It's all in the Content: State of the art Best Answer Prediction based on Discretisation of Shallow Linguistic Features

George Gkotsis, Karen Stepanyan,
Carlos Pedrinaci, John Domingue
Knowledge Media Institute
The Open University
Milton Keynes, UK
firstname.lastname@open.ac.uk

Maria Liakata
Dept. of Computer Science
University of Warwick
Coventry, UK
m.liakata@warwick.ac.uk

ABSTRACT

This paper addresses the problem of determining the best answer in Community-based Question Answering websites by focussing on the content. Previous research on this topic relies on the exploitation of community feedback on the answers, which involves rating of either users (e.g., reputation) or answers (e.g. scores manually assigned to answers). We propose a new technique that leverages the content/textual features of answers in a novel way. Our approach delivers better results than related linguistics-based solutions and manages to match rating-based approaches. More specifically, the gain in performance is achieved by rendering the values of these features into a discretised form. We also show how our technique manages to deliver equally good results in real-time settings, as opposed to having to rely on information not always readily available, such as user ratings and answer scores. We ran an evaluation on 21 StackExchange websites covering around 4 million questions and more than 8 million answers. We obtain 84% average precision and 70% recall, which shows that our technique is robust, effective, and widely applicable.

Categories and Subject Descriptors

H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing—*linguistic processing*

General Terms

Algorithms, Design, Experimentation

Keywords

Community Question Answering, Social Media

WebSci'14, June 23–26, 2014, Bloomington, IN, USA.
Copyright 2014 ACM 978-1-4503-2622-3/14/06 ...$15.00.
http://dx.doi.org/10.1145/2615569.2615681.

1. INTRODUCTION

The proliferation of Community-based Question Answering (CQA) websites and their corresponding data has drawn the attention of computer science researchers. The solution to the problem of identification of the best answer is expected to bring several benefits. First of all, since several answers are provided for each question, the readers of these websites will be able to process the candidate answers more efficiently and mitigate the "information overload" phenomenon. Secondly, a mechanism that identifies the high quality answers will increase awareness within the community and will help to put more effort into questions that remain poorly answered. For instance, in Stackoverflow[1] alone, as of September 2013, we found that approximately 33% of the questions have yet to be marked as resolved (i.e., out of the 5 million, 1.7 million questions have no answer marked as "accepted"). More generally, the study of the characteristics of answers is expected to improve our understanding of information seeking activities and social media reception in general.

Typically, CQAs adopt a simple model where the discussion is centred around a question posted by a user with answers addressing it submitted by community members. A question remains "unresolved" until the questioner marks exactly one of the answers as the "accepted" one. Research so far has indicated that communities cannot be examined statically. In particular, the dynamic nature of online communication and communities alters the distribution of different roles in a community and may affect its sustainability [15]. In this work we also discuss how the content/linguistic features change over time and the implications this change has for the community's perception of good content quality.

The study of publicly available corpora and the continuously increasing volume of user-generated content through social media is at the focus of web science. Researchers in related fields have used lexical, syntactic, and discourse features to produce a predictive model of readers' judgments [14]. In several cases, the use of shallow features, i.e. features that do not employ semantic or syntactic parsing such as sentence length [8] or word length [13], are proven to be effective in assessing properties such as ease of reading or usefulness. However, with respect to CQA, research efforts towards the exploitation of shallow features report

[1] `http://stackoverflow.com/`

relatively low results (e.g., Burel et al. report 70% precision [5] and Tian et al. report 71% prediction accuracy [17] for a balanced dataset). To improve the efficacy of their models, researchers refer to more contextual information, such as the *score* of each answer, the *comments* received or the *reputation* of the user.

Solutions that are based on *answer* or *user ratings* have been shown to be far more effective compared to linguistic ones. For instance Burel et al. [5] achieve 85% precision largely due to the received score (answer rating), while Anderson et al. [3] find that authors with a high reputation are behind good quality answers (user rating). At the same time, there is growing research interest around sites like StackExchange that employ badges and how this may affect the development of a community and the acceptance of answers. There is particular interest in studying well known behaviours, such as preferential attachment (the "rich get richer" effect), which may be a side-effect of systems that support community-based content assessment [11]. In such cases preferential attachment poses a threat to the development of the community, since the reputation framework reinforces the pre-existing community hierarchy.

In addition to the above concerns around the utilisation of reputation-based platforms, another issue pertains to the usage of answers' ratings, since these cannot be applied in a real-time setting due to an inherent delay between the answerer's submission and the expected community feedback. To provide a solution that is applicable in a real-time setting we address the problem of best-answer identification in CQAs by leveraging purely textual features of the candidate answers. Our decision to ignore further contextual information is based on the fact that when examining a question and its candidate answers we do not always have at our disposal information such as answer ratings or the reputation information for new users.

The main goal of our work is to address the problem of best answer identification and prediction using solely textual features. To do so, we examine 21 of the most active StackExchange websites, including the most popular one, Stackoverflow. We study the evolution of language characteristics over time and across different communities. We investigate the distinct properties of accepted answers and we devise a classification strategy to achieve this prediction efficiently. Our paper makes the following contributions:

- We introduce a novel way of exploiting various shallow textual features with state-of-the-art performance that outperforms previous linguistics-based solutions
- We evaluate and validate the results of the proposed technique on 21 StackExchange (SE) websites. To our knowledge, the scope and diversity of this evaluation is the largest so far.
- We show how our solution is generically applicable without the use of training data from the target SE website.

The remainder of the paper is organised as follows: Section 2 reviews related work. Section 3 presents information around StackExchange and the corresponding dataset that we used. Section 4 introduces the features that we used for addressing our problem, including the proposed, novel methodology for devising discretised linguistic features. We then proceed to Section 5 where we present the results of our evaluation. Finally, Section 6 discusses how our approach compares to others as well as some ideas for future work.

2. RELATED WORK

The past years have seen the publication of several papers addressing the quality of answers in CQA. We first discuss work on best answer identification for StackExchange (SE) and Yahoo! Answers[2] (YA) and then move on to work on quality assessment of answers.

The most recent work on SE comes from Burel et al. [5]. The authors introduce three different classes of features for predicting the best answers. These classes contain features involving the content, user and thread information of answers. The combination of these features yields a precision of 85% for the case of two StackExchange websites (Server Fault and Cooking). The results show that the model deployed is mostly based on the "Score Ratio" feature (the proportion of scores given to a post from all the scores received in a question thread). According to our approach, this feature constitutes part of "future knowledge", as the score value cannot be collected near the submission time of an answer and is therefore against our initial input assumption. Furthermore, when using purely textual features, the authors report a precision drop for Server Fault[3] down to 65%. We show how the textual features can be leveraged to improve performance.

Tian et al. [17] share similar objectives with this work as they focus solely on the content of posts rather than user background information (e.g., user rating). They identify contextual information as the most important factor for successfully predicting the best answer. More specifically, they develop their model by using the questions together with the corresponding answers. However, some of the attributes used include comments, which are disregarded in our approach as they constitute future knowledge. This requirement for the existence of information such as the comments is the reason why the dataset they used included only around 196k answers from Stackoverflow that were at least a year old. The final prediction accuracy reported in this case was 72%. Our solution overcomes this limitation for the need for long-lived questions and answers and exhibits higher performance.

In general, YA adopts similar operation mechanics but differs in the nature of questions submitted by the users, since questions are more debatable, subjective and are hosted on a single website divided into different thematic categories. Shah and Pomerantz [16] construct a dataset of resolved questions each one containing exactly 5 answers (the ratio of answers is 4:1). The model employed contains a number of shallow textual features, such as the length of the subject and content for each answer, as well as information about a user profile and the score received. The authors start by acknowledging that the baseline of the constructed dataset has an accuracy of 80% (i.e. negative classifier classifying all answers as non-accepted) and manage to improve the classification up to 84.52%. The authors also report a lower performance when employing readability annotations from

[2] http://answers.yahoo.com/

[3] http://serverfault.com

Mechanical Turk[4] due to the inherent subjectivity of the assessments. This is an important finding that demonstrates the subjectivity and difficulty inherent in best answer identification. Finally, Adamic et al. [1] also focus on YA and introduce a number of thread and content features. Looking at questions under the "Programming" category, they report a precision of 72.9% using features such as thread length, user number of best answers and user number of replies.

Work more broadly related to ours includes papers that study the activity of questions in StackExchange, such as whether a question will receive any answer (Yang et al. [18]), or whether questions have been answered sufficiently (Anderson et al. [3]). Yang et al. [18] use the question length as a linguistic feature in addition to 6 more features pertaining to the asker's background and they experiment with different classification algorithms. The highest reported F-Measure is 0.325. Anderson et al. [3] use several features to assess the longevity of a question and highlight the importance of the number of answers, the sum of scores on answers to question, as well as the length of the highest-scoring answer. Liu et al. [12] present a framework for estimating question difficulty. The authors follow a competition-based approach which models together the level of question difficulty with the level of user expertise.

Finally, numerous papers have been published that focus on the assessment of user-generated content quality. Agichtein et al. [2] use human editors to train a classifier for high and low quality questions and answers in YA. They use different features including baseline linguistic features such as word n-grams and report 67% precision (0.805 AUC) for an unbalanced dataset comprised of a few thousand answers. Furthermore, their study reports that the length of an answer is a significant indicator of answer quality.

3. STACKEXCHANGE DATASET

StackExchange (SE) is the engine that powers some of the most popular CQAs such as Stackoverflow (SO), Mathematics and Server Fault. Webpages in SE consist of one question and an arbitrary number of answers submitted by users. As of February 2014, 115 SE websites are available, each focussing on one topic. Topics are diverse, ranging from programming, system and network administrating to cooking, scientific skepticism and English language. As indicated in the mission statement, SE "is all about getting answers, it's not a discussion forum, there's no chit-chat". In order to maintain the quality of both questions and answers, posts are curated by the members of the community and if a question or an answer is deemed to be inappropriate or irrelevant, the post is removed from the website. In addition to the above, the reputation system introduced incentivises users to receive accreditation from the community and create high quality content, which is rewarded through badges and extra rights (such as the right of content removal). The high quality of the content has lead SE's premier website, Stackoverflow (SO), to grow vigorously and attract almost 3 million users in approximately 5 years[5]. In total, as of February 2014, SE websites host 4.8 million users, 8.3 million questions and 14.7 million answers.

The full content – except users' personal information – of SE is distributed under a Creative Commons licence. For our work, we downloaded the dump of September 2013[6]. In addition to SO, our focus is on 20 of the biggest SE websites (in terms of generated content size). The total number of answers in our dataset is over 12 million and the number of questions is almost 7 million. For the purposes of the evaluation study, we excluded content created by users that had their account removed or deleted. Furthermore, for evaluating the performance of our model classifier, we only kept questions with an accepted answer. The resulting dataset contains more than 8 million answers and almost 4 million questions (see Table 1 for an overview).

Table 1: Overview of the StackExchange websites dataset. Columns refer to the number of accepted (A), non-accepted (NA) and total number of answers (Total).

SE Website	A	NA	Total
stackoverflow.com	3,375,817	3,795,276	7,171,093
apple$^{\text{se.com}}$	14,471	14,149	28,620
askubuntu.com	37,907	33,746	71,653
drupal$^{\text{se.com}}$	14,393	8,558	22,951
electronics$^{\text{se.com}}$	11,726	14,942	26,668
english$^{\text{se.com}}$	17,369	31,617	48,986
gamedev$^{\text{se.com}}$	9,866	11,106	20,972
gaming$^{\text{se.com}}$	24,019	20,457	44,476
gis$^{\text{se.com}}$	10,015	8,724	18,739
math$^{\text{se.com}}$	98,351	78,294	176,645
mathoverflow.net	21,447	23,660	45,107
meta.stackoverflow.com	27,682	26,060	53,742
physics$^{\text{se.com}}$	10,851	10,389	21,240
programmers$^{\text{se.com}}$	15,998	52,694	68,692
serverfault.com	82,315	89,833	172,148
skeptics$^{\text{se.com}}$	2,041	1,421	3,462
stats$^{\text{se.com}}$	9,360	7,297	16,657
superuser.com	89,251	91,247	180,498
tex$^{\text{se.com}}$	30,642	20,249	50,891
unix$^{\text{se.com}}$	16,283	16,155	32,438
wordpress$^{\text{se.com}}$	19,420	10,788	30,208
Total	3,939,224	4,366,662	8,305,886

$^{\text{se.com}}$ .stackexchange.com

4. FEATURES FOR BEST ANSWER PREDICTION

In this section we present the features used for training and evaluating our classifier. We initially present some shallow text features and one simple vocabulary, lexical-based feature. We then proceed by showing how we propose to exploit our features more efficiently. In order to assess the performance of the proposed model more holistically, we have also added a number of features referring to the rating of answers and users.

4.1 Linguistic features

The term "shallow features" refers to those used by traditional *readability* metrics [8] which have been used for several decades. The original purpose of these metrics was to estimate the average number of years of education required for being able to read and understand written text. The measurements use "surface", aggregated values of text properties, such as the average word length, the average number

[4] https://www.mturk.com/

[5] http://stackexchange.com/sites

[6] http://www.clearbits.net/torrents/2155-sept-2013. The SE dump is now available from the Internet Archive https://archive.org/details/stackexchange.

of words in sentences or the number of sentences in a paragraph. In addition to being simple to understand, these features are computationally cheap compared to other more language-sensitive and context-sensitive features. More specifically, readability metrics are defined through a formula (based on regression analysis) which returns the expected number of years of education. Our metrics originate from similar yet more recent approaches. More specifically, we adopt as our baseline the features in Pitler and Nenkova [14], employed in the context of modelling readability judgements for the Wall Street Journal corpus, in terms of how well the articles are written. These features are the *average number of characters per word, average number of words per sentence, number of words in the longest sentence and answer length* (number of characters).

In addition to the above, we also considered using simple vocabulary features. Vocabulary features, compared to syntactic or discourse features, are cheap in terms of deployment (language-agnostic) as well as cost (linear time and space) and have been proven useful for content assessment [6, 14]. Other studies have examined how the language of a community evolves and affects the language use of individual members. Danescu et al. [7] assessed the evolution of lexical corpora within an online community and use this change to predict a member's lifecycle. To this effect we used a probability-based vocabulary feature from [14] which is constructed from a unigram language model, where the probability of an answer is defined as:

$$\prod_w P(w|M)^{C(w)}$$

$P(w|M)$ is the probability of word w according to a background corpus M, and $C(w)$ is the number of times w appears in the answer. In our case, the background corpus is built from the content of each SE website separately.

The log likelihood (noted as LL from now on) of an answer is then:

$$\sum_w C(w)log(P(w|M))$$

Finally, in order to avoid any bias in favour of short answers, we normalise LL by dividing it over the number of unique words in the answer. Hence, this feature measures the probability of the answer being close to the vocabulary used by the SE community: the closer this value is to 0, the closer the answer is to the "community vocabulary".

Figure 1 shows the average feature values for the accepted answers together with the non-accepted ones of SO using a one-month window time frame[7]. As seen from the figure, the linguistic features manage to clearly differentiate the accepted from the non-accepted answers. More specifically, accepted answers tend to be longer, use a less common vocabulary, contain longer words, more words per sentence and the longest sentences are lengthier. Even though the above remarks look promising concerning best answer prediction, when training a binary classifier *precision* remains weak (58% on average for all SE websites). Since the results that we obtained for a classification based on shallow features are comparable to similar approaches (e.g. [5, 17]) these results will constitute our baseline for evaluating the proposed solution.

A more thorough investigation towards the explanation of this poor performance leads us to identify two main issues. Firstly, as illustrated in Figure 1, the characteristics of language evolve over time; in most SE websites users follow a more eloquent language (perhaps because of the increasing complexity of questions or because of what is considered good practice and is rewarded accordingly). For example, the SE website on English language shows that around early 2012 the average length of accepted answers is lower than the average length of non-accepted answers one year later. Hence, even though there is a steady gap between the values of accepted and non-accepted answers, the rapid change in the *absolute values* of the adopted shallow features is responsible for the poor classification.

We experimented with using a sliding window and examining the features in a narrow time frame (e.g., one month, as used for Figure 1). However, the large inherent *diversity* of the posts persists together with a large variance in values. Since this is not visible in Figure 1, we discuss one example regarding the length: the average length of answers in SO during September 2008 is 482 characters with a standard deviation of 544. More specifically, for the same time period, the shortest accepted answer is only 2 characters[8] whereas the longest is around 18,000 characters. This deviation is also discussed at a later section where features are presented all together.

Finally, even if a well-performing classifier existed for a single SE website and we used the features proposed above, the same classifier would have very low performance on another SE website. Indeed, as the reader may have anticipated, the characteristics of accepted answers vary significantly across the SE websites. For instance the accepted answers in Superuser have overall average length of 577 characters, whereas the corresponding value for Skeptics SE is 2,154 characters. As already stated, our paper aims at developing a best answer prediction model independent of the community website.

4.2 Feature discretisation

In order to overcome the above weaknesses and effectively make use of the linguistic features introduced, our approach is to treat the collection of answers for *each question* as an *information unit* which can improve the training process. Instead of treating each answer independently of the other answers it is competing with, our approach is to assess the value of the features of each answer *in relation* to each other. We introduce a new set of features that stem from the linguistic features used so far: instead of dealing with continuous values, these new features are the result of *grouping*, *sorting*, and *discretisation*.

We will present an example for the *Length* feature. Let us consider the example of Table 2 where for one question there are two candidate answers (i.e., question with Id 5 having answers with Id 6 and 7). We have already shown in Section 4.1 that the longer an answer is, the more likely it is to be accepted. In order to represent this preference, we group all answers by their corresponding questions (*grouping*). For each group, we then sort the answers in descending order

[7] Similar behaviour is identified for all SE websites and is omitted due to space limitations.

[8] "No" is the best answer to the question "Is there any difference between "string" and 'string' in Python?" `http://stackoverflow.com/questions/143714`

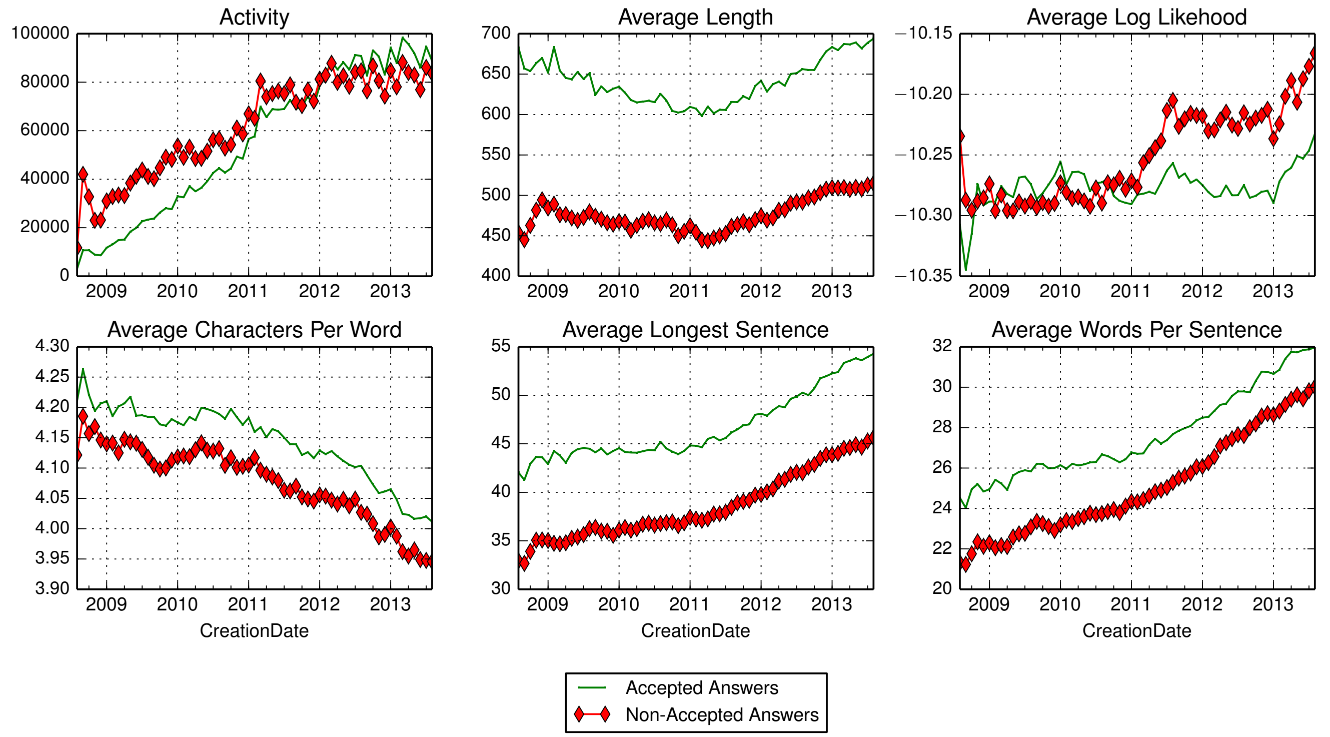

Figure 1: Activity and values of the linguistic features (y-axis) for the Stackoverflow dataset over time (x-axis). Top left sub-plot shows the number of answers posted every month. The remaining sub-plots show the average values for the accepted and non-accepted answers.

Table 2: Example of feature discretisation for the case of $Length$**, 5 submitted answers and 2 questions. Column Question Id refers to the question under which the answer is submitted.**

Question Id	Answer Id	*Length*	$Length_D$
1	2	200	2
	3	150	3
	4	250	1
5	6	250	1
	7	200	2

(*sorting*) and assign a rank for each answer, starting from 1 and incrementing this rank by 1 (*discretisation*). Thus, the answer with the longest $Length$ will receive $Length_D$ of value 1 (answer Id 6 with length 250) while the answer that comes second a value of 2 (answer Id 7 with length 200 - note that we are representing the discretised form of each $feature$ as $feature_D$). The result of this process is the introduction of an equal number of linguistic features without the usage of any further information (apart from the necessary association of a question and its corresponding answers[9]).

As a result of the discretisation process on all of our shallow features, the information added and used for training purposes improved significantly. This is manifested by the information gain (about 20 times higher) and which we present in the following subsection. Additionally, the benefits of this discretisation are discussed thoroughly in Section 5, where we present the classification results. It may appear that our discretisation process is dependent on “future knowledge”, since discretised values may alter as more answers are submitted. Our method is no more time dependent than the notion of a best answer is, as it allows for best answer prediction at any point, in a real-time setting, which is not possible when relying on answer ratings. As more answers are entered, the discretised values change and a new current best answer can be derived.

In the following subsection we will discuss the inclusion in our classifier of two popular non-linguistic features, to allow us a more thorough evaluation.

4.3 User and Answer Rating Features

Until now we have discussed the linguistic features and how the proposed discretisation process is expected to yield better results. In order to have a more complete view of the performance of our classifier we have integrated some non-linguistic features. It is worth noting that these are included for evaluation purposes only; they do not form part of our approach. We group these features into different sets, following the discussion in Section 1. The first set of features (*user*) describes *past* or background knowledge and more specifically the *user profile*, such as the *reputation*, the number of *profile views*, number of *up*- and *down-votes* and the $UserUpDownVotes$ feature, which we define as the difference over the sum of Up and $Down$ votes, as follows:

[9] Note that other approaches typically omit this information.

Table 3: Summary of features used. The last column indicates the improvement on the information gain for the features that have been re-used as discretised. Values are for averages for all SE websites.

Category	Name	Information Gain
Linguistic	$Length$	0.0226
	$LongestSentence$	0.0121
	LL	0.0053
	$WordsPerSentence$	0.0048
	$CharactersPerWord$	0.0052
Linguistic Discretisation	$Length_D$	0.2168
	$LongestSentence_D$	0.1750
	LL_D	0.1180
	$WordsPerSentence_D$	0.1404
	$CharactersPerWord_D$	0.1162
Other	Age	0.0539
	$CreationDate_D$	0.1575
	$AnswerCount$	0.3270
User Rating	$UserReputation$	0.0836
	$UserUpVotes$	0.0535
	$UserDownVotes$	0.0412
	$UserViews$	0.0528
	$UserUpDownVotes$	0.0508
Answer rating	$Score$	0.0792
	$CommentCount$	0.0286
	$ScoreRatio$	0.4539

$$UserUpDownVotes = \frac{|UserUpVotes| - |UserDownVotes|}{|UserUpVotes| + |UserDownVotes|}$$

The second set of features (entitled as *Answer rating*), includes information concerning the community feedback on answers, such as the number of *comments*, the *score* and the *score ratio* ("the proportion of scores given to a post from all the scores received in a question thread", as indicated by Burel et al. as the most informative feature [5]). Finally, another set of features (*Other*) was used, such as the *AnswerCount*, the *Age* (real number representing days) of answers and the corresponding *CreationDate*$_D$ (answer speed is linked to good answer quality [3]). The total number of features is 21 and are shown in Table 3.

Table 3 shows the values for each feature in addition to their corresponding information gain. Information gain is a measurement based on entropy used for machine learning and has been employed in classification tasks to identify important features. Information gain $InfoGain$ of an attribute A for class C is defined using the entropy H measurement as follows:

$$InfoGain(C, A) = H(C) - H(C|A)$$

We can clearly see that the task of discretisation improves the information gain for all features. In particular, the information gain for *linguistic* features has increased on average 20 times. For the case of *Length*, the improvement is so significant that it manages to outperform well-known features, such as all those based on User Rating, and to rank as the third most important feature. At the same time, both *Length*$_D$ and *LongestSentence*$_D$ carry more information gain than *CreationDate*$_D$ which is also a popular feature shown to yield good performance.

5. EVALUATION: BEST ANSWER PREDICTION

Having experimented with a number of different classifiers, our evaluation shows that we obtain the best results by using *Alternate Decision Trees* (ADT) [9]. Even though we received good results with different classifiers available in Weka [10], we attribute the high performance of ADTs to the fact that they constitute a well-known binary, boosting classifier for numerical data, which suits our goals. Our evaluation was conducted using 10-fold cross-validation. In order to verify the performance of the proposed solution we conducted different experiments, each one aiming at validating the characteristics of the proposed solution.

5.1 Prediction

Table 4 presents the first results concerning the performance of our classifier without the inclusion of features based on answer or user ratings. The table shows that the macro averaged (unweighted) precision using *linguistic* and *other* (namely *Age*, *CreationDate*$_D$ and *AnswerCount*) features with *discretisation* is *84*%. The remaining evaluation metrics (recall, F-Measure) maintain high values resulting in an average AUC of *0.87.* The website with the lowest precision is Programmers SE with 76%, which can be attributed to the fact that the dataset for this website is heavily imbalanced (only 23% of the dataset's answers are accepted – see Table 1). On the contrary, Skeptics SE has 87% precision with 0.91 AUC value, which can be explained as follows: Firstly 58% of the answers in the dataset are accepted (the third highest ratio from all SE websites). The second reason stems from the website topic and the type of discourse that takes place: questions in Skeptics SE mainly attract scientific reasoning without much technical information, hence prose and linguistic features play a more important role. This performance is also confirmed by the value of information gain for the discretised version of *Length*, which is 0.27 (Skeptics) whereas the average value for *Length*$_D$ is 0.22 (see Table 3). The English SE dataset is also imbalanced (only 35% of the answers are accepted, close to programmers SE), but language-based features manage to overcome this challenge, most likely due to the nature of the discourse (i.e. similar to skeptics SE). The resulting prediction has 77% precision and 0.83 AUC.

5.2 Improvement due to discretisation

We have already shown the improvement in information gain after discretising the linguistic features (see Table 3). Here we aim to analyse the benefits of this process in the task of best answer prediction. To do so, we compare the performance of our classifier to other classifiers that use more sets of features, including features produced from ratings. Our goal in performing this comparison is to examine the information loss when choosing to disregard information coming from ratings.

Table 5 presents the results when using different sets of features and 10-fold validation. The table contains the average values for all SE websites as the output of different evaluations. Initially, we use the absolute values of textual features (also mentioned in Section 4) with low results 58% (Case 1). The second and third Cases both utilise the discretised features, while the third is additionally using the *other* set of features. Cases 2 and 3 constitute our proposed prediction method (Case 3 was presented in detail in subsection 5.1

Table 4: Results for best answer prediction using *linguistic* and *other* features with discretisation. Columns show macro averaged precision (P), recall (R), F-measure (FM) and Area-Under-Curve (AUC) using 10-fold validation.

SE Website	P	R	FM	AUC
stackoverflow.com	0.82	0.66	0.73	0.85
apple.stackexchange.com	0.84	0.68	0.75	0.86
askubuntu.com	0.84	0.74	0.79	0.88
drupal.stackexchange.com	0.87	0.79	0.83	0.89
electronics.stackexchange.com	0.79	0.65	0.71	0.84
english.stackexchange.com	0.77	0.52	0.62	0.83
gamedev.stackexchange.com	0.82	0.71	0.76	0.87
gaming.stackexchange.com	0.87	0.79	0.83	0.91
gis.stackexchange.com	0.85	0.73	0.78	0.87
math.stackexchange.com	0.85	0.74	0.79	0.87
mathoverflow.net	0.83	0.70	0.76	0.87
meta.stackoverflow.com	0.87	0.69	0.77	0.87
physics.stackexchange.com	0.86	0.71	0.78	0.88
programmers.stackexchange.com	0.76	0.40	0.52	0.84
serverfault.com	0.83	0.66	0.74	0.85
skeptics.stackexchange.com	0.87	0.83	0.85	0.91
stats.stackexchange.com	0.85	0.79	0.82	0.89
superuser.com	0.84	0.65	0.73	0.85
tex.stackexchange.com	0.87	0.77	0.82	0.88
unix.stackexchange.com	0.81	0.68	0.74	0.85
wordpress.stackexchange.com	0.88	0.80	0.84	0.89
Average	0.84	0.70	0.76	0.87

and Table 4). Furthermore Case 4 refers to a “traditional” approach that relies in plain linguistics *and* user ratings. We can see that while a whole new set of features is added into the dataset, the performance of classification remains lower than Case 3, which is linguistics-based. Case 5 keeps the user ratings in addition to incorporating all features of Case 3. Hence, classification accuracy is the highest compared to all previous classifications, but almost identical to Case 3 which is strictly based on content and discretisation (lower precision 82% vs. 84%, higher AUC 0.88 vs. 0.87). Finally, Case 6 uses all features presented in Table 3, including the *answer ratings*. This set of features uses all features but most importantly user-entered scores and manages to outperform all of the previous cases. Case 6 shows that the information contained within answer ratings is independent – to a certain extent – of the information found in previous features.

In summary, results in Table 5 show that the discretisation of linguistic features manages to outperform significantly the classifier based on linguistic features only. Moreover, we can also see that user rating features such as reputation do not improve our classification, a sign that discretisation is a process that extracts very useful information and delivers very strong results. Figure 2 shows the AUC curves for Stackoverflow for all 6 cases and confirms the above remarks.

5.3 Generality

The final part of our evaluation aims to examine whether our solution is generic enough to be applied without the need to train our classifier on data from a new website. If the answer to this question is positive, we can assume that our classifier is generic enough to be applied to almost any SE website and to a large extent contains cross-domain intuitions about the mechanics of best answer identification. In order to have a positive answer to our research question,

Table 5: Results for best answer prediction using different sets of features (Cases 1 to 6) for all SE websites. Columns show macro average precision (P), recall (R), F-Measure (FM) and Area-Under-Curve (AUC) for all 21 SE websites using 10-fold validation. Case 3 was presented in detail in Table 4.

No.	Features Used	P	R	FM	AUC
1	Linguistic	0.58	0.60	0.56	0.60
2	Linguistic & Discretisation	0.81	0.70	0.74	0.84
3	Linguistic & Discretisation & Other	0.84	0.70	0.76	0.87
4	Linguistic & Other & User Rating (no discretisation)	0.82	0.69	0.75	0.86
5	Linguistic & Other & User Rating (with discretisation)	0.82	0.72	0.77	0.88
6	All features (Answer and User Rating with discretisation)	0.88	0.85	0.86	0.94

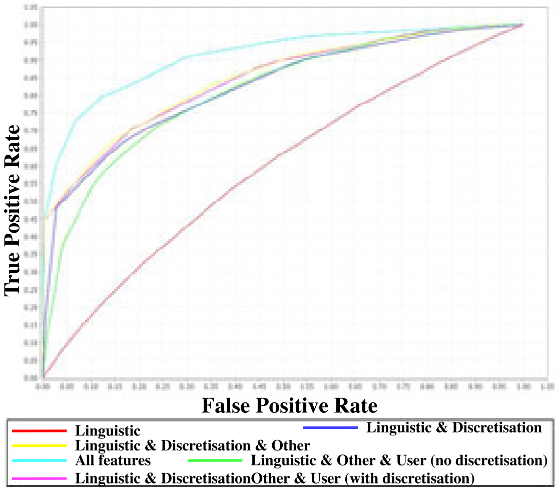

Figure 2: AUC for Stackoverflow. Different curves show the results for 10-fold cross validation using different sets of features (Cases 1 to 6). The 4 overlapping curves in the middle show that the discretisation of features outperforms the linguistic-based approach (bottom curve), matches the classification based on reputation and approaches the classification using all features (top curve) including user and answer ratings.

two requirements must be satisfied. Firstly, our classifier should be able to describe the characteristics of the best answers accurately for each SE website (robustness). Secondly, the features used in this model must neutralise the special characteristics of each SE website (generality). To examine the above hypothesis, we created new datasets following a leave-one-out strategy for each SE website. For instance, for the case of English language SE, we merge the remaining 19 SE websites[10] into one training dataset and use English language as the test dataset. For the evaluation purposes we applied classification using the features of Case 3.

The results of the evaluation shows that the average values for our evaluation metrics remain intact. More specifically, average precision fell by 1%, while recall, F-Measure and AUC remained the same (see Case 3, Table 4 for the values). Hence, we can claim that our classifier manages to remain effective without requiring access to the specific knowledge of the SE website. We believe that this result strengthens the value of discretisation even further. Despite the inherent variance in shallow feature values across answers and – even more – across SE websites, the discretisation process is able to demonstrate both robustness and generality.

6. DISCUSSION

Here we review and discuss our results in relation to previous related work and also discuss some issues raised as a result of the proposed methodology and potential extensions of this work.

6.1 Comparison

As already discussed in Section 2, the paper by Burel et al. [5] predicts accepted answers for Server Fault and Cooking SE. Our work did not include Cooking SE, but we include the larger, more up-to-date dataset of Server Fault (95k vs. 172k answers). Burel et al's classifier based on content delivers a precision of 64.7%, 0.628 F-Measure and 0.679 AUC. Our methodology which employs discretisation of linguistic features outperforms their work by 18-21%, since for Server Fault our precision is 83%, F-Measure is 0.74, AUC is 0.85 (Case 3) and 86% precision, 0.69 F-Measure and 0.83 AUC (Case 2). Moreover, our results when they consider contextual features such as user and answer ratings are similar to ours achieving the same F-Measure 0.84, our precision and AUC being at 89% (5% higher) and 0.93 (0.02 higher) respectively.

Similarly to us Tian et al. [17], look at the content of answers. However, they also exploit features related to what we refer to as answer ratings, since they also consider the number of comments to each answer, a feature which is reported as amongst the most informative ones. The authors report a prediction accuracy of 72.27% on a SO dataset of 196k answers at least one year old. By comparison our SO dataset contains 7.1 million answers and our classier returns 82% precision, 0.73 F-Measure and 77% prediction accuracy, which constitutes a noticeable increase in performance.

While the work concerning YA cannot be compared directly to ours, we highlight some analogies and discuss the results. For instance, Shah and Pomerantz [16] constructed a negative classifier with a dataset comprised of a 1:4 ratio of accepted to non-accepted answers. Adamic et al. [1] consider Programming questions submitted in YA and – similarly to us – disregard the ratings of answers and users. The authors report 72.9% precision, which is similar to our linguistics-based findings. Hence, we can assume that our classifier may be able to increase performance also in the case of YA.

6.2 Future work

To our knowledge, the proposed technique of dealing with continuous and multi-dimensional data found in shallow features constitutes a novel approach for assessing user-generated content. We intend to explore this direction further and apply it on other cases of social media, to fully examine the effectiveness of this technique. For example, one direction would be to analyse the linguistic characteristics of different roles in online communities, such as initiators, conversationalists, etc. (see for example [4]). Another possibility is to follow up on the work conducted by Anderson et al. [3] and explore the assortativity between user reputation and linguistic characteristics of user input.

6.3 Conclusions

Previous research on best answer prediction has shown that linguistics-based features can be helpful to a limited extent. The relevant literature shows that features based on user reputation and answer ratings manage to boost the performance of classifiers and outperform purely content-based approaches. Our approach adopts a novel way of processing linguistic features and manages to bridge the above gap. To do so, instead of processing all answers as one solid training dataset, the proposed discretisation process manages to highlight the distinct characteristics of each answer compared to its candidate, "competing" answers. The information that is produced from this process dramatically improves the performance of our classifier. Our extensive evaluation shows that shallow features, such as length and longest sentence, can be very informative, contradicting the findings of earlier work. Hence, encoding this information into a discretised form allows us to train a classifier that is effective enough to match other classifiers which do use and depend upon non-linguistic contextual information.

Our evaluation shows that the performance of our proposed approach matches the performance of reputation-based classification. Contrary to our intuition, the inclusion of more information such as user background information does not improve the classification, a sign that reputation information is not independent of information found in linguistic features. Finally, our classification methodology is generic and can be applied to the rest of the SE websites, without the need for training data from the target website. Shallow features, such as answer length and longest sentence can be used effectively for assessing user-generated text, following our methodology.

Acknowledgments

This work was supported by the BlogForever (No.269963), COMPOSE (No.317862) and CARRE (No.611140) projects funded by the European Commission Framework Programme 7.

[10] We chose to exclude Stackoverflow from training due to its large size which would slow the training process dramatically.

7. REFERENCES

[1] L. A. Adamic, J. Zhang, E. Bakshy, and M. S. Ackerman. Knowledge sharing and yahoo answers: everyone knows something. In *Proceedings of the 17th international conference on World Wide Web*, pages 665–674. ACM, 2008.

[2] E. Agichtein, C. Castillo, D. Donato, A. Gionis, and G. Mishne. Finding high-quality content in social media. In *Proceedings of the 2008 International Conference on Web Search and Data Mining*, pages 183–194. ACM, 2008.

[3] A. Anderson, D. Huttenlocher, J. Kleinberg, and J. Leskovec. Discovering value from community activity on focused question answering sites: a case study of stack overflow. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 850–858. ACM, 2012.

[4] S. Angeletou, M. Rowe, and H. Alani. Modelling and analysis of user behaviour in online communities. In *The Semantic Web–ISWC 2011*, pages 35–50. Springer, 2011.

[5] G. Burel, Y. He, and H. Alani. Automatic identification of best answers in online enquiry communities. In *The Semantic Web: Research and Applications*, pages 514–529. Springer, 2012.

[6] J. Callan and M. Eskenazi. Combining lexical and grammatical features to improve readability measures for first and second language texts. In *Proceedings of NAACL HLT*, pages 460–467, 2007.

[7] C. Danescu-Niculescu-Mizil, R. West, D. Jurafsky, J. Leskovec, and C. Potts. No country for old members: User lifecycle and linguistic change in online communities. In *Proceedings of the 22nd international conference on World Wide Web*, pages 307–318. International World Wide Web Conferences Steering Committee, 2013.

[8] L. Feng, M. Jansche, M. Huenerfauth, and N. Elhadad. A comparison of features for automatic readability assessment. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 276–284. Association for Computational Linguistics, 2010.

[9] Y. Freund and L. Mason. The alternating decision tree learning algorithm. In *ICML*, volume 99, pages 124–133, 1999.

[10] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The weka data mining software: an update. *ACM SIGKDD Explorations Newsletter*, 11(1):10–18, 2009.

[11] J. Jones and N. Altadonna. We don't need no stinkin'badges: examining the social role of badges in the huffington post. In *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work*, pages 249–252. ACM, 2012.

[12] J. Liu, Q. Wang, C.-Y. Lin, and H.-W. Hon. Question difficulty estimation in community question answering services. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 85–90, 2013.

[13] S. T. Piantadosi, H. Tily, and E. Gibson. Word lengths are optimized for efficient communication. *Proceedings of the National Academy of Sciences*, 108(9):3526–3529, 2011.

[14] E. Pitler and A. Nenkova. Revisiting readability: A unified framework for predicting text quality. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 186–195. Association for Computational Linguistics, 2008.

[15] M. Rowe, M. Fernandez, S. Angeletou, and H. Alani. Ontology paper: Community analysis through semantic rules and role composition derivation. *Web Semantics: Science, Services and Agents on the World Wide Web*, 18(1):31–47, 2013.

[16] C. Shah and J. Pomerantz. Evaluating and Predicting Answer Quality in Community QA. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 411–418. ACM, 2010.

[17] Q. Tian, P. Zhang, and B. Li. Towards predicting the best answers in community-based question-answering services. In *Seventh International AAAI Conference on Weblogs and Social Media*, 2013.

[18] L. Yang, S. Bao, Q. Lin, X. Wu, D. Han, Z. Su, and Y. Yu. Analyzing and predicting not-answered questions in community-based question answering services. In *AAAI*, 2011.