Predicting Quality Flaws in User-generated Content: The Case of Wikipedia

Maik Anderka
Bauhaus-Universität Weimar
99421 Weimar, Germany
maik.anderka@gmail.com

Benno Stein
Bauhaus-Universität Weimar
99421 Weimar, Germany
benno.stein@uni-weimar.de

Nedim Lipka
Bauhaus-Universität Weimar
99421 Weimar, Germany
lipka.nedim@gmail.com

ABSTRACT

The detection and improvement of low-quality information is a key concern in Web applications that are based on user-generated content; a popular example is the online encyclopedia Wikipedia. Existing research on quality assessment of user-generated content deals with the classification as to whether the content is high-quality or low-quality. This paper goes one step further: it targets the *prediction of quality flaws*, this way providing specific indications in which respects low-quality content needs improvement. The prediction is based on user-defined cleanup tags, which are commonly used in many Web applications to tag content that has some shortcomings. We apply this approach to the English Wikipedia, which is the largest and most popular user-generated knowledge source on the Web. We present an automatic mining approach to identify the existing cleanup tags, which provides us with a training corpus of labeled Wikipedia articles. We argue that common binary or multiclass classification approaches are ineffective for the prediction of quality flaws and hence cast quality flaw prediction as a *one-class classification problem*. We develop a quality flaw model and employ a dedicated machine learning approach to predict Wikipedia's most important quality flaws. Since in the Wikipedia setting the acquisition of significant test data is intricate, we analyze the effects of a biased sample selection. In this regard 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. The flaw prediction performance is evaluated with 10 000 Wikipedia articles that have been tagged with the ten most frequent quality flaws: provided test data with little noise, four flaws 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*

Keywords

User-generated Content Analysis, Information Quality, Wikipedia, Quality Flaw Prediction, One-class Classification

SIGIR'12, August 12–16, 2012, Portland, Oregon, USA.

1. INTRODUCTION

Web applications that are based on user-generated content come under criticism for containing low-quality information. This applies also to Wikipedia, the largest and most popular user-generated knowledge source on the Web; Wikipedia contains articles from more than 280 languages, the English version contains more than 3.9 million articles, and wikipedia.org ranks among the top ten most visited Web sites. The community of Wikipedia authors is heterogeneous, including people with different levels of education, age, culture, language skills, and expertise. In contrast to printed encyclopedias, the contributions to Wikipedia are not reviewed by experts before publication. These factors make clear that the most important, but probably the most difficult challenge for Wikipedia pertains to the quality of its articles. Wikipedia founder Jimmy Wales announced in a recent interview: "Our goal is to make Wikipedia as high-quality as possible. [Encyclopædia] Britannica or better quality is the goal." [36] However, the size and the dynamic nature of Wikipedia render a comprehensive manual quality assurance infeasible. This is underlined by the fact that only a small number of articles are labeled as *featured*, i.e., are considered as well-written, comprehensive, well-researched, neutral, and stable.[1]

The existing research on automatic quality assessment of Wikipedia articles targets the classification task "Is an article featured or not?" Although the developed approaches perform nearly perfect in distinguishing featured articles from non-featured ones, they provide virtually no support for quality assurance activities. The classification is based on meta features that correlate with featured articles in general, but cannot (and were not intended to) provide a rationale in which respects an article violates Wikipedia's featured article criteria. This goal, however, is addressed in this paper where we try to predict the flaws of an article that need to be fixed to improve its quality. We exploit the fact that Wikipedia users who encounter some flaw (but who are either not willing or who don't have the knowledge to fix it) can tag the article with a so-called cleanup tag. The existing cleanup tags give us the set of quality flaws that have been identified so far by Wikipedia users. The tagged articles are used as a source of human-labeled data that is exploited by a machine learning approach to predict flaws of untagged articles.

1.1 Research Questions and Contributions

Our contributions relate to the fields of user-generated content analyses, data mining, and machine learning and focus on the following research questions:

What cleanup tags exist in Wikipedia? Cleanup tags are realized by templates, which are special Wikipedia pages that can be included into other pages. The identification of templates that define

[1]At the time of this writing, less then 0.1% of the English Wikipedia articles are tagged with the "featured" label.

cleanup tags is a non-trivial task since there is no dedicated qualifier for cleanup tags and Wikipedia contains nearly 320 000 different templates. We hence implement an automated mining approach to extract the existing cleanup tags from Wikipedia. Altogether 388 cleanup tags are identified.

How to model quality flaws? A large body of features—allegedly predicting article quality—has been proposed in previous work on automatic quality assessment in Wikipedia. We have compiled a comprehensive breakdown, implement more than 100 features from previous work, and introduce 13 new features that directly target particular quality flaws. Moreover, we distinguish two modeling paradigms: intensional and extensional. The former allows for an efficient and precise prediction of certain flaws based on rules, the latter resorts to the realm of machine learning.

How to predict quality flaws? To the best of our knowledge, an algorithmic prediction of quality flaws in Wikipedia has not been operationalized before. We suggest to cast quality flaw prediction in Wikipedia as a one-class problem: Given a sample of articles that have been tagged with flaw f, decide whether or not an article suffers from f. We adapt a dedicated one-class classification machine learning approach to tackle this problem.

How to assess classifier effectiveness? A representative sample of Wikipedia articles that have been tagged to *not* contain a particular quality flaw is not available. To assess the effects of a biased sample selection our analysis is performed on both an optimistic test set, using featured articles as outliers, as well as a pessimistic test set, using random untagged articles as outliers. Given the optimistic test set and a balanced class distribution, four flaws can be detected with a precision close to 1. Since the true flaw-specific class imbalances in Wikipedia are unknown, we illustrate the classifiers' precision values as functions of the class size ratio.

The paper is organized as follows. Section 2 discusses related work on quality assessment in Wikipedia. Section 3 describes our cleanup tag mining approach. Section 4 presents the quality flaw model. Section 5 gives a formal problem definition and describes the employed one-class classification approach. Section 6 presents the evaluation and discusses the results. Finally Section 7 concludes this paper and gives an outlook on future work.

2. RELATED WORK

From its debut in 2001 till this day Wikipedia is subject of ongoing research in different academic disciplines.[2] This section surveys the research related to information quality, whereas the focus is on automatic quality assessment. We start with a discussion of the general concept of information quality.

Information quality is a multi-dimensional concept and combines criteria such as accuracy, reliability, and relevance. A good deal of the existing research focuses on the mapping between criteria sets and classification schemes [20], for which Madnick et al. [24] give a comprehensive overview. A widely accepted interpretation of information quality is the "fitness for use in a practical application" [33]: the assessment of information quality requires the consideration of context and use case. In Wikipedia the context is well-defined, namely by the encyclopedic genre. It forms the ground for Wikipedia's information quality ideal, which has been formalized—better: made communicable and quantifiable—within the so-called featured article criteria[3]. That the relation between information quality and organizational outcome can be measured is reported by Slone [28]. It stands also to reason that incorporating information quality metrics into information retrieval approaches can significantly improve the search effectiveness of Web search environments [7, 26, 38, 39].

The machine-based assessment of information quality is becoming a topic of enormous interest. This fact is rooted, among others, in the increasing popularity of user-generated Web content [6] and the (unavoidable) divergence of the delivered content's quality. Most of the prior research on automatic quality assessment deals with the identification of high-quality content, see for instance [2]. The relevant literature mentions a variety of approaches to automatically assess quality in Wikipedia. These approaches differ in their document model, i.e., the feature number, the feature complexity, and the rationale to quantify the quality of an article. We have compiled a comprehensive overview of the proposed article features, organized along the four dimensions content, structure, network, and edit history; see Table 3 in Appendix A.

Lih [21] models quality by the number of edits and the number of unique editors; the higher these values are the higher shall be an article's quality. However, an analysis whether the proposed metrics correlate with high-quality articles is missing. Stvilia et al. [29] use exploratory factor analysis to group 19 article features into 7 quality metrics. By classifying 236 featured articles and 834 random articles under these metrics they achieve an F-measure of 0.91 for the featured set and 0.975 for the random set. Hu et al. [19] model the quality of an article via the mutual dependency between article quality and author authority. They perform several quality ranking experiments and show that their model is superior to a baseline model that relies on word counts. Wilkinson and Huberman [34] show that featured articles can be distinguished from non-featured articles by the number of edits and distinct editors. They also find that featured articles are characterized by a higher degree of cooperation, which is quantified by the number of revisions of the particular Wikipedia discussion pages. Blumenstock [8] shows that still a single word count feature can compete with sophisticated features when classifying 1 554 featured articles and 9 513 random articles. Dalip et al. [13] classify the articles along six abstract quality schemes. Their comparison of different feature sets shows that textual features perform best. Lipka and Stein [22] employ character trigrams, originally applied for writing style analysis, to classify a balanced set of featured and non-featured articles. They achieve an F-measure value of 0.964 for featured articles. They also show that character trigrams are superior to part of speech trigrams, word counts, and bag of words models.

Although the mentioned approaches differ in their robustness and complexity, they perform nearly perfect in distinguishing featured articles from non-featured ones or, stated generally, high-quality articles from low-quality articles. As already motivated, the practical support for Wikipedia's quality assurance process is marginal. A first step towards automatic quality assurance in Wikipedia is *the detection* of quality flaws, which we proposed in previous research [3, 4]. Here, we push this idea further and extend our previous work with (1) a comprehensive breakdown of prior work on quality assessment, (2) an in-depth discussion of the cleanup tag mining approach, (3) a description of the quality flaw model, and (4) a detailed analysis of the one-class problem. The cleanup tag mining approach has also been used in [5], were we analyzed the incidence and the extent of quality flaws in the English Wikipedia.

There is also notable research that relates indirectly to quality in Wikipedia: trust and reliability of articles [12, 37], accuracy and formality [14, 16], author reputation [1, 32], and automatic vandalism detection [27].

[2]Academic studies of Wikipedia: `http://en.wikipedia.org/wiki/Wikipedia:Wikipedia_in_academic_studies`.

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

3. MINING CLEANUP TAGS

Cleanup tags provide a means to tag flaws in Wikipedia articles. As shown in Figure 1, cleanup tags are used to inform readers and editors of specific problems with articles, sections, or certain text fragments. However, there is no silver bullet to compile a complete set of all cleanup tags. Cleanup tags are realized with templates, whereas templates in turn are special Wikipedia pages that can be included in other pages. Although templates can be separated from other pages by their namespace (the prefix "Template:" in the page title), there is no dedicated qualifier to separate templates that are used to implement cleanup tags from other templates. A complete manual inspection is infeasible as Wikipedia contains about 320 000 different templates. To cope with this situation we developed an extraction approach that exploits several sources within Wikipedia containing meta information about cleanup tags (Section 3.2). In this regard we also analyzed the frequency and nature of the extracted tags (Section 3.3). At first, we describe the data underlying our mining approach (Section 3.1).

3.1 Data Base and Preprocessing

To ensure reproducibility, the analyses in this paper are based on a snapshot instead of investigating Wikipedia up-to-the-minute. Wikipedia snapshots are provided by the Wikimedia Foundation in monthly intervals. Because of its size and popularity we consider the English language edition most appropriate, and we use the English Wikipedia snapshot from January 15, 2011.[4] A Wikimedia snapshot comprises a complete copy of Wikipedia. The wikitext sources of all pages (and of all revisions) are available as a single XML file that is enriched by some meta information. In addition, several tables of the Wikipedia database are available in the form of SQL dumps, totaling about 40GB. In a preprocessing step, we create a local copy of the Wikipedia database by importing the SQL dumps into a MySQL database. Since we do not target a content analysis, a processing of the XML dumps is not necessary. The local copy of the Wikipedia database allows for efficient parsing and mining without causing traffic on the Wikimedia servers. Note that all of our analyses can be performed on the original Wikipedia database as well.

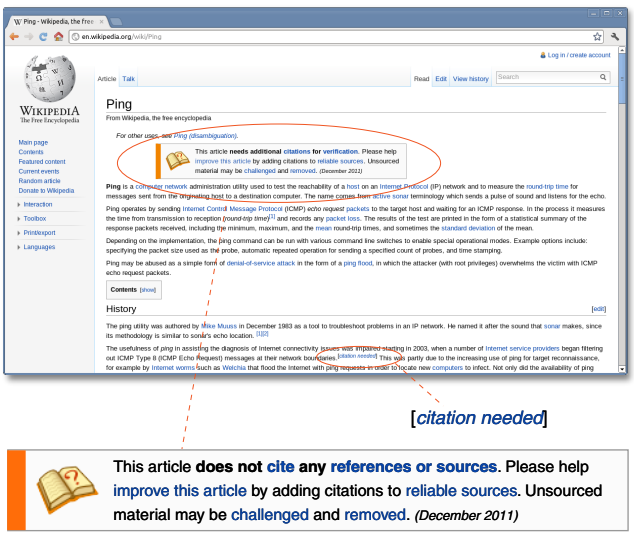

Figure 1: The Wikipedia article "Ping" with two cleanup tags. The tag box *Unreferenced* refers to the whole article while the inline tag *Citation needed* refers to a particular claim.

[4]Wikimedia downloads: `http://download.wikimedia.org`.

3.2 Method

We employ a two-step approach to compile the set of cleanup tags: (1) an initial set of cleanup tags is extracted from two meta sources within Wikipedia, and (2) the initial set is further refined by applying several filtering substeps.

Step 1: Extraction The first meta source that we employ is the Wikipedia administration category *Category:Cleanup_templates*, which comprises templates that are used for tagging articles as requiring cleanup. The category also has several subcategories to further organize the cleanup tags by their usage, e.g., inline cleanup templates or cleanup templates for WikiProjects. The page titles of those templates linking to the category or some subcategory are obtained from the local Wikipedia database, which results in 272 different cleanup tags. The second source is the Wikipedia meta page *Wikipedia:Template_messages/Cleanup*, which comprises a manually maintained listing of cleanup templates that may be used to tag articles as needing cleanup. From a technical point of view, the page is a composition of several pages (transclusion principle). For each of these pages, the content of the revision from the snapshot time is retrieved using the MediaWiki API.[5] A total of 283 different cleanup tags are extracted from the wikitexts of the retrieved pages using regular expressions. Merging the findings from both sources gives 457 different cleanup tags.

Step 2: Refinement A cleanup tag may have several alternative titles linking to it through redirects. For example, the tag *Unreferenced* has the redirects *Unref*, *Noreferences*, and *No refs* among others. We resolve all redirects using the local Wikipedia database. Moreover, we discard particular subtemplates from the initial set of cleanup tags. Subtemplates are identified by a suffix in the page title, and they are used for testing purposes (suffixes "/sandbox" and "/testcases") or provide a template description (suffix "/doc"). We also discard meta-templates, i.e., templates that are solely used either as building blocks inside other templates or to instantiate other templates with a particular parameterization. Meta-templates are derived from the Wikipedia categories *Wikipedia_metatemplates* and *Wikipedia_substituted_templates*. Altogether we collect a set of 388 cleanup tags.

Discussion To evaluate our mining approach, we manually inspected all documentation pages of the 388 cleanup tags. They give information about purpose, usage, and scope of a template, and our analysis reveals that all tags are indeed related to a particular cleanup task. I.e., each of the 388 cleanup tags defines a particular quality flaw. Our mining approach does not guarantee completeness though, since the true set of cleanup tags is unknown in general. However, from a quantitative point of view we are confident that we identify the most common cleanup tags, and hence the most important quality flaws.

3.3 Analysis

The Wikipedia snapshot comprises 3 557 468 articles, from which 979 299 (27.53%) are tagged with at least one quality flaw. Some articles are tagged with multiple flaws (multi-labeling). The number of flaws per article ranges from 1 to 17. However, the majority (74.95%) of the tagged articles are tagged with exactly one flaw. The number of tagged articles is an underestimation of the true frequencies of the quality flaws, since due to the size and the few control mechanisms in Wikipedia it is more than likely that many flawed articles are still not identified. From the 388 cleanup tags 18 merely state that some cleanup is required at all but do not

[5]We use the API in favor of parsing the XML dumps. MediaWiki API: `http://www.mediawiki.org/wiki/API`.

Table 1: The ten most frequent quality flaws of English Wikipedia articles along with a description and the number of articles that have been tagged with the respective cleanup tag.

Flaw name	Description	Tagged articles
Unreferenced	The article does not cite any references or sources.	273 230
Orphan	The article has fewer than three incoming links.	166 933
Refimprove	The article needs additional citations for verification.	89 686
Empty section	The article has at least one section that is empty.	46 184
Notability	The article does not meet the general notability guideline.	32 396
No footnotes	The article's sources remain unclear because of its inline citations.	26 920
Primary sources	The article relies on references to primary sources.	21 836
Wikify	The article needs to be wikified (internal links and layout).	14 333
Advert	The article is written like an advertisement.	7 186
Original research	The article contains original research.	6 630

provide further information. Moreover, 307 cleanup tags refer to the whole article (e.g., *Unreferenced* in Figure 1), whereas 81 refer to particular claims (e.g., *Citation needed*). In this paper, we target the prediction of specific flaws referring to the whole article, and hence we discard the 18 unspecific tags as well as the 81 inline tags.[6] Note in this respect that only 2,01% of the tagged articles are tagged with one of the 18 unspecific tags and that the majority (80.02%) of tagged articles are tagged with a flaw that refer to the whole article. 289 cleanup tags remain that define specific article flaws; the ten most frequent are listed in Table 1. 896 953 articles are tagged with the 289 quality flaws, out of which 76.41% are tagged with the ten flaws shown in Table 1.

4. QUALITY FLAW MODEL

The modeling of quality flaws can happen intensionally or extensionally, depending on a flaw's nature and the knowledge that is at our disposal.[7] An intensional model of a flaw f can be understood as a set of rules, which define, in a closed-class manner, the set of articles that contain f. An extensional model is given by a set of positive examples, and modeling means learning a classifier that discriminates positive instances (containing the flaw) from all other instances. Of course, the basis of both modeling paradigms are expressive features of Wikipedia articles.

4.1 Features

To this day a large body of article features—allegedly quality predicting—has been proposed; we have compiled a comprehensive breakdown, see Table 3 in Appendix A. Within our implementation all ticked features are employed, i.e., we capture the state-of-the-art with respect to feature expressiveness and information quality research. Some features are omitted since their implementation effort exceeds the expected benefit by far. In addition we devise several new features, which are marked with an asterisk. The features are organized along four dimensions: content, structure, network, and edit history. The source of information that is required for feature computation as well as the computational complexity differs for each dimension. Our model can be adjusted with respect to its transferability to other text documents than Wikipedia articles as well as to its computational complexity, by restricting to the features from a subset of the four dimensions.

Content features rely on the plain text of an article and are intended to quantify aspects like writing style and readability. Also, the new feature "special word rate", introduced to measure the presence of certain predefined words, provides evidence of unwanted content and article neutrality. E.g., peacock words, such as legendary, great, and brilliant, may be an indicator of advertising or promotional content; similarly, the presence of sentiment-bearing words can be considered as an indicator of missing neutrality. The content features can be computed with a complexity of $O(|d|)$, where $|d|$ denotes the length of an article d in characters.

Structure features are intended to quantify the organization of an article. We employ features which measure quantity, length, and nesting of sections as well as of subsections. Special attention is paid to the lead section, also called intro, as several flaws directly refer to it. Moreover, the usage of images, tables, files, and templates is quantified, as well as the categories an article belongs to; the usage of lists is quantified for the first time. Other features quantify the usage of references, including citations and footnotes and shall target flaws related to an article's verifiability. We introduce new features that check the presence of special sections that are either mandatory, such as "References", or that should be avoided, such as "Trivia". Wikiprep is used to determine the number of related pages, i.e., pages that are linked in special sections like "See also" and "Further reading".[8] The computation of structure features is governed by the complexity of parsing an article's markup, which is in $O(|d|)$.

Network features quantify an article's integration by means of hyperlinks. Here we distinguish the following types of outgoing links:

- Internal links, which point to articles in the same language.
- Inter-language links, which point to the same article in a different language.
- External links, which point to sources outside of Wikipedia.

These features count the number of outgoing links as well as their frequency relative to the article size. We introduce the feature of incoming links, where the origin has to be an article (i.e. links from disambiguation, redirect, and discussion pages are excluded). The in-link count targets the flaw *Orphan*, and the out-link counts target the flaw *Wikify*. The computation of network features is based on the link graph and is in $O(|d| \cdot |D|)$, where D denotes the set of all Wikipedia articles.

Edit history features model article evolution, which pertains to the frequency and timing of revisions as well as to the community of editors. These features have been proven valuable to classify featured articles [13, 21, 29, 34]; they address aspects like stability, maturity, and cooperation. The computation of edit history features is in $O(|d| \cdot r_d)$, where r_d denotes the number of revisions of an article d.

4.2 Intensional Modeling

The flaw descriptions in Table 1 show that three flaws from the set of the ten most frequent flaws, namely *Unreferenced*, *Orphan*, and *Empty section* can be modeled with rules based on the aforementioned features.

[6] Our prediction approach can be applied to inline flaws as well, by breaking the articles into paragraphs or sentences.

[7] For special cases also a hybrid model is conceivable, where a filtering step (intensional) precedes a learning step (extensional).

[8] `http://sourceforge.net/apps/mediawiki/wikiprep`.

An article suffers from the flaw *Unreferenced* if it does not cite any references or sources. Wikipedia provides different ways of citing sources, including inline citations, footnotes, and parenthetical referencing.[9] Here, we summarize all types of citations under the term "references". Using the structure features "reference count" and "reference sections count" we define the predicate $unreferenced(d)$:

$$unreferenced(d) = \begin{cases} 1, \text{ if } & reference\text{-}count(d) = 0 \\ & \text{and } reference\text{-}sections\text{-}count(d) = 0 \\ 0, \text{ else} \end{cases}$$

An evaluation on $D^-_{Unreferenced}$, the set of articles that have been tagged to be unreferenced, reveals that the $unreferenced$-predicate is fulfilled for 85.3% of the articles. We analyzed the remaining 14.7% and found that they actually provide references, and hence are mistagged. This observation shows a well-known problem in the Wikipedia community, and there is a WikiProject dedicated to cleanup mistagged unreferenced articles.[10] The fact that there is no such WikiProject for other quality flaws suggests that this problem is not considered to be serious for other flaws.

The *Orphan* flaw is well-defined: an article is called orphan if it has fewer than three incoming links. In this regard the following page types are not counted: disambiguation pages, redirects, soft redirects, discussion pages, and pages outside of the article namespace.[11] Using the network feature "in-link count" we define the predicate $orphan(d)$:

$$orphan(d) = \begin{cases} 1, & \text{if } \quad in\text{-}link\text{-}count(d) < 3 \\ 0, & \text{else} \end{cases}$$

An evaluation on D^-_{Orphan} reveals that the $orphan$-predicate is fulfilled for 98.4% of the articles.

An article suffers from the flaw *Empty section* if it has a section that does not contain content at all. Using the structure feature "short section length" we define the predicate $empty\_section(d)$:

$$empty\_section(d) = \begin{cases} 1, & \text{if } \quad short\text{-}section\text{-}length(d) = 0 \\ 0, & \text{else} \end{cases}$$

An evaluation on $D^-_{Empty\_section}$ reveals that the $empty\_section$-predicate is fulfilled for 99.1% of the articles.

The intensional modeling paradigm is very efficient since no training data is required and since the computation relies on few basic features. Moreover, as the above evaluations show, it is effective at the same time. Note however, that if the definition of a flaw changes, an explicit model needs to be adapted as well.

4.3 Extensional Modeling

The majority of quality flaws is defined informally and cannot be modeled by means of explicit rules (see Table 1); the knowledge is given in the form of examples instead. For an article $d \in D$ we model these flaws as a vector $\mathbf{d}$, called document model. The dimensions of $\mathbf{d}$ quantify the features ticked in Table 3, and, for a set D of Wikipedia articles, $\mathbf{D}$ denotes the set of associated document models. By means of machine learning a mathematical decision rule is computed from $\mathbf{D}$ that discriminates between elements from D^- and $D \setminus D^-$ (see Figure 2).

[9]Guidelines for citing sources: `http://en.wikipedia.org/wiki/Wikipedia:Citing_sources`.

[10]WikiProject: `http://en.wikipedia.org/wiki/Wikipedia:Mistagged_unreferenced_articles_cleanup`.

[11]Criteria for orphaned articles: `http://en.wikipedia.org/wiki/Wikipedia:Orphan#Criteria`.

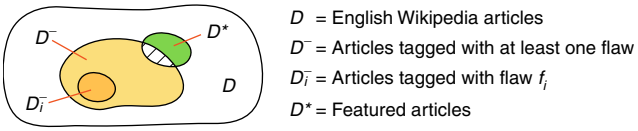

Figure 2: Sets of Wikipedia articles distinguished in this paper. Without loss of generality we assume in our experiments that the hashed area $D^- \cap D^*$ is empty, i.e., featured articles are flawless.

5. PREDICTING QUALITY FLAWS

We argue that the prediction of quality flaws is essentially a one-class problem. This section gives a formal problem definition (Section 5.1) and devises a tailored one-class machine learning approach to address the problem (Section 5.2).

5.1 Problem Statement

Let D be the set of Wikipedia articles and let F be a set of 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:[12]

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

where 2^F denotes the power set of F. Basically, there are two strategies to tackle multi-labeling problems:

1. by multiclass classification, where an individual 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 $F \setminus f_i$. When spotting 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 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.

As an additional illustration consider the flaw *Refimprove*, which is described in Table 1. An even large sample of articles that suffer from this flaw can be compiled without problems (89 686 articles have been tagged with this flaw). However, it is impossible to compile a representative sample of articles that have a reasonable number of proper citations for verification. Although many articles with sufficient citations exist (e.g., featured articles), they cannot be considered as a *representative sample*. The fact that featured articles are not representative for typical Wikipedia articles becomes

[12]Possibly existing correlations among the flaws in F will not affect the nature of the multi-labeling problem.

clear when looking at Figure 3, which shows a sample of Wikipedia articles represented under the first two principle components. Figure 3 also shows that quality flaw prediction is a significantly harder problem than discriminating featured articles. Training a binary classier using featured articles and flawed articles would lead to a biased classifier that is not able to predict flaws on the entire Wikipedia. Also, using random articles and flawed articles to train a binary classier is unacceptable because, as already mentioned, it is more than likely that many flawed articles are not yet identified. Stated another way, quality flaw prediction is a one-class problem.

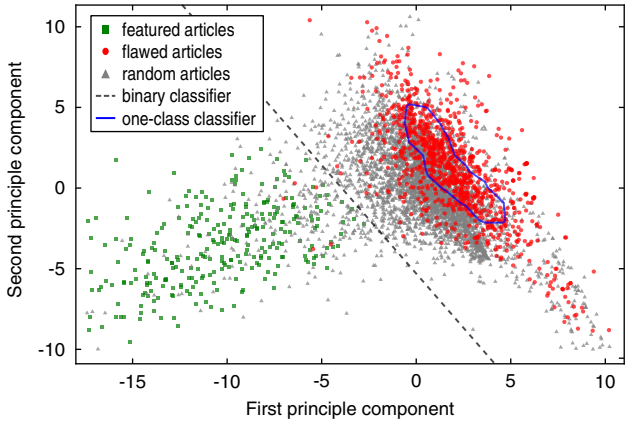

Figure 3: Distribution of featured articles, articles that are tagged with the flaw *Refimprove*, and random articles in the English Wikipedia, represented under the first two principle components. The binary classifier is trained using featured articles and flawed articles, the one-class classifier is trained solely on the set of flawed articles.

5.2 Method

Following Tax [31], three principles to construct a one-class classifier can be distinguished: density estimation methods, boundary methods, and reconstruction methods. Here we resort to a one-class classification approach as proposed by Hempstalk et al. [17], which combines density estimation with class probability estimation. There are two reasons for using this approach: (1) Hempstalk et al. show that it is able to outperform state-of-the-art approaches, including a one-class SVM, and (2) it can be used with arbitrary density estimators and class probability estimators. Instead employing an out-of-the-box classifier we apply dedicated density estimation and class probability estimation techniques to address the problem defined above.

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 $P(f_i \mid \mathbf{d})$ denote 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, i.e., 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.

The one-class classifier is built as follows: at first a class with artificial examples is generated, whereas the feature values obey a Gaussian distribution with $\mu = 0$ and $\sigma^2 = 1$. We employ the Gaussian distribution 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 where a voting over all trees is run in order to obtain a classification decision [18, 10]. The decision trees of a forest differ with respect to their features. Here, each tree is build with a subset of $log_2(|features|) + 1$ randomly chosen features, i.e., no tree minimization strategy is followed at training time. The learning algorithm stops if either all leaves contain only examples of one class or if no further splitting is possible. Each decision tree perfectly classifies the training data—but, because of its low bias the obtained generalization capability is poor [35, 25]. However, the combination of several classifiers in a voting scheme reduces the variance and introduces a stronger bias. While the bias of a random forest results from several feature sets, the bias of the bagging approach results from the employment of several training sets, and it is considered as being even stronger [9].

6. EVALUATION

We report on experiments to assess the effectiveness of our modeling and classification approach in detecting the ten most frequent quality flaws shown in Table 1. As already mentioned, 76.41% of the tagged Wikipedia articles suffer from these flaws (see Section 3.3). 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 6.1).
2. Since users (Wikipedia editors) have diverse 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 6.2).
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 6.3).

Preprocessing We use the same data basis that underlies our cleanup tag mining approach, i.e., the English Wikipedia snapshot from January 15, 2011. The articles' plain texts and wikitexts are extracted from the "pages-articles" dump, which is included in the Wikimedia snapshot and which comprises the current revisions of all articles in a single XML file of about 25GB size. The plain texts and the wikitexts form the basis to compute the content features and the structure features. Our local copy of the Wikipedia database, which is described in Section 3.1, is used to compute the network features. The computation of the history features is based on the "pages-meta-history" dump, which is included in the Wikimedia snapshot and which comprises the content of each revision in a single XML file of about 7.3TB size. The XML dumps are processed on an Apache Hadoop cluster using Google's MapReduce.

6.1 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 a way out of the dilemma is the generation of uniformly distributed outlier examples [31]. 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 2.[13] Under this setting one introduces some bias, since featured articles cannot be considered as a representative sample of Wikipedia articles (see Figure 3).

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 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.

6.2 Effectiveness of Flaw Prediction

Experiment Design The evaluation is performed for the set $F' \subset F$ of the ten most frequent quality 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 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.

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 4 illustrates the operating point analyses exemplary for the flaw *Unreferenced*: with increasing target rejection rate the recall value decreases 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).

[13]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.

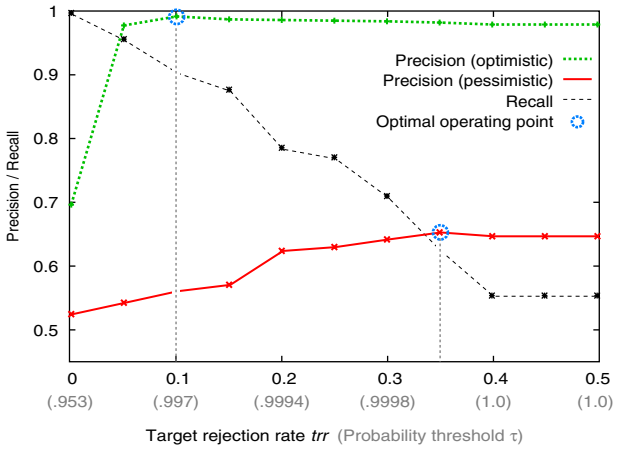

Figure 4: Precision and recall over target rejection rate for the flaw *Unreferenced*. The figure illustrates the difference in terms of precision under the optimistic setting, using featured articles as outliers, and the pessimistic setting, using random articles as outliers. The recall is the same under both settings. The optimal operating points correspond to the target rejection rates that maximizes classifier precision.

The precision of a one-class classifier cannot be adjusted arbitrarily since the target rejection rate controls only the probability 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 4).

Results Table 2 shows the performance values for each of the ten quality flaws. The values correspond to the performance at the respective optimal operating point. The performance is quantified as precision (*prec*) and recall (*rec*). We also report the area under ROC curves (AUC) [15], 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. As expected, the precision values for the flaws *Unreferenced*, *Orphan*, and *Empty section* are very high; recall that these flaws can also be modeled intensionally. 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 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.

Table 2: Individual performance values for each of the ten most frequent 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	Optimistic setting			Pessimistic setting			Flaw
		prec	*rec*	AUC	*prec*	*rec*	AUC	ratio
f_1	Unreferenced	0.99	0.90	0.95	0.63	0.63	0.63	1:3
f_2	Orphan	1.00	0.90	0.95	0.72	0.59	0.68	1:5
f_3	Refimprove	0.83	0.87	0.85	0.57	0.56	0.57	1:10
f_4	Empty section	0.90	0.70	0.82	0.74	0.70	0.72	1:21
f_5	Notability	0.99	0.96	0.98	0.66	0.61	0.65	1:30
f_6	No footnotes	0.82	0.87	0.84	0.59	0.59	0.58	1:36
f_7	Primary sources	0.94	0.90	0.92	0.61	0.59	0.61	1:44
f_8	Wikify	0.96	0.87	0.92	0.64	0.58	0.63	1:68
f_9	Advert	0.86	0.91	0.88	0.65	0.58	0.63	1:136
f_{10}	Original research	0.76	0.64	0.71	0.56	0.80	0.59	1:147

6.3 Flaw-specific Class Imbalances

The performance values in Table 2 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, 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:

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

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 5 shows the precision values as a function of the flaw distribution under the optimistic setting.

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^-. Table 2 lists the estimated flaw ratio for each of the ten most frequent flaws. For example, the ratio of the flaw *Unreferenced* is about 1:3 (273 230 : 979 299). In other words, about every fourth article is expected to contain this flaw.

Figure 5 shows 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 prediction performance is independent of the class imbalance. Although the flaw ratio of the flaw *Notability* is 1:30, 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 regarding the employed document model, the weak performance indicates also that the training set of the one-class classifiers may be too small.

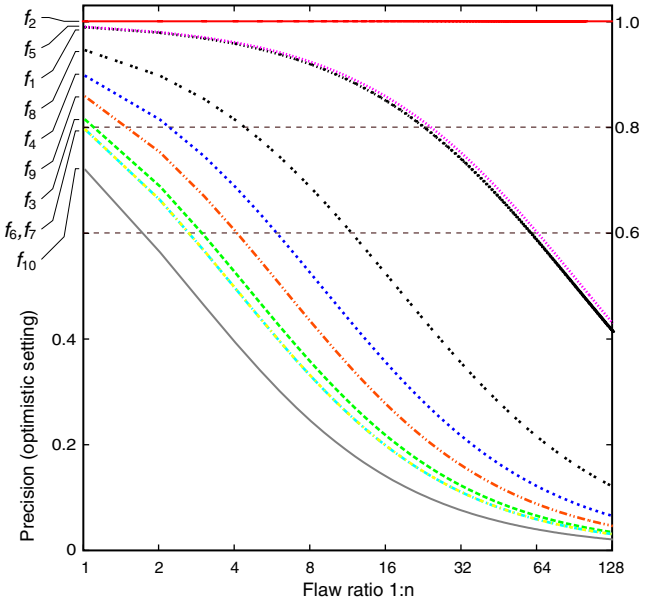

Figure 5: Precision in the optimistic setting over flaw ratio for the ten most frequent quality flaws: 1:n (flawed articles : flawless articles) with $n \in [1; 128]$. The figure puts the classification performances reported in Table 2 into perspective, since it considers imbalances in the test sets that might occur in the wild.

7. CONCLUSIONS AND OUTLOOK

We treat quality flaw prediction as a process where for each known flaw an expert is asked whether or not a given Wikipedia article suffers from it; the experts in turn are operationalized by one-class classifiers. The underlying document model combines several new features with the state-of-the-art quality assessment features. Our evaluation is based on a corpus comprising 10 000 human-labeled Wikipedia articles, compiled by utilizing cleanup tags. We report on precision values close to 1 for four out of ten important 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 without human interaction. We plan to operationalize our classification technology in the form of a Wikipedia bot that autonomously identifies and tags flawed articles. Our approach also supports the principle of *intelligent task routing* [11], which addresses the automatic delegation of particular flaws to appropriate human editors. Though the proposed quality flaw prediction approach is evaluated in the Wikipedia context, it is also applicable to other user-generated Web applications where cleanup tags are used.

Our current research targets the development of knowledge-based predictors for individual quality flaws. Instead of resorting to a single document model, we develop a flaw-specific view that combines feature selection, expert rules, and multi-level filtering. In this respect, we analyze in detail which features prove essential for the prediction of a certain quality flaw and how effective the newly introduced features are. Moreover, instead of resorting to a single learning approach, we investigate the amenability of different one-class classification approaches with respect to the different flaws. We are also investigating whether a learning approach can benefit from the untagged articles, e.g., using partially supervised classification or PU-learning [23].

8. REFERENCES

[1] B. Adler and L. de Alfaro. A content-driven reputation system for the Wikipedia. In *Proc. of WWW'07*, pages 261–270, 2007.

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

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

[4] M. Anderka, B. Stein, and N. Lipka. Detection of text quality flaws as a one-class classification problem. In *Proc. of CIKM'11*, pages 2313–2316, 2011.

[5] M. Anderka and B. Stein. A breakdown of quality flaws in Wikipedia. In Proc. of WebQuality'12, pages 11–18, 2012.

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

[7] M. Bendersky, W. B. Croft, and Y. Diao. Quality-biased ranking of Web documents. In *Proc. of WSDM'11*, pages 95–104, 2011.

[8] J. Blumenstock. Size matters: word count as a measure of quality on Wikipedia. In *Proc. of WWW'08*, pages 1095–1096, 2008.

[9] L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.

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

[11] D. Cosley, D. Frankowski, L. Terveen, and J. Riedl. Using intelligent task routing and contribution review to help communities build artifacts of lasting value. In *Proc. of CHI'06*, pages 1037–1046, 2006.

[12] T. Cross. Puppy smoothies: improving the reliability of open, collaborative wikis. *First Monday*, 11(9), 2006.

[13] 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 *Proc. of JCDL'09*, pages 295–304, 2009.

[14] W. Emigh and S. Herring. Collaborative authoring on the Web: a genre analysis of online encyclopedias. In *Proc. of HICSS'05*, 2005.

[15] T. Fawcett. Roc graphs: Notes and practical considerations for researchers. Technical report, HP Laboratories, 2004.

[16] J. Giles. Internet encyclopaedias go head to head. *Nature*, 438(7070):900–901, 2005.

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

[18] T. K. Ho. Random decision forests. In *Proc. of ICDAR'95*, pages 278–282, 1995.

[19] M. Hu, E. Lim, A. Sun, H. Lauw, and B. Vuong. Measuring article quality in Wikipedia: models and evaluation. In *Proc. of CIKM'07*, pages 243–252, 2007.

[20] Y. Lee, D. Strong, B. Kahn, and R. Wang. AIMQ: a methodology for information quality assessment. *Information and Management*, 40(2):133–146, 2002.

[21] A. Lih. Wikipedia as participatory journalism: reliable sources? Metrics for evaluating collaborative media as a news resource. In *Proc. of ISOJ'04*, 2004.

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

[23] B. Liu, Y. Dai, X. Li, W. S. Lee and P. Yu. Building text classifiers using positive and unlabeled examples. In *Proc. of ICDM'03*, pages 179–186, 2003.

[24] S. Madnick, R. Wang, Y. Lee, and H. Zhu. Overview and framework for data and information quality research. *Journal of data and information quality*, 1(1):1–22, 2009.

[25] T. Mitchell. *Machine Learning*. McGraw-Hill Higher Education, 1997.

[26] A. Pirkola and T. Talvensaari. A topic-specific Web search system focusing on quality pages. In *Proc. of ECDL'10*, pages 490–493, 2010.

[27] M. Potthast, B. Stein, and R. Gerling. Automatic vandalism detection in Wikipedia. In *Proc. of ECIR'08*, pages 663–668, 2008.

[28] J. Slone. *Information quality strategy: an empirical investigation of the relationship between information quality improvements and organizational outcomes*. PhD thesis, Capella University, 2006.

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

[30] B. Stvilia, M. Twidale, L. Smith, and L. Gasser. Information quality work organization in Wikipedia. *Journal of the american society for information science and technology*, 59(6):983–1001, 2008.

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

[32] F. Viégas, M. Wattenberg, and K. Dave. Studying cooperation and conflict between authors with history flow visualizations. In *Proc. of CHI'04*, pages 575–582, 2004.

[33] R. Wang and D. Strong. Beyond accuracy: what data quality means to data consumers. *Journal of management information systems*, 12(4):5–33, 1996.

[34] D. Wilkinson and B. Huberman. Cooperation and quality in Wikipedia. In *Proc. of WikiSym'07*, pages 157–164, 2007.

[35] D. Wilson and R. Randall. Bias and the probability of generalization. In *Proc. of IIS'97*, pages 108–114, 1997.

[36] The Wall Street Journal. Jimmy Wales on Wikipedia quality and tips for contributors. November 2009. URL: `http://blogs.wsj.com/digits/2009/11/06`.

[37] H. Zeng, M. Alhossaini, L. Ding, R. Fikes, and D. McGuinness. Computing trust from revision history. In *Proc. of PST'06*, 2006.

[38] Y. Zhou and B. W. Croft. Document quality models for Web ad hoc retrieval. In *Proc. of CIKM'05*, pages 331–332, 2005.

[39] X. Zhu and S. Gauch. Incorporating quality metrics in centralized/distributed information retrieval on the World Wide Web. In *Proc. of SIGIR'00*, pages 288–295, 2000.

APPENDIX

A. SURVEY OF ARTICLE FEATURES

A variety of features has been proposed in the literature on automatic quality assessment of Wikipedia articles (the relevant literature is reviewed in Section 2). Table 3 gives a comprehensive overview of the proposed features, organized along the four dimensions content, structure, network, and edit history (the dimensions are described in Section 4.1). The overview is intended as a resource for other researcher, and we consider it as the first complete compilation of this kind.

Table 3: Overview of Wikipedia article features, classified along four dimensions: content, structure, network, and edit history. Features that are employed within our document model are marked with ticks (✓), new features that are used for the first time are marked with asterisks (*).

Feature	Description	Reference
Content		
Character count	Number of characters	[8, 13, 29] ✓
Word count	Number of words	[8, 19, 22] ✓
* Word length	Average word length in characters	✓
Syllables count	Number of syllables	[8] ✓
* Word syllables	Average syllables per word	✓
One-syllable word count	Number of one-syllable words	[8] ✓
* One-syllable word rate	Percentage of one-syllable words	✓
Sentence count	Number of sentences	[8, 13] ✓
* Sentence length	Average sentence length in words	✓
Long/short sentence rate	Percentage of long sentences (at least 48 words) and short sentences (at most 33 words)	[13] ✓
Question rate	Percentage of questions	[13] ✓
Passive sentence rate	Percentage of passive voice sentences	[13] ✓
* Long/short sentence length	Number of words of the longest and shortest sentence	✓
Paragraph count	Number of paragraphs	[13] ✓
Paragraph length	Average paragraph length in sentences	[13] ✓
Readability indices	Flesh, Kincaid, Lix, ARI, SMOG-Grading, Gunning Fog, Coleman-Liau, Bormuth, Dale-Chall, Miyazaki	[8, 29] ✓
Word usage rate	Percentage of auxiliary and "to be" verbs, conjunctions, pronouns, prepositions, normalizations	[13] ✓
Sentence beginning rate	Percentage of sentences beginning with a pronoun, interrogative pronoun, article, conjunction, subordinating conjunction, preposition	[13] ✓
* Special word rate	Percentage of weasel-, peacock-, doubt-, editorializing-, long- (> 7 characters), easy- (in Dale-Chall word list), difficult- (not in Dale-Chall word list), complex- (> 3 syllables), stop words, idioms, aphorisms, proverbs	✓
Information-to-noise ratio	Vocabulary size / word count	[29] ✓
Trigrams	Character and part of speech trigrams	[22]
Structure		
Section count	Number of (sub-, subsub-) sections and total number of sections	[8, 13] ✓
Section length	Average (sub-, subsub-) section length in words	[13] ✓
Long/short section length	Number of words of the longest and shortest (sub-, subsub-) section	[13] ✓
Section nesting	Average number of (sub-) subsections per (sub-) section	[13] ✓
Heading count	Total number of sub-, subsub-, and section headings	[8, 13] ✓
Lead length	Number of words / characters of the lead section	[13] ✓
Lead rate	Percentage of words in the lead section	[13] ✓
Image count	Number of images	[8, 13, 29] ✓
Table count	Number of tables	[8] ✓
File count	Number of files (videos, pdfs etc.)	[8] ✓
Category count	Number of Wikipedia categories the article belongs to	[8] ✓
* Template count	Number of (different) Wikipedia templates	✓
* List ratio	Number of words in lists / word count	✓
Reference count	Number of references (*ref*-tags and citation templates)	[8, 13] ✓
Reference rates	Reference count / word count and reference count / section count	[13] ✓
* Reference sections count	Number of reference sections, e.g., "References", "Footnotes", "Sources", "Bibliography", "Citations"	✓
* Trivia sections count	Number of trivia sections, e.g., "Trivia", "Miscellanea", "Facts", "Other facts"	✓
* Mandatory sections count	Number of mandatory sections, e.g., "Further reading", "See also"	✓
* Related page count	Number of related pages, based on "See also", "Further reading", etc. (Wikiprep)	✓
Network		
Internal link count	Number of outgoing internal links	[8, 29] ✓
Broken internal link count	Number of outgoing internal broken links	[29] ✓
Language link count	Number of outgoing inter-language links	[13] ✓
External link count	Number of outgoing external links	[8, 13, 29] ✓
Out-link rate	Number of all out-links / word count	[13] ✓
* In-link count	Number of incoming internal links (from articles)	✓
PageRank	Google PageRank of an article	[13] ✓
Citation measures	In- and out-degree, associativity, clustering coefficient, reciprocity	[13]
Edit history		
Age	Age in days	[29] ✓
Currency	Days between last update and now (date of the snapshot)	[29] ✓
Edit count	Number of edits	[13, 21, 29, 34] ✓
Editor count	Number of distinct editors	[21, 29, 34] ✓
Editor role	Number of registered, anonymous, and admin editors	[13, 29] ✓
Editor rate	Number of edits per editor	[13, 34] ✓
Revert count	Number of reverts	[29]
Revert time	Median revert time in minutes	[29]
Edit rates	Ratio between age and edit count, percentage of edits per day	[13] ✓
Connectivity	Number of articles with common editors	[29] ✓
Cooperation	Revisions of the discussion page	[13, 34] ✓
Edit amount	Number of modified lines, compared to an older revision	[13] ✓
Edit currency rate	Percentage of edits made in the last 3 months	[13] ✓
Special editor rates	Percentage of edits of infrequent editors, percentage of edits made by the top 5% editors	[13] ✓