid f_i) < \tau$ with $\tau = 0.5$, the hypothesis that d suffers from f_i could be rejected. However, because of the approximative nature of $P(f_i \mid \mathbf{d})$ and $P(f_i)$ the estimation for $P(\mathbf{d} \mid f_i)$ is not a true probability, and the threshold τ has to be chosen empirically. In practice, the threshold τ is derived from a user-defined target rejection rate, trr, which is the rejection rate of the target class training data.

[3] A special case is the set D^*, which will be discussed later on.

3. ANALYSIS

We report on experiments to assess the effectiveness of our classification approach in detecting ten of the most frequent quality flaws of Wikipedia articles. The evaluation treats the following issues:

1. Since a bias may not be ruled out when collecting outlier examples for a classifier's test set, we investigate the consequences of the two extreme (overly optimistic, overly pessimistic) settings (Section 3.1).
2. Since users (Wikipedia editors) have different expectations regarding the classification effectiveness given different flaws, we analyze the optimal operating point for each flaw-specific classifier within the controlled setting of a balanced class distribution (Section 3.2).
3. Since the true flaw-specific class imbalances in Wikipedia can only be hypothesized, we illustrate the effectiveness of the classifiers in different settings, this way enabling users (Wikipedia editors) to assume an optimistic or a pessimistic position (Section 3.3).

3.1 Outlier Selection

Recall that no articles are available that have been tagged to *not* contain a quality flaw $f_i \in F$. Thus a classifier c_i can be evaluated only with respect to its recall, whereas a recall of 1 can be achieved easily by classifying all examples into the target class of f_i. In order to evaluate c_i with respect to its precision one needs a representative sample of examples from outside the target class, so-called outliers. As motivated above, in a one-class situation it is not possible to compile a representative sample, and one way out of the dilemma is the generation of uniformly distributed outlier examples [14]. Here, we pursue two strategies to derive examples from outside the target class, which result in the following settings:

1. *Optimistic Setting.* Use of featured articles as outliers. This approach is based on the hypothesis that featured articles do not contain a quality flaw at all, see Figure 1.[4] Under this setting one introduces some bias since featured articles cannot be considered as a representative sample of Wikipedia articles.
2. *Pessimistic Setting.* Use of a random sample from $D \setminus D_i^-$ as outliers for each f_i. This approach may introduce considerable noise since the set $D \setminus D_i^-$ is expected to contain untagged articles that suffer from f_i.

The above settings address two extremes: classification under laboratory conditions (overly optimistic) versus classification in the wild (overly pessimistic). The experiment design is owing to the

[4] The hypothesis may hold in many cases but not always: the snapshot comprises 13 featured articles that have been tagged with some flaw. We discarded these articles in our experiments.

facts that "no-flaw features" cannot be stated and that the number of false positives as well as the number of false negatives in the set D^- of tagged articles are unknown.

3.2 Effectiveness of Flaw Detection

We use the English Wikipedia snapshot from January 15, 2011.[5] The articles' plain texts and wikitexts are extracted in a preprocessing step by processing the "pages-articles" XML dump on an Apache Hadoop cluster using Google's MapReduce. Furthermore, a local copy of the Wikipedia database is established by importing the database dumps into a MySQL database. The plain texts, the wikitexts, and the local Wikipedia database form the basis to compute the features of our document model.

Experiment Design The evaluation is performed for the set $F' \subset F$ of the ten most frequent quality flaws that show the following three properties: they describe a single and specific quality aspect, they refer to an article as a whole, and they are not restricted to a particular domain, language, or user group (see Table 1). About 70% of the articles in D^- suffer from these flaws. In the optimistic setting 1 000 outliers are randomly selected from the 3 128 featured articles in the snapshot. In the pessimistic setting 1 000 outliers are randomly selected for each flaw $f_i \in F'$ from $D \setminus D_i^-$. We evaluate our approach under both settings by applying the following procedure: For each flaw $f_i \in F'$ the one-class classifier c_i is evaluated with 1 000 articles which are randomly sampled from D_i^- and the respective 1 000 outliers, applying tenfold cross-validation. Within each run the classifier is trained with 900 articles from D_i^-, whereas testing is performed with the remaining 100 articles from D_i^- plus 100 outliers. Note that c_i is trained exclusively with the examples of the respective target class, i.e., the articles in D_i^-: The training of c_i is neither affected by the class distribution nor by the outlier selection strategy that is used in the respective setting.

The one-class classifier is built as follows: a class with artificial examples is generated, whereas the feature values obey a Gaussian distribution with $\mu = 0$ and $\sigma^2 = 1$. The Gaussian distribution is employed in favor of a more complex reference distribution to underline the robustness of the approach. The proportion of the generated data is 0.5 compared to the target class. As class probability estimators we apply bagged random forest classifiers with 1 000 decision trees and ten bagging iterations. A random forest is a collection of decision trees that differ with respect to their features, and a voting over all trees is run in order to obtain a classification decision (for further details see [4]).

Operating Point Analysis For the major part of the relevant use cases precision is the determining measure of effectiveness; consider for instance a bot that autonomously tags flawed articles. The precision of the one-class classifier is controlled by the hyperparameter target rejection rate. We empirically determine the optimal operating point for each of the ten flaws under both the optimistic and the pessimistic setting. Here, the optimal operating point corresponds to the target rejection rate of the maximum precision classifier. Figure 2 illustrates the operating point analyses exemplary for the flaw *Unreferenced*: with increasing target rejection rate the recall value drops while the precision values increase. Observe that the recall is the same in both settings, since it solely depends on the target class training data. For the flaw *Unreferenced* the optimal operating points under the optimistic and the pessimistic setting are at a target rejection rate of 0.1 and 0.35 respectively (with precision values of 0.99 and 0.63).

The precision of a one-class classifier cannot be adjusted arbitrarily since the target rejection rate controls only the probability

[5]Wikimedia downloads: http://download.wikimedia.org/enwiki.

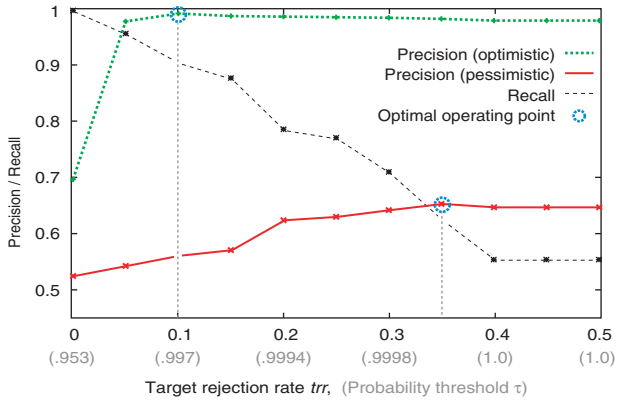

Figure 2: Precision and recall over target rejection rate for the flaw *Unreferenced*. The figure illustrates the difference in terms of precision under the optimistic setting, using featured articles as outliers, and the pessimistic setting, using random articles as outliers. The recall is the same under both settings. The optimal operating points correspond to the target rejection rates that maximizes classifier precision.

threshold τ for the classification decision. For instance, a target rejection rate of 0.1 means that a τ is chosen such that 10% of the target class training data will be rejected, which results in a classifier that performs with an almost stable recall of 0.9. Increasing the target rejection rate entails an increase of τ. However, if τ achieves its maximum no further examples can be rejected, and hence both the precision and the recall remain constant beyond a certain target rejection rate (which is 0.4 for the flaw *Unreferenced*, see Figure 2).

Results and Discussion Table 1 shows the performance values for the ten quality flaws. The values correspond to the performances at their optimal operating points; the performance is quantified as precision (*prec*) and recall (*rec*). We also report the area under ROC curves (AUC), which is important to assess the tradeoff between specificity and sensitivity of a classifier: an AUC value of 0.5 means that all specificity-sensitivity-combinations are equivalent, which in turn means that the classifier is random guessing.

Under the optimistic setting four flaws can be detected with a nearly perfect precision. For the flaw *Notability* even the achieved recall value is very high, which means that this flaw can be detected exceptionally well. As expected, the effectiveness of the one-class classifiers deteriorates under the pessimistic setting. However, the classifiers still achieve reasonable precision values, and even in the noisy test set the flaw *Orphan* can be detected with a good precision. Notice, however, that the expected performance in the wild lies in between the two extremes. For some flaws the effectiveness of the one-class classifiers is pretty low under both settings, including *Original research*. We explain this behavior as follows: (1) Either the document model is inadequate to capture certain flaw characteristics, or (2) the hypothesis class of the one-class classification approach is too simple to capture the flaw distributions.

3.3 Flaw-specific Class Imbalances

The performance values in Table 1 presume a balanced class distribution, i.e., the one-class classifiers are evaluated with the same number of flawed articles and outliers. The real distribution of flaws in Wikipedia is unknown (cf. Section 2), and we hence report precision values as a function of the class imbalance. Given the recall and the false positive rate (*fpr*) of a classifier for the balanced setting, its precision for a class size ratio of 1:n (flawed articles : flawless articles) computes as follows:

$$prec = \frac{rec}{rec + n \cdot fpr}$$

Table 1: Individual performance for each of ten quality flaws at the optimal operating point, using featured articles as outliers (optimistic setting) and using random articles as outliers (pessimistic setting). The class distribution is balanced under both settings. The flaw ratio 1:n (flawed articles : flawless articles) corresponds to the estimated actual frequency of a flaw.

	Flaw name	Flaw ratio	Optimistic setting			Pessimistic setting		
			prec	*rec*	AUC	*prec*	*rec*	AUC
f_1	Unreferenced	1:3	0.99	0.90	0.95	0.63	0.63	0.63
f_2	Orphan	1:5	1.00	0.90	0.95	0.72	0.59	0.68
f_3	Refimprove	1:10	0.83	0.87	0.85	0.57	0.56	0.57
f_4	Empty section	1:21	0.90	0.70	0.82	0.74	0.70	0.72
f_5	Notability	1:26	0.99	0.96	0.98	0.66	0.61	0.65
f_6	No footnotes	1:36	0.82	0.87	0.84	0.59	0.59	0.58
f_7	Primary sources	1:44	0.94	0.90	0.92	0.61	0.59	0.61
f_8	Wikify	1:68	0.96	0.87	0.92	0.64	0.58	0.63
f_9	Advert	1:136	0.86	0.91	0.88	0.65	0.58	0.63
f_{10}	Original research	1:147	0.76	0.64	0.71	0.56	0.80	0.59

The false positive rate is the ratio between the detected negative examples and all negative examples, and hence it is independent from the class size ratio; the same argument applies to the recall. Figure 3 shows the precision values as a function of the flaw distribution under the optimistic setting.

Observe that the expected precision values for the flaws *Unreferenced*, *Orphan*, and *Notability* are still high. The flaw ratio of the flaw *Unreferenced* is 1:3, and thus the expected precision is close to that of the 1:1 ratio. The flaw *Orphan* can be detected with a precision of 1, i.e., the false positive rate is 0, and hence the detection performance is independent of the class imbalance. Although the flaw ratio of the flaw *Notability* is 1:26, the expected precision is still about 0.9, which shows that the respective one-class classifier captures the characteristics of the flaw exceptionally well. The expected precision values for those flaws with a flaw ratio 1:n where $n > 40$ are lower than 0.2. Aside from conceptual weaknesses

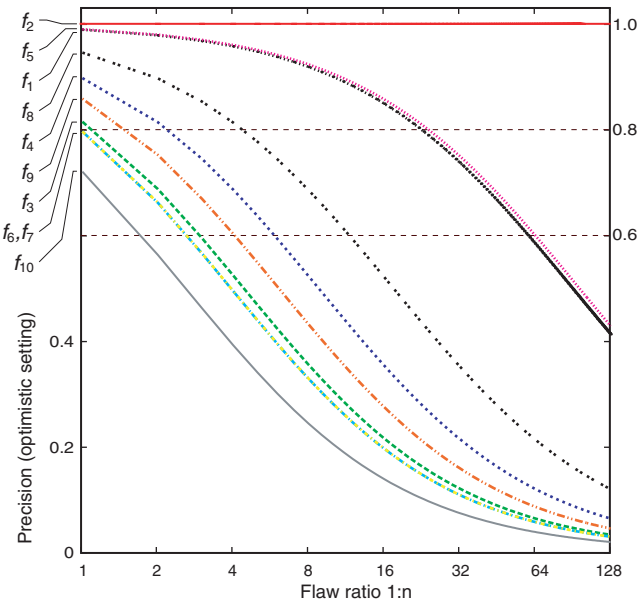

Figure 3: Precision in the optimistic setting over flaw ratio for ten quality flaws: 1:n $\sim$ with flaw : without flaw, with n $\in [1; 128]$. The figure puts the classification performances reported in Table 1 into perspective, since it considers imbalances in the test sets that might occur in the wild.

regarding the employed document model, the weak performance indicates also that the training set of the respective one-class classifiers might be too small.

4. CONCLUSIONS AND OUTLOOK

We treat the detection of text quality flaws as a process where for each known flaw an expert is asked whether or not a given document suffers from it; the experts in turn are operationalized by one-class classifiers. This approach is applied to detect text quality flaws in the English Wikipedia. Our evaluation is based on a corpus comprising 10 000 human-labeled Wikipedia articles. We report on precision values close to 1 for four out of ten important quality flaws—presuming an optimistic test set with little noise and a balanced flaw distribution. Even for a class size ratio of 1:16 three flaws can still be detected with a precision of about 0.9.

We are convinced that the presented or similar approaches will help to simplify Wikipedia's quality assurance process by spotting weaknesses within articles. Our current research on quality flaw detection in Wikipedia targets the investigation of tailored one-class classifiers for each flaw, as well as the development of flaw-specific document models that combine expert rules, multi-level filtering, and feature selection.

5. REFERENCES

[1] M. Anderka, B. Stein, and N. Lipka. Towards automatic quality assurance in Wikipedia. In *Proceedings of WWW'11*, pages 5–6, 2011. ACM.

[2] E. Agichtein, C. Castillo, D. Donato, A. Gionis, and G. Mishne. Finding high-quality content in social media. In *Proceedings of WSDM'08*, pages 183–194, 2008. ACM.

[3] R. Baeza-Yates. User generated content: how good is it? In *Proceedings of WICOW'09*, pages 1–2, 2009. ACM.

[4] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[5] V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection: a survey. *ACM Computing Surveys*, 41(3):15:1–15:58, 2009.

[6] D. Dalip, M. Gonçalves, M. Cristo, and P. Calado. Automatic quality assessment of content created collaboratively by Web communities: a case study of Wikipedia. In *Proceedings of JCDL'09*, pages 295–304, 2009. ACM.

[7] L. Gaio, M. den Besten, A. Rossi, and J. Dalle. Wikibugs: using template messages in open content collections. In *Proceedings of WikiSym'09*, pages 1–7, 2009. ACM.

[8] K. Hempstalk, E. Frank, and I. Witten. One-class classification by combining density and class probability estimation. In *Proceedings of ECML/PKDD'08*, pages 505–519, 2008. Springer.

[9] V. Hodge and J. Austin. A survey of outlier detection methodologies. *Artificial Intelligence Review*, 22(2):85–126, 2004.

[10] M. Koppel and J. Schler. Authorship verification as a one-class classification problem. In *Proceedings of ICML'04*, pages 1–7, 2004. ACM.

[11] N. Lipka and B. Stein. Identifying featured articles in Wikipedia: writing style matters. In *Proceedings of WWW'10*, pages 1147–1148, 2010. ACM.

[12] B. Stein, N. Lipka, and P. Prettenhofer. Intrinsic plagiarism analysis. *Language Resources and Evaluation*, 45(1):63–82, 2011. Springer.

[13] B. Stvilia, M. Twidale, L. Smith, and L. Gasser. Assessing information quality of a community-based encyclopedia. In *Proceedings of ICIQ'05*, pages 442–454, 2005. MIT

[14] D. Tax. *One-Class Classification*. PhD thesis, Delft University of Technology, 2001.