Predicting the Perceived Quality of Web Forum Posts

Markus Weimer, Iryna Gurevych
Ubiquitous Knowledge Processing Lab, Telecooperation Division
Technische Universität Darmstadt, Germany
http://www.ukp.informatik.tu-darmstadt.de
[mweimer,gurevych]@tk.informatik.tu-darmstadt.de

Abstract

Assessing the quality of user generated content is an important problem of Web 2.0. Currently, most web sites need their users to rate content manually, which is labour intensive and thus happens rarely. The automatic systems in the literature are limited to one kind or domain of discourse.

We propose a system to assess the quality of user generated discourse *automatically*. Our system learns from human ratings by applying SVM classification based on features such as *Surface, Lexical, Syntactic, Forum specific and Similarity features*.

Our system has also shown to be adaptable to different domains of discourse in our experiments on three different web forum data sets. The system outperformed the majority class baseline for all three data sets. Our best performing system configuration achieves an accuracy of 89.1%, which is significantly higher than the baseline of 61.82%.

1 Introduction

User generated content is a significant part of Web 2.0. It is characterized by a low publication threshold and a general lack of editorial control. Content is not created by professionally trained authors, but by ordinary users. We focus on automatic quality assessment of *user generated discourse*, which is textual user generated content. User generated discourse occurs for example in systems like Blogs, Wikis, Forums, and Product Reviews.

The nature of its creation not only leads to huge amounts of user generated discourse being created, but also to a varying quality of the content: Much of it is of great value to users, while many parts of it are of bad quality. Thus, users have problems to navigate through these large repositories of information and find information of high quality quickly.

In order to address the information navigation problem outlined above, many web sites, like Google Groups[1] and Nabble[2], have introduced rating mechanisms. Users are asked to rate the content available on the site which has been submitted by other users of the forum. Typically, this rating is expressed on a five-star rating scale. The number of stars corresponds to categories such as *Poor Post* or *Excellent Post*. Table 1 shows the categories as used by Nabble.

User ratings have been shown to be consistent with the user community at large by Lampe and Resnick [2004]. They also showed that user ratings lead to the problem of *premature negative consent*, when combined with filtering based on these ratings. Posts that are once rated to fall below the filtering threshold are not shown to the users anymore. Thus, they can never be rated up again. Additionally, the percentage of manually rated posts is typically very low (about 0.1% in Nabble).

Addressing these issues and departing from pure manual ratings, the main idea explored in the present paper is to investigate the feasibility of automatically assessing the *perceived quality* of user generated discourse, as expressed by the ratings given by the users. The *perceived quality* is not an objective measure. Rather, it models how the community at large perceives quality. We evaluate a machine learning approach to automatically assess it.

The main contributions of the present paper are: (1) A domain-independent system for automatic quality assessment of forum posts that learns from human ratings. Thus, the system adapts itself to new domains of discourse. We evaluate the system on real web forum discussions extracted from Nabble.com. (2) An analysis of the usefulness of different classes of features for the prediction of post quality in different forums.

[1] http://groups.google.com
[2] http://www.nabble.com

2 Related work

Quality assessment of user generated discourse is a new field of research and has been addressed only recently by Weimer et al. [2007] in a first case study. The authors present a similar system to the one discussed in this paper. However, they only apply it to one domain of discussion and thus do not reach the broad applicability we focus on.

There has also been some work on automatic assessment of product review usefulness by Kim et al. [2006c]. They test their system on data from Amazon.com, where users can submit reviews of products. These reviews are then rated by other users for their helpfulness, by answering the clear question "Was this review helpful to you?" with the answer choices *Yes/No*. This study found that the dominant features to predict these ratings are the length of the reviews as well as the rating given to the product on a five star scale by the review. Please note that review helpfulness is a rather clearly defined term on the website. This is not the case for post ratings in web forums.

Automatic essay scoring: One closely related field is the area of automatic essay scoring (Valenti et al. [2003], Chodorow and Burstein [2004], Attali and Burstein [2006]). There, the goal is to automatically assess the grade of an essay written by students. This seems very similar to what we propose in the present paper. However, there exist well established guidelines that define what a good essay is. Thus, these systems do not need to adapt to the prevalent quality standards of the data they are applied to as our system has to. In web forums, different users cast their rating with possibly different quality criteria in mind.

Web forum analysis: Web forums have been in the focus of another track of research, in particular in the context of eLearning. Kim et al. [2006b] found that the relation between a student's posting behavior and the grade obtained by that student can be predicted automatically. To do so, the number of posts, the average post length and the average number of replies to posts of the student have been shown to be the most important features.

In related research, Feng et al. [2006] describe a system to find the most authoritative answer in a forum thread, based amongst others on the author's trustworthiness and lexical similarity. Kim et al. [2006a] add speech act analysis as a feature to their system. Finding the most authoritative post in a thread seems to be very closely related to the task we focus on. However, it is definitely different, as we assess the perceived quality of a given post, currently based solely on its intrinsic features. Any discussion thread may contain an indefinite number of good posts, rather than a single authoritative one.

3 Experiments

The system that we propose should be able to adapt to the quality standards existing in a certain user community by learning the relation between a set of features and the perceived quality of posts. We currently employ features from five classes described in Table 2: *Surface, Lexical, Syntactic, Forum specific and Similarity features.*

3.1 Data

We evaluated our systems on three data sets extracted from discussions on Nabble.com. Nabble.com hosts forums, but also bridges conventional mailing lists into their system. Forums at Nabble.com are categorized. Analysis of the data showed that most of the rated posts are within the "Software" category.[5] As we seek to develop a system that is applicable to many domains of discussion, we extracted the following three data sets that allow us to assess its performance with that respect: **ALL:** All rated posts in the database. This is the broadest of all data sets. **SOFT:** All rated posts of forums that are in the software category. These are posts that concern closely related. This data set is the same as used by Weimer et al. [2007]. **MISC:** All posts that are in ALL, but not in SOFT. This data set is very diverse in topic, even more so than ALL, as half of ALL are posts from SOFT. Topics range from discussions amongst wikipedia community members to discussions of motor bikes.

At Nabble, posts can be rated by multiple users. Table 1 shows the distribution of average ratings on the five star scale employed by Nabble. From this statistics, it becomes evident that users at Nabble prefer extreme ratings. Therefore, we define the task of predicting the post quality as a binary classification task. Posts with less than three stars are rated as "bad". Posts with more than three stars are "good".

We removed the posts, where all ratings are exactly three stars. We also removed the posts that had contradictory ratings from different users. Manual analysis of those posts revealed that they were mostly spam, which was voted high for commercial interest and voted down for being spam.

[5] `http://www.nabble.com/Software-f94.html`

Stars	Label	ALL		SOFT		MISC	
⋆	Poor	1928	45%	1251	63%	677	29%
⋆⋆	Below Avg.	120	3%	44	2%	76	3%
⋆⋆⋆	Average	185	4%	69	4%	116	5%
⋆⋆⋆⋆	Above Avg	326	8%	183	9%	143	6%
⋆⋆⋆⋆⋆	Excellent	1732	40%	421	21%	1311	56%

Table 1: *Categories and their usage frequency. Data on the SOFT data set taken from (Weimer et al. [2007]).*

Feature category	Feature name	Description
Surface Features	Length	The number of tokens in a post.
	Question Frequency	The percentage of sentences ending with "?".
	Exclamation Frequency	The percentage of sentences ending with "!".
	Capital Word Frequency	The percentage of words in CAPITAL, which is often associated with shouting.
Lexical Features Wording of the posts	Spelling Error Frequency	The percentage of words that are not spelled correctly.[3]
	Swear Word Frequency	The percentage of words that are on a list of swear words we compiled from resources like WordNet and Wikipedia[4], which contains more than eighty words like "asshole", but also common transcriptions like "f*ckin".
Syntactic Features		The percentage of part-of-speech tags as defined in the PENN Treebank tag set Marcus et al. [1994]. We used TreeTagger Schmid [1995] based on the english parameter files supplied with it.
Forum specific features Properties that are only present in forum postings	IsHTML	Whether or not a post contains HTML. In our data, this is encoded explicitly, but it can also be determined by regular expressions matching HTML tags.
	IsMail	Whether or not a post has been copied from a mailing list. This is encoded explicitly in our data.
	Quote Fraction	The fraction of characters that are inside quotes of other posts. These quotes are marked explicitly in our data.
	URL and Path Count	The number of URLs and filesystem paths. Post quality in the software domain may be influenced by the amount of tangible information, which is partly captured by these features.
Similarity features		Forums are focussed on a topic. The relatedness of a post to the topic of the forum may influence post quality. We capture this relatedness by the cosine between the posts unigram vector and the unigram vector of the forum.

Table 2: *Features used for the automatic quality assessment of posts.*

We also filtered out the posts that did not contain any text, but only attachments like pictures and program files. Finally, we removed non-English posts using a simple heuristics: Posts that contained a certain percentage of words above a predefined threshold, which are non-English according to an English dictionary, were considered to be non-English. The upper part of Table 3 shows how many posts were removed from the three data sets. Please note that we did the filtering independently for each filter. Thus, posts that matched several filtering criteria are listed more than once. The lower part of that table shows the distribution of good and bad posts after filtering.

3.2 Evaluation procedure

Using the features described in Table 2, we compiled a feature vector for each post. Feature values that were not normalized by definition were scaled to the range $[0.0, \ldots, 1.0]$. To classify the posts, we use support vector machines. In particular, we used a C-SVM with a gaussian RBF kernel as implemented by LibSVM in the YALE toolkit (Mierswa et al. [2006]) in all experiments. We did not perform model selection or fine-tuned the parameters of the SVM or the kernel. The parameters were fixed to $C = 10$ and $\gamma = 0.1$ for all experiments. We performed stratified ten-fold cross validation for performance evaluation.[6]

Several randomly chosen experiments were repeated using the leave one out evaluation scheme. They yielded comparable results to the ones obtained using cross validation. Thus, we only report the latter in this paper. Please note that it is inherently hard to compare the performance of different machine learning algorithms or algorithm configurations and that statistical signifi-

[6] (See (Bishop [2006]) for an in-depth description.

	ALL		SOFT		MISC	
Unfiltered Posts	4291		1968		2323	
All ratings three stars	135	3%	61	3%	74	3%
Contradictory ratings	70	2%	14	1%	56	2%
No text	56	1%	30	2%	26	1%
Non-English	668	15%	361	18%	307	13%
Remaining	3418	80%	1532	78%	1886	81%
Good Posts	1829	54%	947	62%	1244	66%
Bad Posts	1589	46%	585	38%	642	34%

Table 3: *Number of posts filtered out in the different data sets.*

cance of cross validation performance values can be forged to be arbitrarily high when comparing two algorithms or algorithm configurations (see Witten and Frank [2005], chapter 5.5). Thus, we do not report it.

3.3 Experimental Results

Table 4 shows the average cross validation accuracy for all combinations of feature and data sets, whereas we reproduce the results of Weimer et al. [2007] for the SOFT data set. The baseline is based on the majority class. All results but one (SIM/ALL) are equal to or better than the baseline. The usage of all features results in the best or close to best performance for all data sets. The results on the MISC data set are only slightly better than the baseline. The gains on the SOFT and ALL data sets over the baseline are significant. Naively, one may think that the performance on the ALL data set is the average between the performance on MISC and SOFT, as both form approximately one half of the data in ALL. Our results are different, and the performance on ALL is comparable to the performance on SOFT. Thus, the system is able to learn how to classify posts in MISC from posts in SOFT. This leads us to believe that the rating structure in some posts of the MISC data set is very close to the SOFT data set, while the overall rating structure is too diverse to be captured correctly by our system.

The difference in rating structure also shows in the analysis of the best performing feature categories, which are different for each data set. For MISC, the surface features perform best. For SOFT, the forum specific features work best, when only one feature category is used. Weimer et al. [2007] discuss in greater detail, which features from that category have the biggest impact on overall performance. For ALL, two categories share that position: lexical features as well as forum specific features.

It is useful to have a look at the performance of all other feature categories, when the single best one is not present to assess the influence of the best feature category on the overall performance. For MISC, this leads to a performance on the baseline level. For SOFT, the drop in performance is much smaller, yet still measurable. For ALL, the effects are the smallest, being almost zero for the removal of the lexical features.

ALL:

	true good	true bad	sum
pred. good	1517	456	1973
pred. bad	312	1133	1445
sum	1829	1589	3418

SOFT:

	true good	true bad	sum
pred. good	490	72	562
pred. bad	95	875	970
sum	585	947	1532

MISC:

	true good	true bad	sum
pred. good	1231	516	1747
pred. bad	13	126	139
sum	1244	642	1886

Table 5: *Confusion matrix for the system using all features on the three different datasets.*

3.4 Error analysis

Table 5 contains the confusion matrix for the system using all features on the three data sets. The system produces approximately an equal amount of false positives and false negatives on the ALL and SOFT data sets. However, it has a tendency towards false positives on the MISC data set.

Below, we will give descriptions of common errors of our system as well as some examples from

SUF	LEX	SYN	FOR	SIM	ALL	SOFT	MISC
√	√	√	√	√	77.53% (1.45)	89.10% (1.44)	71.95% (1.09)
√	–	–	–	–	64.72% (1.21)	61.82% (1.00)	**71.31%** (1.08)
–	√	–	–	–	**74.08%** (1.38)	71.82% (1.16)	65.96% (1.00)
–	–	√	–	–	69.18% (1.29)	82.64% (1.34)	66.70% (1.01)
–	–	–	√	–	**74.08%** (1.38)	**85.05%** (1.36)	65.96% (1.00)
–	–	–	–	√	46.49% (0.87)	62.01% (1.00)	65.96% (1.00)
–	√	√	√	√	75.92% (1.42)	89.10% (1.44)	66.60% (1.01)
√	–	√	√	√	**77.39%** (1.45)	**89.36%** (1.46)	72.00% (1.09)
√	√	–	√	√	76.27% (1.43)	85.03% (1.38)	70.03% (1.06)
√	√	√	–	√	72.82% (1.36)	82.90% (1.34)	71.74% (1.08)
√	√	√	√	–	76.83% (1.44)	88.97% (1.44)	**72.43%** (1.10)
Baseline					53.51% (1.00)	61.82% (1.00)	65.96% (1.00)

Table 4: *Accuracy with different feature sets. SUF: Surface, LEX: Lexical, SYN: Syntax, FOR: Forum specific, SIM: similarity. The* baseline *results from a majority class classifier.*

the data. We will also provide conclusions on how to improve the current system to overcome the errors. Note that some of the problems were also discussed by Weimer et al. [2007]. We include their analysis, but group it with the errors on the other data sets and discuss means to overcome the limitations of the system.

Ratings based on domain knowledge: The following post from the SOFT data set shows no apparent reason to be rated badly. The human rating of this post seems to be dependent on deep domain knowledge, which is currently not present in our system.

```
> Thank You for the fast response, but I'm not
> sure if I understand you right.  INTERRUPTs can
> be interrupted (by other interrupts or signals) and
> SIGNALS not.

Yup.  And I responded faster than my brain could
shift gears and got my INTERRUPT and SIGNAL crossed.

> All my questions still remain!

Believe J"org addressed everything in full.  That the
compiler simply can't know that other routines have
left __zero_reg__ alone and the compiler expects to
find zero there.
As for SREG, no telling what another routine was
doing with the status bits so it too has to be saved
and restored before any of its contents possibly get
modified.  CISC CPUs do this for you when stacking
the IRQ, and on RTI.
```

Automatically generated mails: Sometimes, automatically generated mails like error messages end up on the mailing lists. These mails can be written very nicely and are thus misclassified by our system as good posts, while they are bad posts from the point of view of the users. One could deal with these posts by integrating features of the sender of the message, as they originate from addresses like `postmaster@domain.com`.

Non-textual content: Especially the SOFT data set contains posts that mainly consist of non-textual parts like source code, digital signatures and log messages from programs. This content confuses our system to misclassify these posts as bad posts.

To overcome this problem, the non-textual parts need to be marked. They can then be ignored in the quality assessment of the textual content. Additionally, the presence and the amount of non-textual content can be used as an additional feature.

Very short posts: Posts which contain only a few words show up as false positives and false negatives equally, as for example a simple "yes" from the master of a certain field might be regarded as a very good post, while a short insult in another forum might be regarded as a very bad post. Domain knowledge from external sources might be helpful in rating these posts.

Opinion based ratings: Some ratings do not rate the *quality* of a post, but the *expressed opinion*. In these cases, the rating is an alternative to posting a reply to the message saying "I do not agree with you".

Take for example the following post which is part of a discussion amongst Wikipedia community members from the MISC data which has been misclassified as a bad post:

```
> But you would impose US law even in a country where
> smoking weed is legal
Given that most of our users and most significant
press coverage is American, yes.  That is why I drew
the line there.
Yes, I know it isn't perfect.  But it's better than
anything else I've seen.
```

Such posts form a hard challenge for automatic systems. However, they may also form the upper bound for this task: Humans are unlikely to predict these ratings correctly without additional knowledge about the rater.

Posts that could be rated based on the reply structure: Most of the posts discussed

above could be classified correctly if the replies to them provided some cues to the quality of the post. The attractive property of integrating features of the replies into the features of a post is that it is domain independent. For example, the simple presence or absence of replies could be part of the perceived quality of a post.

4 Conclusions and future work

Assessing post quality is an important problem for web forums. Currently, most forums need their users to rate the posts manually, which is labour intensive and thus happens rarely.

We presented a system and evaluated it on different data sets from different domains of discussion. Our system has shown to be able to assess the quality of forum posts from very diverse discussion domains. The system applies SVM classification using features such as *Surface, Lexical, Syntactic, Forum specific and Similarity features* to do so. We evaluated our system on three data sets and it performed very well on two of them, while only slightly better than the baseline on the third, most challenging, one. Our best performing system configuration achieves an accuracy of 89.1%, which is significantly higher than the baseline of 61.82%.

Careful error analysis leads us to several future improvements to our system. First of all, the integration of the discourse structure promises improvements. Additionally, external knowledge sources can help to assess the information content of a post, which can be of influence on the perceived post quality.

After evaluating it on different domains of discussion within the same kind of user generated content, we seek to apply our system to other kinds of user generated discourse. The system can obviously be applied to other web forums, but we also seek to apply it to adjunct areas like blog comments and several kinds of user reviews of movies, products, websites.

We believe that this system will support important applications beyond content filtering like automatic summarization systems and user generated discourse specific search.

References

Y. Attali and J. Burstein. Automated essay scoring with e-rater v.2. *The Journal of Technology, Learning, and Assessment*, 4(3), February 2006.

C. M. Bishop. *Pattern Recognition and Machine Learning*. Information Science and Statistics. Springer, 2006.

M. Chodorow and J. Burstein. Beyond essay length: Evaluating e-raters performance on toefl essays. Technical report, ETS, 2004.

D. Feng, E. Shaw, J. Kim, and E. Hovy. Learning to detect conversation focus of threaded discussions. In *Proceedings of HLT-NNACL*, 2006.

J. Kim, G. Chern, D. Feng, E. Shaw, and E. Hovya. Mining and assessing discussions on the web through speech act analysis. In *Proc. of the Workshop on Web Content Mining with Human Language Technologies at ISWC*, 2006a.

J. Kim, E. Shaw, D. Feng, C. Beal, and E. Hovy. Modeling and assessing student activities in on-line discussions. In *Proceedings of the Workshop on Educational Data Mining at AAAI*, Boston, MA, 2006b.

S.-M. Kim, P. Pantel, T. Chklovski, and M. Penneacchiotti. Automatically assessing review helpfulness. In *Proceedings of EMNLP*, 2006c.

C. Lampe and P. Resnick. Slash(dot) and burn: Distributed moderation in a large online conversation space. In *Proceedings of ACM CHI 2004 Conference on Human Factors in Computing Systems, Vienna Austria*, pages 543–550, 2004.

M. P. Marcus, B. Santorini, and M. A. Marcinkiewicz. Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2): 313–330, 1994.

I. Mierswa, M. Wurst, R. Klinkenberg, M. Scholz, and T. Euler. YALE: Rapid prototyping for complex data mining tasks. In *KDD '06: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 935–940, New York, NY, USA, 2006. ACM Press.

H. Schmid. Probabilistic Part-of-Speech Tagging Using Decision Trees. In *International Conference on New Methods in Language Processing*, Manchester, UK, 1995.

S. Valenti, F. Neri, and A. Cucchiarelli. An overview of current research on automated essay grading. *Journal of Information Technology Education*, 2:319–329, 2003.

M. Weimer, I. Gurevych, and M. Mühlhäuser. Automatically assessing the post quality in online discussions on software. In *Companion Volume of the 45rd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2007.

I. H. Witten and E. Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, San Francisco, 2 edition, 2005.