information

MDPI

Article

A Study on Influential Features for Predicting Best Answers in Community Question-Answering Forums

Valeria Zoratto [1], Daniela Godoy [2,3,*] and Gabriela N. Aranda [1]

[1] Facultad de Informática, Universidad Nacional del Comahue (UNComa), Neuquén CP 8300, Argentina; vzoratto@fi.uncoma.edu.ar (V.Z.); gabriela.aranda@fi.uncoma.edu.ar (G.N.A.)

[2] Facultad de Ciencias Exactas, Universidad Nacional del Centro de la Provincia de Buenos Aires (UNCPBA), ISISTAN, Tandil CP 7000, Buenos Aires, Argentina

[3] Instituto Superior de Ingeniería de Software Tandil, ISISTAN, UNCPBA-CONICET, Tandil CP 7000, Buenos Aires, Argentina

* Correspondence: daniela.godoy@isistan.unicen.edu.ar

Abstract: The knowledge provided by user communities in question-answering (QA) forums is a highly valuable source of information for satisfying user information needs. However, finding the best answer for a posted question can be challenging. User-generated content in forums can be of unequal quality given the free nature of natural language and the varied levels of user expertise. Answers to a question posted in a forum are compiled in a discussion thread, concentrating also posterior activity such as comments and votes. There are usually multiple reasons why an answer successfully fulfills a certain information need and gets accepted as the best answer among a (possibly) high number of answers. In this work, we study the influence that different aspects of answers have on the prediction of the best answers in a QA forum. We collected the discussion threads of a real-world forum concerning computer programming, and we evaluated different features for representing the answers and the context in which they appear in a thread. Multiple classification models were used to compare the performance of the different features, finding that readability is one of the most important factors for detecting the best answers. The goal of this study is to shed some light on the reasons why answers are more likely to receive more votes and be selected as the best answer for a posted question. Such knowledge enables users to enhance their answers which leads, in turn, to an improvement in the overall quality of the content produced in a platform.

Keywords: CQA forums; best answer prediction; information retrieval

check for updates

Citation: Zoratto, V.; Godoy, D.; Aranda, G.N. A Study on Influential Features for Predicting Best Answers in Community Question-Answering Forums. *Information* **2023**, *14*, 496. https://doi.org/10.3390/info14090496

Academic Editor: Anselmo Peñas

Received: 11 July 2023
Revised: 22 August 2023
Accepted: 25 August 2023
Published: 7 September 2023

1. Introduction

Community Question-Answering (CQA) platforms, as well as specific domain discussion forums, offer users the possibility of publishing, searching and sharing knowledge within interested communities. In CQA services, users express their information needs by submitting natural language questions with the goal of retrieving answers posted by other users in the community. In contrast to the results of general-purpose web search engines, using keyword queries in CQA websites leads to more specific answers, as the persons responding are more likely to accurately interpret the nature of the question. Websites like *Yahoo! Answers* (https://www.answers.com/) or *Quora* (https://quora.com), as well as domain-specific CQA websites like *StackOverflow* (http://www.stackoverflow.com) or *Mathematics StackExchange* (https://math.stackexchange.com/), belong to a prominent group of successful and popular Web 2.0 applications used daily by millions of users to find answers to complex, subjective or context-dependent questions [1,2].

CQA services consist of three main components: (1) a mechanism for users to submit questions in natural language, (2) a place for users to submit answers to questions and (3) a community built around this exchange [3]. Many sites allow users to submit questions on any topic, such as *Yahoo!Answers*, *WikiAnswers* (https://www.answers.

com/) and *AnswerBag* (https://www.answerbag.com), whereas other sites are limited to a certain domain, for example, *StackOverflow* is restricted to programming questions, *MathOverflow* (http://mathoverflow.net) is used for math questions, *HeadHunterIQ* (http://headhunteriq.com) targets business recruiters and *HomeworkHub* (http://www.stackexchangesites.com/homework-hub) provides help with chores around the house, among other available forums.

Like many other communities, the programming community takes advantage of the information available in these groups to solve problems and learn how to perform specific programming tasks, ranging from solving bugs to reusing pieces of code. For this reason, discussion forums have become an important resource for programmers and a practical mechanism for reusing a great volume of knowledge about languages, code, debugging, common practices and a variety of technical issues.

With the increasing popularity of CQA sites, they have transformed into a vast source of information to find answers to all kinds of questions. However, to fully take advantage of this knowledge, it becomes crucial to identify the best existing answer(s) for a given query and simplify the access to relevant information for users [4]. In addition, there are numerous discussion forums on the same topic where it is possible to find similar questions and the corresponding answers. Then, users have to face the problem of finding the more adequate, high-quality answer to their current problem in multiple sites and a myriad of discussion threads.

Some researchers suggest that the context that gives rise to an information need is unique to each individual, and consequently, the response that is useful for one individual in a particular context will, at best, only be partially useful for other individuals in other contexts [3,5,6]. Therefore, to identify the factors that contribute to the success of answers becomes essential to help users to determine whether a previously provided response is likely to satisfy their information need. This can be, in turn, the base for recommender systems that suggest the best possible answers to posted questions.

Investigating the factors that contribute to the acceptance of answers can serve several purposes. It can be used to provide guidelines for users to improve their answers beforehand, increasing their chances of receiving upvotes and gaining prestige through them within the community. The outcome of this study can even be automatized in a recommender system that suggests improvements to answers according to some established guidelines; for example, it can remind the user to include an example before posting an answer, if this is a relevant feature of the best answers. In other words, knowing the aspects that need to be reinforced in advance is likely to lead to better answers. In turn, QA communities will be benefited, as their overall quality will be raised as a consequence of writing answers more thoughtfully. This study is oriented to shed some light onto the factors that lead answers to receive more upvotes and be selected as the best ones, using prediction as a means for discovering the influence of them in the answer's posterior performance.

CQA sites store textual but also nontextual information about users, such as number of questions answered, number of voted responses and other additional information. In some cases, they also award points or levels of expertise to users according to their performance in answering. Metrics regarding the quality of the answers allow CQA sites to derive user metrics, such as reputation, which in turn propagate to new answers provided by the same user and so on.

Several characteristics from the individual posts and the community itself can be related to an answer's likelihood of being selected as the best answer in a thread. From the textual content, for example, information such as an answer's readability and comprehensibility can be gleaned. From the nontextual part of an answer, certain elements in a post (e.g., the code examples), as well as how people interact with or react to posts, can also provide some insights into an answer's probability of properly satisfying an information need.

In this paper, we analyze the role of the different textual and nontextual elements that can be extracted from CQA sites and its effect on predicting the best possible answers in a discussion thread. Based on data extracted from a real question-answering site such as $StackOverflow$, we empirically evaluate the impact of a range of features describing answers to determine their importance in identifying satisfactory responses. It is worth noting that this study is not focused on improving prediction but on investigating the influence of features in detecting answers that better satisfy an information need. In addition, this study concentrates on the relationships among several features and answers selected as the best answers. Although there are possibly several high-quality answers for any question, this study is focused on those considered as the best ones, i.e., those the users posting the query indicated as the ones that solved their information need. The aim of this study is not to judge an answer quality in general terms but to identify the elements that make it the best for accomplishing the user's information-seeking goal.

This paper is organized as follows. Section 2 summarizes related research in the area. Section 3 presents the different aspects we analyzed and the rationale behind their selection to identify the best answers. Section 4 describes the data used for experimentation and the methodology employed, whereas Section 5 reports the achieved results. Discussion and conclusions are presented in Section 6.

2. Related Works

The amount of information available on the Internet is growing every day. In particular, Community Question-Answering (CQA) services have become popular places for Internet users to search for information. CQA sites, such as *Quora* or *StackExchange* (www.stackexchange.com), rely on the *wisdom of the crowd* [7], that is, the collective opinion of a group of individuals considered over the opinion of a single, possibly expert, user. Users can contribute to the community by asking questions, providing answers, voting for relevant posts and more. The possible interactions with the CQA forum varies from site to site, whereas most of them are human-moderated. Often, traditional machine learning (ML) and deep learning (DL) have been leveraged to explore the ever-growing volume of content that CQAs engender [8]. The following subsections discuss the related literature tending to exploit CQA knowledge to properly answer questions.

2.1. User Behavior and Roles

Several authors have investigated the behavior of users, their roles and their motivation for participating in QA communities. Users in these forums can be broadly divided in three groups: users who only ask questions, users who only answer questions and users who both ask and answer questions [9]. Adamic et al. [9] analyzed *Yahoo! Answers* categories and clustered them according to content characteristics and patterns of interaction among the users. Clustering categories according to thread length allowed to differentiate two broad groups: on the one hand, discussions not focused on factual answers that tend to have longer threads, broader distributions or activity levels, and the participation of users both posing and replying questions and, on the other hand, categories favoring factual questions, with shorter thread lengths and participating users with either seeking or answerer roles. In [10], the authors tried to shed some light on why people prefer not to contribute publicly on online communities, where only a small fraction of members post messages. Based on an online survey, the authors concluded that there are many reasons why people lurk on online discussion communities (not needing to post, needing to find out more about the group before participating and poor software usability, among others) and, more importantly, most lurkers are not selfish free-riders (people who take and do not give back).

Regarding the role of users in CQA, there are works focusing on the localization of experts [11] as well as, in the opposite extreme, on the identification of spammers [12]. Finding experts consist mainly in modeling the expertise of a user on a given topic based on their answering history. User profiles are created for each user based on the questions they answered, and thereby, for a new question, the best matching profiles can be chosen to look for good answers. As an example, Riahi et al. [13] proposed a segmented topic model (STM) that can discover the latent topics in a hierarchical structure for routing new questions to the right group of experts. In [14], a framework was proposed to detect the top contributors in their early stage by integrating different signals from users. These "rising star" users can help the health of the community due to their high quality and quantity of contributions.

In CQA sites, there are also spammers that pretend to ask questions with the goal of selecting answers which were published by their partners or themselves as the best answers, then deriving some reward for being good answerers. Li et al.'s [12] work is concerned with how spammers disseminate promotion campaigns in CQA platforms. A framework is designed to detect such campaigns based on only some question information and questioner profiles. A supervised learning framework [15] is also proposed to identify whether a QA pair is spam based on propagated promotion intents. Early detection prevents legitimate users from being affected by these low-quality answers.

In addition to the role played by a user in a discussion thread and as a part of a community, the reputation and trust of users becomes an important element to judge the quality of an answer according to the user that provided it. The underlying user network of forums can be exploited to glean user scores regarding these aspects. In [16], a study of *StackOverflow* where expertise and user participation are recognized and rewarded through a reputation scheme, is presented. Interestingly, they found that while the majority of questions on the site are asked by low-reputation users, on average, a high-reputation user asks more questions than a user with low reputation. They also used graph analysis methods for detecting influential and anomalous users in the underlying user interaction network and found they are effective in detecting extreme behaviors such as those of spam users. A model for online thread retrieval based on inference networks that utilize the structural properties of forum threads is proposed in [17]. In this model, a score of user authority is defined so that content provided by authoritative users is considered more important. In [18], the concept of scope was introduced to assess the authoritativeness and convincing ability of a user toward other users on social platforms. Moreover, authors of this paper define the scope of the sentiment of a user on a topic in a social network. Trust and reputation have been investigated for communities of people in several contexts; consequently, they can be applied to QA forums. In [19], the trust and reputation of a thing in a multiple IoTs scenario are investigated, and a context-aware approach to evaluate them is proposed.

2.2. Quality of CQA Sites

Because of its free and unsupervised nature, user-generated content in CQA sites is known to be of varied quality. Finding high-quality content is therefore an important issue in Community Question-Answering sites, which has been recognized and investigated in several studies. Shah et al. [20] divide CQA services into two categories, social QA provided by services such as *Yahoo! Answers* and *Wiki Answers*, and virtual reference, provided by libraries that deliver reference services within online environments. Although not directly comparable, the two categories have three core concepts in common: relevance, quality and satisfaction. The authors suggested that hybrid systems can take advantage of the positive aspects of both platforms. For example, social QA services are known to have lower-quality content but higher volume, so that virtual reference can be used when low-quality answers are identified on a social QA site. In [21], an approach to the evaluation of CQA websites is proposed with the goal of helping users in selecting high-quality CQA websites and assisting operators in performing improvements. The work proposes a multicriteria decision-making (MCDM) model to consider the ratings given by multiple decision makers from various perspectives. It was applied to evaluate the quality of five CQA websites, demonstrating its feasibility and practicality.

2.3. Quality of Questions and Answers

Beyond the quality of CQA websites as a whole, it has been argued that the quality of the question itself can have an important effect on the likelihood of obtaining useful answers. Baltadzhieva et al. [22] reviewed existing research on question quality in CQA websites, discussing the possible measures of question quality and the question features that have been shown to influence question quality. Ravi et al. [23] used latent topic models to automatically predict the quality of questions based on their content. This work studied questions from *StackOverflow* and defined the quality of a question based on a careful analysis of the interplay between the number of views and the number of upvotes a question garnered. Then, a binary classifier was used to distinguish good and bad questions.

Several works in the literature have addressed the problem of evaluating the quality of answers, as well as predicting the best possible answer for a question in a QA forum. Le et al. [24] proposed a framework to automatically assess the quality of educational questions by integrating different sets of characteristics (personal, community, textual and contextual) and building a classification model to determine what constitutes answer quality. Another framework is proposed in [25] to systematically and statistically process nontext features from some CQA services (*Google Answers* (http://answers.google.com/answers/) and *Yahoo! Answers*, among others), including click counts, recommendations, etc. Oppositely, Gkotsis et al. [6] considered that all the necessary information to determine the best answer in a forum is in its content and proposed a technique for prediction that leverages the content/textual features of answers. This technique was validated with 21 *StackExchange* sites, obtaining a high classification performance, proving that it is a robust, effective and applicable technique. Later, Burel et al. [4] generalized this technique, formalized it and compared the impact of the structural normalization of features. This work also studied how and why the importance of features changes when structural normalization is applied. Other authors propose hybrid models to predict the best answers based on content and noncontent characteristics. For example, a hybrid hierarchy-of-classifiers framework to model QA pairs is proposed in [26]. The question type is first analyzed to guide the selection of the right answer quality model, and then the information from question analysis is used to predict the expected answer features and train the type-based quality classifiers to hierarchically aggregate an overall answer quality score. Also, in [27], a hybrid model consisting of two modules is proposed. The first module considers three types of content features: question–answer features, answer content features and answer–answer features. The second module considers noncontent features by using a novel reputation score function. The results of both modules are merged for prediction.

In [28], a quality-aware framework that select answers from a community QA site considering answer quality in addition to answer relevance is proposed. In this framework, the quality of an answer is determined by the answer content as well as from the knowledge (e.g., expertise) about the user contributing the answer. Then, answers are ranked by combining both the relevance and quality scores.

The work in [3] also focused on the problem of assessing and predicting the quality of answers in QA services using features from the question, the answer and the user profile. They found that the several aspects of the overall quality of an answer, such as informativeness, completeness, novelty, etc., are highly correlated but limited when used for prediction. Instead, the profile of the person who answers, as measured by the points earned on *Yahoo! Answers*, and the order of the answer in the list of answers for a given question were the most significant features for predicting the best-quality answers. In [29], a framework that provides an effective way of integrating state-of-the-art transformers with linguistic properties via the use of traditional classifiers is proposed, primarily focused on gender screening across CQA sites. In [30], new interaction-based features depicting the amount of distinct interactions between an asker and answerer over time are introduced, improving the performance in predicting whether an answer will be selected as the best answer. In a different approach, Xie et al. [31] used previous knowledge from similar question–answer pairs to bridge the lexical and semantic gaps between questions and answers.

From the perspective of recommender systems, the recommendation of QA items has been addressed in recent works. For example, in [32], a context-awareness content-based (CA–CB) recommendation approach that introduces context awareness based on topic detection within current trend interest is presented. Specifically, the application domain is question-answering items (QA items), given that QA items have a strong component of textual information. In [33], an expert-recommendation task for community question answering is approached using a principled fusion of various types of information. The integrated information enables a characterization of community members in terms of three inherent properties, i.e., their tag-based temporally discounted interest, expertise and willingness to respond. Two model-based approaches to recommend repliers in question-answering communities are proposed in [34]. Both approaches incorporate temporal information of posts, their tags and the temporally discounted, tag-based propensity of experts to provide replies.

Early detection of high-quality content poses the challenge that only the value of a few features is available in a short time after submission of a content in CQA; therefore, Neshati [2] views the content quality from the perspective of the voting outcome. Questions and answers with more votes than a certain threshold are considered high-quality posts. The author observed two important patterns for the early detection of high-quality content: accepted answer effect (the chance of a high-quality question to receive an accepted answer) and answer competition effect (only a small number of answers of a specific question will be high-quality answers). Based on these two effects, in the proposed framework, the quality of a given question and its associated answers can be simultaneously predicted shortly after its publication. In [35], the problem of modeling the conversation history to answer the current question is addressed. The proposed solution enables seamlessly integrating the conversation history into a conversational question-answering (ConvQA) model built on BERT.

2.4. Contributions

The study carried out in this work is focused on evaluating the impact of a range of different features of answers on the prediction of the best answer in a discussion thread. To accomplish this goal, we analyze several aspects of an answer content (linguistic patterns, readability, the presence of certain elements such as examples, etc.) as well as features describing the interaction of users with the posted answer and the post history. This study is centered on a domain-specific forum, namely *Stack Overflow*, and data were collected about questions concerning a given programming language (*Java*).

In Community Question-Answering (CQA) sites, it is important to understand the factors that contribute to identifying high-quality responses. The quality of the responses is a key aspect for the success, growth and durability of collaborative platforms. CQA users seek reliable and accurate answers to their questions. A satisfactory experience not only encourages users to return to the site but also contributes to higher retention and active participation by all members of the community. When users feel that their efforts to provide valuable answers are recognized and valued, they are more motivated to actively participate, which enriches the community.

Having high-quality responses not only benefits users but also makes CQA a trusted source of collective knowledge. This generates more users wanting to participate in the community, regarding both asking and answering. Another important point is to eliminate unreliable, misleading and noisy information, as well as misinformation. By highlighting and promoting high-quality responses, erroneous or irrelevant content is filtered out, helping to maintain a more trustworthy and secure environment for users.

Ultimately, understanding the factors that lead to best responses in CQA fosters a collaborative and mutual learning environment. By analyzing and learning from the most effective answers, users can improve their abilities to ask more precise questions and provide more informative answers. This creates a virtuous cycle of growth and development within the community, where all members benefit by acquiring new knowledge and skills.

3. Features for Predicting Best Answers

In order to predict the best answer in a thread originated by a posted query, we analyzed several groups of features under the hypothesis that there are multiple factors related to an answer that make it valuable for the community and likely to become the best answer. Therefore, for an accurate prediction of answer quality, a good representation of such factors becomes an important issue.

In the selection of these features, we considered multiple aspects that can affect the quality of answers and, in turn, their chances to become the best answer for a question posted. First, the content similarity between question and answers is calculated as a crucial element in information retrieval systems. Then, features that can be derived from the text are included. Linguistic patterns captured by the corresponding features help to describe the user writing style (usually applied to author identification) and analyze the quality of texts. Likewise, automated readability assessment enables to quantify the difficulty in which text information can be read and understood. Thus, readability features indicate the ease of reading a text, in this case a provided answer. Other auxiliary elements that can help to enrich an answer and its understanding are the examples provided, the referenced material and, in development forums, the inclusion of source code (for example, to solve a problem or identifying a bug). Also, implicit indicators that an answer can be selected as the best one are given by the reaction of the community to a posted answer. This aspect is considered by features that describe the interaction of the community with a given question (e.g., how fast it was accepted). Finally, a temporal perspective of such interaction is included by considering the post history and how the post evolves over time.

For all of these aspects, a set of features was chosen based on the characteristics of the forum studied and the analysis of previous works. Table 1 lists the features used in our study. The column Ref. in the table cites works in which the mentioned features have been previously considered.

Table 1. List of features used in the study.

Feature Name	Feature Description	Ref.
	Content Similarity	
cosineSimilarity	Measure of similarity between question and answer	[5,24]
	Linguistic features	
postLength	Post length, measured in characters	[6,24,25,30]
#MaxWordsInSentence	Number of words in the longest sentence	[6]
avgWordsInSentence	Average number of words per sentence	[6]
avgCharactersInWords	Average number of character for words	[6]
#ComplexWords	Number of complex words in the answer	[30]
#MisspelledWords	Number of misspelled words	[24,30]
#CodeLines	Number of lines of source code	
#Links	Number of links	
	Readability features	
ARI	Automated Readability Index (ARI)	[24]

Table 1. *Cont.*

Feature Name	Feature Description	Ref.
GFI	The Gunning Fog Index (GFI)	[36,37]
FRES	Flesch Reading Ease Formula (FRES)	[24]
	Additional content elements	
hasExample	Include examples in the answer	
hasCodeSelection	Include source code in the answer	
hasLinks	Include references to external material	[3]
	Interaction features	
velAccepted	Difference between the timestamp of when the question was chosen as the best answer by the author and the timestamp of when the question was posted	[30]
velAnsw	Difference between the date and time the answer was posted and when the question was posted	[24,30]
	Post history features	
ageAnsw	Age in days of the answer from when it was published until the day it was retrieved	[4,6]
#Answers	Number of answers to a question in a thread	[3,4,6,25]
#AnswComments	Number of comments received by an answer	[3,4]

Ultimately, a broad range of features describing the various elements constituting an answer in a thread and its context are considered in this study and classified according to their type. Some of these features represent characteristics of the answer content, such as the text itself, whereas others involve contextual clues and feedback received from the community.

The studied features can be broadly categorized into six groups: content similarity, linguistic features, readability features, additional content elements, post history and interaction features. The impact of each group on the quality of predictions was assessed by evaluating the performance of the different groups separately, as well as their different combinations.

The rationale behind including each of these features for describing answers and how they are calculated is described in the following sections.

3.1. Content Similarity

The basic mechanism to retrieve answers from a given query is based on text similarity. Under the same hypothesis search engines are based on, the higher the matching among a question and the posted answers, the more related to the user information need and, in turn, the more likely to satisfy it.

To consider this aspect in predicting the best answer for a query, we include a feature measuring the content similarity between a question and a given answer in the thread. The traditional cosine similarity between word vectors is used to calculate this similarity.

$$sim(v_i, v_j) = \cos(\alpha) = \frac{\sum_{k=1}^{n} w_{ik} \cdot w_{jk}}{\sqrt{\sum_{k=1}^{r} w_{ik}^2} \cdot \sqrt{\sum_{k=1}^{r} w_{jk}^2}} \quad (1)$$

where v_i and v_j are the vectors corresponding to a posted query and a possible answer in a thread, and w_{ik} and w_{jk} are the weights of the work k in both vectors.

3.2. Linguistic Features

Linguistics takes into account the form of a text, so that certain linguistic characteristics can make an answer easy to comprehend for a reader and, therefore, make it more adequate for successfully satisfying the information need behind the question. Under this assumption, we consider a number of linguistic features which are directly extracted from the text contained in the answers.

- $postLength$: The post length measured in characters, excluding characters in source code when it is present as part of an answer.
- $\#MaxWordsInSentence$: The number of words in the longest sentence from the text of an answer. It is assumed that long sentences are harder to read and understand than shorter ones, thus affecting the quick understanding of an answer.
- $avgWordsInSentence$: The average number of words per sentence. This is also an indication of the presence of long and possibly difficult to understand sentences.
- $avgCharactersInWords$: The average number of characters for words. This feature is related to the use of long words, which are assumed to also be more difficult to read than shorter ones.
- $\#ComplexWords$: The number of complex words in the post. Based on the criteria used in [30], a word is considered complex if it has three or more syllables.
- $\#MisspelledWords$: The number of misspelled words. Some posts contain numerous misspelled words, which is related to informality and may affect the clarity of an answer's writing. A dictionary was used to check for misspelled words (http://jazzy.sourceforge.net/). The final number of misspelled words is estimated by looking up the detected words in a dictionary to confirm they are English words and that they are not a stop-word in the Onix (https://www.lextek.com/manuals/onix/stopwords1.html) corpus nor a contraction or a digit.
- $\#CodeLines$: The number of source code lines. Since the collected dataset contains a set of threads with the tag *java*, the number of source code lines is computed as the number of times the semicolon symbol ";" appears among open brackets. For every `for` sentence appearing in the code, 2 is subtracted from the number of lines due to its formal syntax. The semicolon in the heading of a `for` sentence separates the variable initialization, the condition and the increment of the control variable.
- $\#Links$: The number of links existing in the answer, which indicates references to external material to support an answer. This features counts the number of times the HTML tag *"<a href="* appears in the response.

3.3. Readability Features

Readability is a characteristic of texts that describe whether an individual reader will find a text accessible. For the purpose of predicting a good answer, it can be considered that the more readable and comprehensible a text is, the more chances it has of becoming the best answer for a given question.

In other terms, readability features measure how difficult to understand the text of a post (p_i) is. In this study, we used three metrics (*ARI*, *GFI* and *FRES*) to assess text readabilit, as detailed below.

The mentioned metrics use the average sentence length asl_{p_i}, the percentage of complex words pcw_{p_i} and the number of syllables within a word sww, which are calculated as follows:

$$asl_{p_i} = \frac{\#words}{\#sentences}$$

$$pcw_{p_i} = 100 \cdot \left(\frac{\#complex\,world}{\#words} \right)$$

$$sww = \left(\frac{\# of\ syllables}{\# of\ words} \right)$$

Based on these counts, the mentioned readability features are derived as follows:

- *ARI* (Automated Readability Index) [38]: This metric outputs a number which approximates the grade level needed to comprehend a given text. Higher scores indicate that the text requires a higher level of education to be understood than those with lower scores. *ARI* is measured by

$$ARI = 4.71 \cdot \left(\frac{\# characters}{\# words} \right) + 0.5 \cdot asl_{p_i} - 21.43 \tag{2}$$

- *GFI* (Gunning Fog Index) [39]: The Gunning Fog index of a post p_i is calculated using the average sentence length and the percentage of complex words. A higher score indicates easier-to-understand content, and it is calculated as

$$GFI = 0.4 \cdot \left(asl_{p_i} + pcw_{p_i} \right) \tag{3}$$

- *FRES* (Flesch Reading Ease Formula) [40]: The *FRES* formula is used to assess the difficulty of a reading passage. It is based on the number of syllables within a word. Higher *FRES* scores indicate the text is easier to understand.

$$FRES = 206.8 - 1.01 \cdot asl_{p_i} - 84.6 \cdot sww \tag{4}$$

3.4. Additional Content Elements

Answers might also contain certain elements to help the user better understand the written explanation, such as examples for illustrating a certain point, a piece of code or a link to additional material. The inclusion of these elements can be considered a hint to identify good answers. In this regard, we consider the following binary features in this study:

- *hasCodeSelection*: It is determined by the existence of the HTML $< code >$ tag in the response.
- *hasExample*: To determine the existence of an example within an answer, we search for the words *example*, *ex*. or *for instance*, in the text.
- *hasLinks*: It is determined by the existence of the HTML tag "*<a href=*" in the response.

3.5. Interaction Features

Interaction features are those representing information about the interaction along time between the user who asks a question and the users providing answers. This includes the amount of interaction between such users, the time span between the time the question was posted and the answer was given and the time span between the time the question was posted and the answer was marked as the best one. In this study, we use the last two.

- *velAnsw*: The time span between the moment the question was posted in the forum and the moment which it was answered. It can be presumed that quick responses can be rapidly adopted as the best answer if they satisfy the formulated information need. This feature is defined as the difference in days between question and answer, as indicated in Equation (5).

$$velAnsw = dateAnswer - dateQuestion \tag{5}$$

- *velAccepted*: The time span between the moment the question was posted in the forum and the moment the answer was deemed the best answer by the user who posted the question. It is calculated in the same way as *velAnsw*.

3.6. Post History

Each question opens a thread that runs its own course, which is registered in the forum. This group of features is designed to extract information from the thread history from its creation until the moment the snapshot of the forum was taken for this study. This information includes when a question was created, the amount of received answers and the number of associated comments.

- $ageAnsw$: The age in days of the answer from the moment it was published.
- $\#Answers$: The number of answers in the thread.
- $\#AnswComments$: The number of comments received by the answer.

4. Materials and Methods

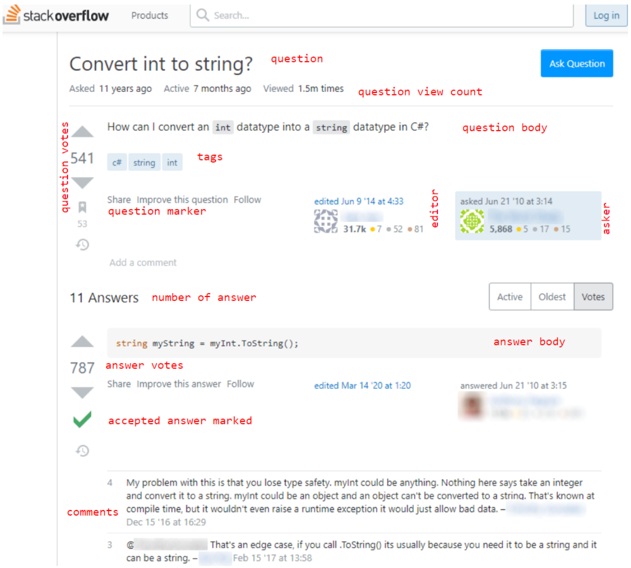
To investigate the relation of each set of features with the likelihood of an answer to become the best possible answer to a question posted in the forum, we collected data from a widely used programming forum, and then we empirically evaluated the impact of different features on predicting whether a given answer was selected as the best one.

4.1. Data Description

StackExchange (SE) is a popular question-answering community where users post questions on diverse subjects [41]. In particular, *Stack Overflow* (SO) represents a knowledge database within this question-answering community oriented to solving technological challenges, mainly about software engineering, programming and related issues. The knowledge made available in this repository is therefore very useful for reducing efforts in learning new technologies and dealing with complex technological issues [42].

SO works as a platform for collaborative information exchange, where users can question, answer and, if they are active members of the community, vote and edit published messages, which allows not only improving quality but also fostering active participation. For users in the community, the higher the quality of their posts and level of participation, the more privileges they can obtain, acquiring reputation and badges accordingly. Besides gaining reputation with questions and answers, users receive badges for being especially helpful. Badges appear on their profile page, flair and posts (https://stackoverflow.com/help/badges). For example, users earn ten points (+10) for every positive vote to their answers and a badge for their valuable contributions (e.g., the badge "good question" is obtained when a question reaches a score higher than 10). This process results in a sort of gamification (the application of typical elements of game playing, e.g., point scoring, competition with others and rules of play, to other areas of activity, typically as an online marketing technique to encourage engagement with a product or service.) within the QA site [2].

Figure 1 shows an example of a discussion thread in SO (https://stackoverflow.com/questions/3081916/convert-int-to-string). Each thread has a title or question, a question body, a user who is making the question and, optionally, a user that edits the question to improve its comprehension. The question receives votes and has a differential between positive and negative votes. Each question has 5 tags at most (https://stackoverflow.com/help/tagging), and different users can add question marks, which means they are also interested in a future answer. For the example in the figure (Figure 1), 53 question marks can be observed. The thread also displays the number of visits or views and the date and time, which are visible for every question posted by the community.



stackoverflow Products Search... Log in

Convert int to string?

Ask Question

Asked 11 years ago Active 7 months ago Viewed 1.5m times

541

How can I convert an `int` datatype into a `string` datatype in C#?

c# string int

Share Improve this question Follow

53

edited Jun 9 '14 at 4:33
[redacted]
31.7k ●7 ●52 ●81

asked Jun 21 '10 at 3:14
[redacted]
5,868 ●5 ●17 ●15

Add a comment

11 Answers

Active Oldest Votes

```
string myString = myInt.ToString();
```

787

Share Improve this answer Follow

edited Mar 14 '20 at 1:20

answered Jun 21 '10 at 3:15
[redacted]

✓

4 My problem with this is that you lose type safety. myInt could be anything. Nothing here says take an integer and convert it to a string. myInt could be an object and an object can't be converted to a string. That's known at compile time, but it wouldn't even raise a runtime exception it would just allow bad data. – [redacted] Dec 15 '16 at 16:29

3 @[redacted] That's an edge case, if you call .ToString() its usually because you need it to be a string and it can be a string. – [redacted] Feb 15 '17 at 13:58

Figure 1. An example of a question on Stack Overflow.

A thread has the number of answers for the question posted; in turn, each answer has a body and also might have comments. The green check box in Figure 1 indicates that the answer was accepted by the person who posted the question in the first place. In SO, "accepting an answer is not meant to be a definitive and final statement indicating that the question has now been answered perfectly. It simply means that the author received an answer that worked for him or her personally, but not every user comes back to accept an answer, and of those who do, they may not change the accepted answer if a newer, better answer comes along later" (https://stackoverflow.com/help/accepted-answer). Like questions, answers receive votes; the question in the figure, for example, has a differential in positive and negative votes of 787 to the day the information was retrieved.

The data used in this work were obtained from SE, particularly from the *Stack Overflow* forum. SE makes available to the community a repository with all databases of discussion forums, including questions, answers, comments, votes, users, etc. Particularly, the data used for experimentation in this paper are a dump made on 3 June 2019. The entire dataset was filtered for questions in a two-year period, from 1 January 2017 to 31 December 2018. Questions were restricted to those having at least 5 answers to ensure a minimum of information for prediction, resulting in a total of 4020 threads. From these threads, 412 threads were discarded because they had no answers with more than zero votes, reaching a final number of 3608 threads. This decision was based on the observation that a good-quality answer is one with more positive votes among all answers in the thread.

For each of the 3608 remaining threads, we determined the most-voted answers (one or more if they had equal score), i.e., those having the best score. Then, we looked for the less-voted answers, i.e., those with the lowest scores. In this process, another 173 threads were discarded, as the best and worst scores were equal, so that 3435 threads remained in the dataset. From them, 4714 answers were extracted with the best scores and 9107 with the worst scores.

To achieve a balanced dataset, we took the complete set of answers with higher scores (because it is the minority class) and we randomly chose 4714 posts from those receiving the lowest scores. Thus, the final balanced dataset has 9428 posts of answers (4714 of each class: the best- and worst-scoring answers). A summary of the final dataset can be found in Table 2.

Table 2. Data Summary.

	Discussion Threads
Site	StackOverflow
Language	English
URL	http://stackoverflow.com/
Date	June 2019
selection criteria	2 years (2017–2018)
# initial_threads	4020
# final_threads	3435
# total_posts	9428
# best_scores	4714
# worst_scores	4714

4.2. Methodology

In order to evaluate the performance of the different set of features previously described, we used classical machine learning algorithms with the corpus of question–answer threads collected from *Stack Overflow*. As previously mentioned, the goal of this study is to dig into the relationships of multiple features and the best answers selected by users through prediction. It is worth noting that the impact of features can be also assessed thought an exploratory analysis based on information theory metrics (e.g., information gain, mutual information and other metrics), which measure how much each feature is expected to contribute to learning (pretraining), or through explanation methods (e.g., SHAP values), which measure how much a feature actually impacts the model learned with a given machine learning algorithms (post-training). In the first case, the interaction of features during learning is neglected since they are analyzed in isolation, whereas in the second one, the results are tied to a specific learning algorithm. Considering these drawbacks, we decided to assess the impact of features, analyzed in groups, by measuring the gains in prediction with different learning algorithms.

The influence of features is, therefore, assessed through their impact on classifying answers as being chosen as the best answer or not. The influence of features can be affected by the way supervised learning algorithms use them for learning a model. For example, decision tree algorithms work top-down by choosing a feature at each step that best splits the data, thus treating features in an isolated manner, whereas other algorithms consider the interaction among features. For this reason, three different well-known classification algorithms, naïve Bayes (NB), logistic regression (LR) and random forest (RF), were applied in this work. The goal is not to optimize prediction results but to ensure that the findings are not biased by the way a particular algorithm ponders features.

The main concern in this study was to evaluate the role of different features and how the different sets of features interact with each other in the prediction of whether an answer was chosen as the best one or not. Therefore, the applied methodology was to evaluate the impact of the different groups alone and incorporate the next-best group in each step until all the sets were included in the set of features employed for learning.

As mentioned in Section 4.1, a total of 9428 posts were used for the experiments. The goal of this empirical evaluation was to determine the impact of each set of features in classification, considering multiple learning algorithms.

5. Experimental Evaluation

To accomplish the previous goal, the learning algorithms were first run using each individual set of features described in Table 1 to assess their performance separately. From evaluating the results, we looked closely at the *true positive rate* (TPR), the number of correct decisions about the best answers, as we are interested in distinguishing such answers from lower-quality ones. In other words, we wanted to observe whether a given algorithm was able to recognize the best answers given a particular set of features. The F-measure is also reported as a global metric of classification performance.

In the first experiment, therefore, classifiers used as input each individual set of features and learned from the examples provided. Thus, a classifier was learned using only the linguistic features, then another was learned using the readability ones and so on. After analyzing the performance of the individual feature sets, the best-performing sets were combined with other groups of features to analyze their interaction.

Table 3 shows the results obtained for each set of features evaluated individually with the different classifiers. In the table, the best values of TPR and F-measure are in bold, as these are the metrics we considered to guide the experimental evaluation, since we are interested in the hits on the positive class (best answer). The table also reports the values of precision, recall and ROC area.

Table 3. Classification results for an individual set of features.

Classifier/Feature Set	**NB**					**LR**					**RF**				
	TPR	**Prec**	**Recall**	**F1**	**ROC**	**TPR**	**Prec**	**Recall**	**F1**	**ROC**	**TPR**	**Prec**	**Recall**	**F1**	**ROC**
Content similarity (*Csim*)	0.541	0.549	0.541	0.545	0.564	0.507	0.553	0.507	0.529	0.564	0.512	0.515	0.512	0.514	0.524
Linguistic (*Lng*)	0.306	0.599	0.306	0.405	0.583	0.471	0.599	0.471	0.527	0.602	**0.57**	0.554	0.57	**0.562**	0.577
Readability (*Rea*)	**0.767**	0.52	0.767	**0.620**	0.546	**0.585**	0.517	0.585	**0.549**	0.514	0.513	0.517	0.513	0.515	0.524
Content elements (*Con*)	0.312	0.589	0.312	0.408	0.563	0.304	0.596	0.304	0.403	0.567	0.288	0.6	0.288	0.39	0.566
Interaction (*Int*)	0.332	0.92	0.332	0.487	0.665	0.333	0.92	0.333	0.489	0.669	0.346	0.859	0.346	0.493	0.657
Post history (*His*)	0.06	0.537	0.06	0.109	0.536	0.547	0.524	0.547	0.535	0.54	0.453	0.465	0.453	0.459	0.453

As can be observed, readability features (*Rea*) outperformed all other features, reaching 0.767 with naïve Bayes and 0.585 with a logistic regressor, and they are the second-best using random forest classification (0.513). Linguistic features (*Lng*) obtained the best result with a random forest classifier, reaching a score of 0.57. In all cases, the classifier with the best TPR also obtained the best F-measure scores.

An important issue to consider in the analysis of these results is that the classification tasks proposed have some difficulty, since classifiers have to predict if a given answer was chosen as the best one or not among several possibly good answers (e.g., the second-best answer is labeled as belonging to the negative class). More likely, there are good answers in the dataset that were not selected as the best ones for a number of reasons, in spite of their good quality. There is also some subjectivity involved in the decision of the user who posted the query in selecting one answer over any other as the best one. Given the nature of the task, therefore, the performance scores might be underestimated in their capacity of predicting good answers, as only one answer was chosen by the user who posted the query.

These initial experiments allow us to infer that readability is a characteristic that can be deemed relevant to qualify an answer as a high-quality one. Based on this result, we combine the *Rea* features with the other sets to further explore their impact on classification. The naïve Bayes (NB) classifier was used in the remaining experiments, as it was the one reaching the best overall results. Thus, the following pairs of feature sets were evaluated: *Rea-Lng*, *Rea-Con*, *Rea-Int*, *Rea-His* and *Rea-Csim*.

Table 4 summarizes the results achieved by these combinations in terms of both TPR and F1. The best performance in identifying the best answers was reached by combining readability features with content similarity (*Rea-Csim*), with 0.741 in terms of TPR. Although not the best performing in terms of overall classification performance (F1), it was close to the best scores, too. In spite of being the best result, it did not improve the value of readability features alone (0.767), which means that content similarity might reduce the impact of readability in identifying a good answer.

Table 4. Results of NB classification for different combinations of feature sets.

Number of Combined Sets	TPR	Prec	Recall	F1	ROC
			1st set		
Rea	**0.767**	0.52	0.767	**0.62**	0.546
			2-sets combination		
Rea-Lng	0.386	0.589	0.386	0.467	0.583
Rea-Con	0.687	0.552	0.687	0.612	0.59
Rea-Int	0.626	0.622	0.626	**0.624**	**0.69**
Rea-His	0.542	0.547	0.542	0.545	0.56
Rea-Csim	**0.741**	0.531	0.741	0.619	0.575
			3-sets combination		
Rea-Csim-Lng	0.413	0.594	0.413	0.488	0.592
Rea-Csim-Con	**0.676**	0.561	0.676	0.613	0.601
Rea-Csim-Int	0.591	0.644	0.591	**0.616**	**0.698**
Rea-Csim-His	0.535	0.563	0.535	0.549	0.581
			4-sets combination		
Rea-Csim-Con-Lng	0.405	0.596	0.405	0.483	0.601
Rea-Csim-Con-Int	**0.591**	0.662	0.591	**0.625**	**0.711**
Rea-Csim-Con-His	0.512	0.581	0.512	0.545	0.601
			5-sets combination		
Rea-Csim-Con-Int-Lng	**0.522**	0.702	0.522	**0.599**	**0.712**
Rea-Csim-Con-Int-His	0.51	0.716	0.51	0.596	0.71
			6-sets combination		
Rea-Csim-Con-Int-Lng-His	**0.51**	0.714	0.51	**0.595**	**0.71**

In a similar way, we combine the best pair of sets (*Rea-Csim*) with a third one, as is reported in the table. In this case, it is possible to see that the best TPR score (0.676) was achieved by combining the sets *Rea-Csim-Con*. Like before, we then added the remaining sets to *Rea-Csim-Con*, creating *Rea-Csim-Con-Int*, which was the best combination. Lastly, the combination of the six possible sets achieves a TPR score of 0.51, far from the initial 0.767 achieved by the *Rea* features alone.

Considering the best results for each combination of feature sets, in Figure 2, it is possible to observe the decay in TPR as new sets are included in the representation of answers. At the same time, the figure shows that F1 scores remain rather stable as feature groups are added. This implies that the classifiers become increasingly worse in identifying the best possible answers but compensate for this loss by accurately classifying bad answers.

Readability (*Rea*) features alone are the ones reaching the best classification performance, denoting that this is an important factor for detecting the best answers. When another set is added in combination with *Rea* features, the classification performance diminishes. The amount of these performance drops is initially small, but comparing the first group and the last one, the TPR suffers a significant loss (≈25%).

In order to corroborate these results, i.e., the strong influence of readability in predicting the best answers, we repeated the same experimental procedure using logistic regression as well as random forest classifiers.

Table 5 shows the result of using logistic regression (LR) with the different sets of features and their combinations. Again, the classifiers achieved the best TPR results with *Rea* features (0.585), followed by the combination of the readability and post history features (*Rea-His*). However, no combination was able to overcome the use of *Rea* features alone.

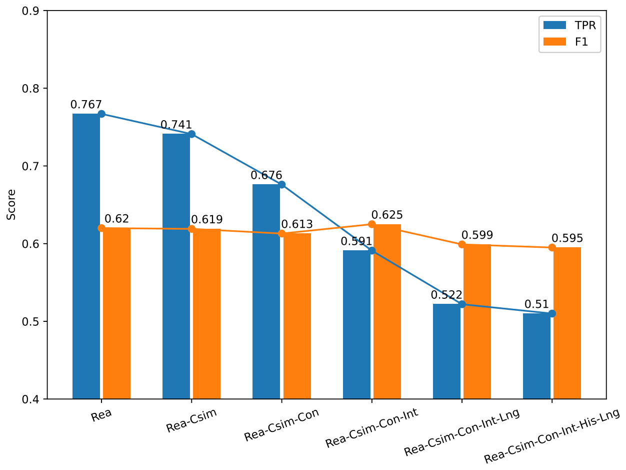


Figure 2. Learning performance for different combinations of feature sets using NB.

Table 5. Results of LR classification for different combinations of feature sets.

Number of Combined Sets	**TPR**	**Prec**	**Recall**	**F1**	**ROC**
		1st set			
Rea	**0.585**	0.517	0.585	0.549	0.514
		2-sets combination			
Rea-Lng	0.493	0.591	0.493	0.538	0.606
Rea-Con	0.481	0.564	0.481	0.519	0.586
Rea-Int	0.333	0.92	0.333	0.489	**0.671**
Rea-His	**0.557**	0.53	0.557	**0.543**	0.544
Rea-Csim	0.525	0.551	0.525	0.538	0.566
		3-sets combination			
Rea-His-Lng	0.517	0.584	0.517	0.548	0.609
Rea-His-Con	**0.559**	0.561	0.559	**0.56**	0.593
Rea-His-Int	0.333	0.92	0.333	0.489	**0.68**
Rea-His-CSim	0.549	0.554	0.549	0.551	0.577
		4-sets combination			
Rea-His-Con-Lng	0.551	0.595	0.551	**0.572**	0.622
Rea-His-Con-Int	0.44	0.782	0.44	0.563	**0.709**
Rea-His-Con-Csim	**0.56**	0.57	0.56	0.565	0.604
		5-sets combination			
Rea-His-Con-Csim-Lng	**0.56**	0.592	0.56	**0.576**	0.627
Rea-His-Con-Csim-Int	0.44	0.783	0.44	0.563	**0.71**
		6-sets combination			
Rea-His-Con-Csim-Lng-Int	**0.471**	0.764	0.471	0.583	**0.725**

Figure 3 depicts the impact of different combinations of feature sets on classification performance. As mentioned, the *Rea* features alone reached the best performance. Combined with other sets, the performance slightly dropped, increasing again when content similarity and additional content elements were included (0.559 and 0.56, respectively). As opposed to the performance of NB, the loss in TPR values as feature sets added to training is small, except when all sets are taken together, in which case it seems the classifier is worse at distinguishing good answers but still good as an overall classifier.

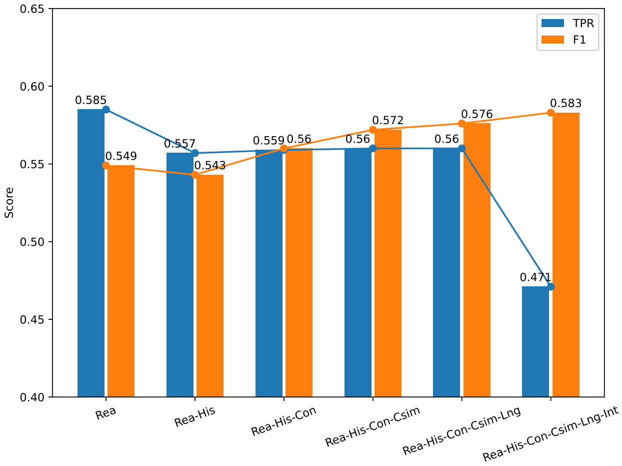


Figure 3. Learning performance for different combinations of feature sets using LR.

As with the previous results, Table 6 shows the result of using random forest classification with the different sets of features. In this case, RF classifiers reached their best performance in terms of TPR (0.57) with *Lng* features. Then, the results show that the combination of linguistic features and content similarity (*Lng-Csim*) achieved a TPR score of 0.575, outperforming the set *Lng* alone. Again, the *Rea* features have a positive impact on prediction.

Table 6. Results of RF for different combinations of feature sets.

Number of Combined Sets	**TPR**	**Prec**	**Recall**	**FM**	**ROC**
(**a**)					
1st set					
Lng	**0.57**	0.554	0.57	**0.562**	**0.577**
2-sets combination					
Lng-Rea	0.566	0.554	0.566	0.56	0.581
Lng-Con	0.567	0.552	0.567	0.56	0.577
Lng-Int	0.561	0.664	0.561	**0.608**	**0.698**
Lng-His	**0.578**	0.565	0.578	0.572	0.593
Lng-Csim	0.575	0.561	0.575	0.568	0.59
3-sets combination					
Lng-His-Rea	**0.578**	0.562	0.578	0.57	0.596
Lng-His-Con	0.574	0.563	0.574	0.569	0.595
Lng-His-Int	0.54	0.676	0.54	**0.601**	**0.702**
Lng-His-Csim	0.584	0.578	0.584	0.581	0.607
4-sets combination					
Lng-His-Con-Rea	**0.571**	0.566	0.571	**0.569**	0.593
Lng-His-Con-Int	0.554	0.654	0.554	0.6	**0.69**
Lng-His-Con-Csim	0.569	0.554	0.569	0.562	0.584
5-sets combination					
Lng-His-Con-Int-Rea	0.549	0.684	0.549	**0.609**	**0.71**
Lng-His-Con-Int-Csim	**0.533**	0.679	0.533	0.597	0.703
6-sets combination					
Lng-His-Con-Int-Rea-Csim	**0.537**	0.693	0.537	**0.605**	0.708

Table 6. *Cont.*

Number of Combined Sets	TPR	Prec	Recall	FM	ROC
			(**b**)		
			1st set		
Rea	**0.513**	0.517	0.513	**0.515**	0.524
			2-sets combination		
Rea-Lng	**0.566**	0.554	0.566	0.56	0.581
Rea-Con	0.533	0.534	0.533	0.533	0.547
Rea-Int	0.548	0.637	0.548	**0.589**	**0.671**
Rea-His	0.536	0.522	0.536	0.529	0.537
Rea-Csim	0.544	0.532	0.544	0.538	0.549
			3-sets combination		
Rea-Lng-Con	0.574	0.558	0.574	0.566	0.581
Rea-Lng-Int	0.555	0.676	0.555	**0.609**	**0.704**
Rea-Lng-His	**0.578**	0.562	0.578	0.57	0.596
Rea-Lng-Csim	0.576	0.568	0.576	0.572	0.593
			4-sets combination		
Rea-Lng-His-Con	**0.582**	0.566	0.582	0.574	0.595
Rea-Lng-His-Int	0.547	0.685	0.547	**0.608**	**0.704**
Rea-Lng-His-Csim	0.579	0.577	0.579	0.578	0.608
			5-sets combination		
Rea-Lng-His-Con-Int	0.548	0.687	0.548	**0.61**	**0.704**
Rea-Lng-His-Con-Csim	**0.581**	0.575	0.581	0.578	0.608
			6-sets combination		
Rea-Lng-His-Con-Csim-Int	**0.537**	0.693	0.537	**0.605**	**0.708**

The combination obtaining the best result is *Lng-His*, i.e., linguistic features and the post history features, with a TPR of 0.578. Adding readability features maintained such performance, which is improved using content similarity, reaching 0.584. Adding other elements of content (*Lng-His-Csim-Con*) leads to further improvement, reaching the highest overall performance in distinguishing the best answers. In terms of F1 score, it can be observed that scores are about 0.60 with several combinations of sets, even considering only *Lng* and *Int* combined. Since we are interested in identifying good answers, we analyzed the true positive rate, as it judges the classifier decisions in this regard.

Since the *Rea* features were shown to have a strong impact on the NB and LR classifiers, we also observed their behavior by taking them as the initial set with RF classifiers. In this setting, the results improve as new sets are included into training, with the best result being the combination *Rea-Lng*. The performance improves as new sets are added, except for *Int*, which causes a decrease in TPR score.

Figures 4 and 5 compare the results of random forest (RF) classification starting with the *Lng* and *Rea* features, respectively. In both cases, TPR grows with new added features until the last two groups of features are added. A steady increase in TPR can be observed until the fourth combination, starting from which the values of TPR diminished. Similarly, in Figure 5, a big drop in TPR is caused by the inclusion of *Int* features.

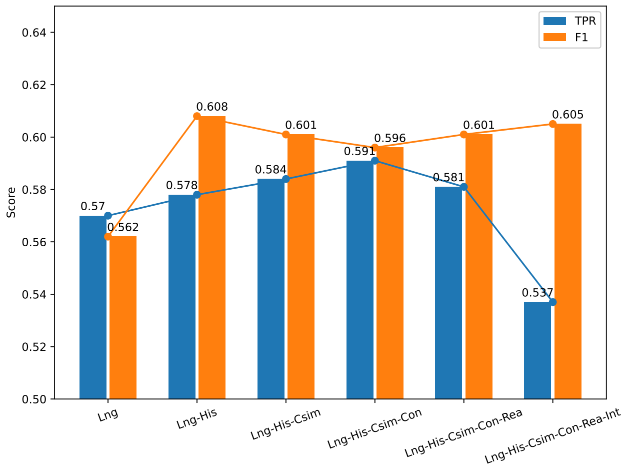


Figure 4. Learning performance for different combinations of feature sets using RF starting with *Lng* features.

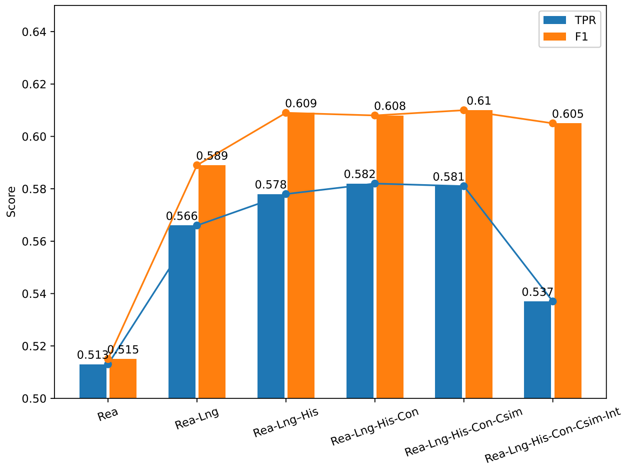


Figure 5. Learning performance for different combinations of feature sets using RF starting with *Rea* features.

To sum up, the results of the three classification models learned show that *Rea* are the most salient features in identifying the best possible answer to a question. This finding has interesting practical implications for both individual users and developers of CQA platforms. For users that are keen to improve their reputation and earn badges, this means that they have to direct their efforts to generate more readable content. For CQA platforms trying to provide high-quality content, actions can be put in place to guide users in this direction (e.g., providing guidelines).

Content similarity is also a contributing factor in most scenarios, leading to small increases in performance when it was incorporated into classifiers. The addition of examples, source code or links, described by *Con* features, had relatively less weight in the identification of the best answers. The *Lng* features showed different behavior depending on the classifier used, being less relevant in NB classification but more important for RF classifiers. Oppositely to the previously mentioned feature sets, the *Int* features seem to degrade the performance of most of the algorithms evaluated in this study.

Although results in the literature cannot be compared directly with ours, not only because they did not share the same goal but also because the datasets as well as the methodology used to build them was different, there are some some analogies that can be mentioned to contextualize and discuss the achieved results. In [36], human editors were used to train a classifier for high- and low-quality questions and answers in Yahoo!

Answers. Features applied included linguistic ones, such as word n-grams, and authors reported 67% precision for identifying high-quality answers. They also reported an F1 value of 58% for the task of finding high-quality questions using only usage features. In this work, a decision tree algorithm was used for prediction. Adamic et al. [9] combined user attributes and answer characteristics to predict, within a given category of Yahoo! Answers, whether a particular answer will be chosen as the best answer by the asker. The authors report 72.9% of accuracy using logistic regression for prediction. Having experimented with a number of different classifiers, the work in [6] reported results of the best-performing algorithm, Alternate Decision Trees (ADT), in predicting the best answers in several *StackExchange* sites. Linguistic and other features (e.g., age and creation date) reached a 76% F1 score, which was only improved by using user ratings. As previously mentioned, these results cannot be strictly compared with the ones achieved in this work, but they exhibit the same order of magnitude of the performance metrics and use the same types of classification algorithms used in this work (classical algorithms such as decision trees). The aim of this study was not to optimize classification results but to analyze the comparative performance of different sets of features.

Ths study provides valuable insights into the role of different features; however, it is worth mentioning some limitations and threats to validity. First, even though a large dataset was used, the study is centered on a single dataset oriented to programmers, so that conclusions cannot be straightforwardly extrapolated to other QA sites in diverse domains. In addition, the selection of the best answers was based on the criteria of the most-voted answer, but other provided answers might also be good ones. This can lead to underestimating the learning results, as only one answer (the most-voted one) is selected as the best and other, possibly equally good answers, are neglected. Also, in the selection of threads, some decisions were imposed (like number of votes) which can reduce the coverage of the information employed by the study. Regarding the learning approaches used for comparison, classical machine learning algorithms were used, but a wider battery of algorithms can help to further validate the achieved results.

6. Conclusions

Finding the best answers for questions posted on a CQA site is a task that is becoming increasingly important as the volume of knowledge on these platforms grows exponentially. Several features can be used to describe both the textual content as well as nontextual elements related to an answer for a posted question. In this study, the impact of these features in predicting the best answers was evaluated using real data from the *Stack Overflow* forum.

Several classification algorithms were applied in order to learn to predict the best answer in a discussion thread. The reported results show that an answer readability, described in terms of several readability metrics (*Rea* features), is the most relevant element for prediction. This result has practical implications, as expert users can work on improving readability to increase their reputation in CQA, and platforms could even produce guidelines to improve the content generated on the site and its overall quality. Content similarity (*Csim*) is also an important feature and should not be discarded, but it is possibly better combined with readability for obtaining better ranking. Finally, interaction features (*Int*) showed not to be effective for prediction. Possibly, the features used did not fully describe this aspect of answers and can be improved with further elements. Likewise, the social role of users is a factor that needs to be further analyzed in this context to determine whether their reputation or social influence contributes to predicting the best answers. Moreover, social influences, as well as trust and reputation, should be considered within a certain defined scope.

Author Contributions: Conceptualization, V.Z., D.G. and G.N.A.; methodology, V.Z., D.G. and G.N.A.; software, V.Z. and G.N.A.; validation, V.Z. and G.N.A.; formal analysis, V.Z., D.G. and G.N.A.; investigation, V.Z., D.G. and G.N.A.; resources, V.Z., D.G. and G.N.A.; data curation, V.Z. and G.N.A.; writing—original draft preparation, V.Z., D.G. and G.N.A.; writing—review and editing, V.Z., D.G. and G.N.A.; visualization, V.Z. and G.N.A.; supervision, D.G. and G.N.A.; project administration, D.G. and G.N.A.; funding acquisition, D.G. and G.N.A. All authors have read and agreed to the published version of the manuscript.

Funding: This work has been partially funded by ANPCyT (Argentina) under grant PICT-2020-SERIEA-01375. It was also partially supported by the GIISCo Research Group, research project 04/F018 from Comahue National University.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Srba, I.; Bielikova, M. A comprehensive survey and classification of approaches for community question answering. *ACM Trans. Web TWEB* **2016**, *10*, 1–63. [CrossRef]
2. Neshati, M. On early detection of high voted Q&A on Stack Overflow. *Inf. Process. Manag.* **2017**, *53*, 780–798.
3. Shah, C.; Pomerantz, J. Evaluating and predicting answer quality in community QA. In Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, Geneva, Switzerland, 19–23 July 2010; ACM: Frisco, TX, USA, 2010; pp. 411–418.
4. Burel, G.; Mulholland, P.; Alani, H. Structural normalisation methods for improving best answer identification in question answering communities. In Proceedings of the 25th International Conference Companion on World Wide Web, Montreal, QC, Canada, 11–15 April 2016; International World Wide Web Conferences Steering Committee: Geneva, Switzerland, 2016; pp. 673–678.
5. Niemann, M.M. The Duality of Expertise: Identifying Expertise Claims and Community Opinions within Online Forum Dialogue. Ph.D. Thesis, Monash University, Melbourne, Australia, 2015.
6. Gkotsis, G.; Stepanyan, K.; Pedrinaci, C.; Domingue, J.; Liakata, M. It's all in the content: State of the art best answer prediction based on discretisation of shallow linguistic features. In Proceedings of the 2014 ACM Conference on Web Science, Bloomington, IN, USA, 23–26 June 2014; ACM: Frisco, TX, USA, 2014; pp. 202–210.
7. Surowiecki, J. *The Wisdom of Crowds: Why the Many are Smarter than the Few and How Collective Wisdom Shapes Business, Economies, Societies and Nations*, 1st ed.; Time Warner Books UK: London UK, 2004; p. 320.
8. Roy, P.K.; Saumya, S.; Singh, J.P.; Banerjee, S.; Gutub, A. Analysis of Community Question-Answering issues via machine learning and deep learning: State-of-the-art review. *CAAI Trans. Intell. Technol.* **2023**, *8*, 95–117. [CrossRef]
9. Adamic, L.A.; Zhang, J.; Bakshy, E.; Ackerman, M.S. Knowledge sharing and Yahoo Answers: Everyone knows something. In Proceedings of the 17th International Conference on World Wide Web, Beijing, China, 21–25 April 2008; pp. 665–674.
10. Preece, J.; Nonnecke, B.; Andrews, D. The top five reasons for lurking: Improving community experiences for everyone. *Comput. Hum. Behav.* **2004**, *20*, 201–223. [CrossRef]
11. Yang, Z.; Liu, Q.; Sun, B.; Zhao, X. Expert recommendation in community question answering: A review and future direction. *Int. J. Crowd Sci.* **2019**, *3*, 348–372. [CrossRef]
12. Li, X.; Liu, Y.; Zhang, M.; Ma, S. Early detection of promotion campaigns in community question answering. In Proceedings of the Chinese National Conference on Social Media Processing, Nanchang, China, 29–30 October 2016; Springer: Berlin/Heidelberg, Germany, 2016; pp. 172–185.
13. Riahi, F.; Zolaktaf, Z.; Shafiei, M.; Milios, E. Finding expert users in community question answering. In Proceedings of the 21st International Conference on World Wide Web (WWW '12 Companion), Lyon, France, 16–20 April 2012; pp. 791–798.
14. Le, L.T.; Shah, C. Retrieving rising stars in focused Community Question-Answering. In Proceedings of the Intelligent Information and Database Systems, Da Nang, Vietnam, 14–16 March 2016; Nguyen, N.T., Trawiński, B., Fujita, H., Hong, T.P., Eds.; Springer: Berlin/Heidelberg, Germany, 2016; pp. 25–36.
15. Li, X.; Liu, Y.; Zhang, M.; Ma, S.; Zhu, X.; Sun, J. Detecting promotion campaigns in community question answering. In Proceedings of the 24th International Conference on Artificial Intelligence (IJCAI'15), Buenos Aires, Argentina, 25–31 July 2015; pp. 2348–2354.
16. Movshovitz-Attias, D.; Movshovitz-Attias, Y.; Steenkiste, P.; Faloutsos, C. Analysis of the reputation system and user contributions on a question answering website: StackOverflow. In Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2013), Niagara Falls, ON, Canada, 25–28 August 2013; pp. 886–893. [CrossRef]
17. Bhatia, S.; Mitra, P. Adopting inference networks for online thread retrieval. In Proceedings of the AAAI Conference on Artificial Intelligence, Atlanta, GA, USA, 11–15 July 2010; Volume 24; pp. 1300–1305. [CrossRef]
18. Bonifazi, G.; Cauteruccio, F.; Corradini, E.; Marchetti, M.; Sciarretta, L.; Ursino, D.; Virgili, L. A space-time framework for sentiment scope analysis in social media. *Big Data Cogn. Comput.* **2022**, *6*, 130. [CrossRef]

19. Ursino, D.; Virgili, L. An approach to evaluate trust and reputation of things in a Multi-IoTs scenario. *Computing* **2020**, *102*, 2257–2298. [CrossRef]
20. Shah, C.; Kitzie, V. Social Q&A and virtual reference comparing apples and oranges with the help of experts and users. *J. Am. Soc. Inf. Sci. Technol.* **2012**, *63*, 2020–2036.
21. Li, M.; Li, Y.; Peng, Q.; Wang, J.; Yu, C. Evaluating Community Question-Answering websites using interval-valued intuitionistic fuzzy DANP and TODIM methods. *Appl. Soft Comput.* **2021**, *99*, 106918. [CrossRef]
22. Baltadzhieva, A.; Chrupała, G. Question quality in community question answering forums: A survey. *ACM SIGKDD Explor. Newsl.* **2015**, *17*, 8–13. [CrossRef]
23. Ravi, S.; Pang, B.; Rastogi, V.; Kumar, R. Great question! Question quality in community Q&A. In Proceedings of the International AAAI Conference on Web and Social Media, Ann Arbor, MI, USA, 1–4 June 2014; Volume 8, pp. 426–435.
24. Le, L.T.; Shah, C.; Choi, E. Evaluating the quality of educational answers in community question-answering. In Proceedings of the 16th ACM/IEEE-CS on Joint Conference on Digital Libraries, Newark, NJ, USA, 19–23 June 2016; ACM: Frisco, TX, USA, 2016; pp. 129–138.
25. Jeon, J.; Croft, W.B.; Lee, J.H.; Park, S. A framework to predict the quality of answers with non-textual features. In Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Seattle, WA, USA, 11–15 July 2006; ACM: Frisco, TX, USA, 2006; pp. 228–235.
26. Toba, H.; Ming, Z.Y.; Adriani, M.; Chua, T.S. Discovering high quality answers in community question answering archives using a hierarchy of classifiers. *Inf. Sci.* **2014**, *261*, 101–115. [CrossRef]
27. Elalfy, D.; Gad, W.; Ismail, R. A hybrid model to predict best answers in question answering communities. *Egypt. Inform. J.* **2018**, *19*, 21–31. [CrossRef]
28. Suryanto, M.A.; Lim, E.P.; Sun, A.; Chiang, R.H. Quality-aware collaborative question answering: Methods and evaluation. In Proceedings of the Second ACM International Conference on Web Search and Data Mining, Melbourne, Australia, 9–12 February 2009; ACM: Frisco, TX, USA, 2009; pp. 142–151.
29. Figueroa, A. Refining fine-tuned transformers with hand-crafted features for gender screening on question-answering communities. *Inf. Fusion* **2023**, *92*, 256–267. [CrossRef]
30. Shah, C. Building a parsimonious model for identifying best answers using interaction history in community Q&A. In Proceedings of the 78th ASIS&T Annual Meeting: Information Science with Impact: Research in and for the Community, St. Louis, MI, USA, 6–10 November 2015; p. 51.
31. Xie, Z.; Nie, Y.; Jin, S.; Li, S.; Li, A. Answer quality assessment in CQA based on similar support sets. In *Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 309–325.
32. Castro, J.; Yera Toledo, R.; Alzahrani, A.A.; Sánchez, P.J.; Barranco, M.J.; Martínez, L. A big data semantic driven context aware recommendation method for question-answer items. *IEEE Access* **2019**, *7*, 182664–182678. [CrossRef]
33. Costa, G.; Ortale, R. Ask and Ye shall be Answered: Bayesian tag-based collaborative recommendation of trustworthy experts over time in community question answering. *Inf. Fusion* **2023**, *99*, 101856. [CrossRef]
34. Costa, G.; Ortale, R. Here are the answers. What is your question? Bayesian collaborative tag-based recommendation of time-sensitive expertise in question-answering communities. *Expert Syst. Appl.* **2023**, *225*, 120042. [CrossRef]
35. Qu, C.; Yang, L.; Qiu, M.; Croft, W.B.; Zhang, Y.; Iyyer, M. BERT with history answer embedding for conversational question answering. In Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'19), Paris, France, 21–25 July 2019; pp. 1133–1136. [CrossRef]
36. Agichtein, E.; Castillo, C.; Donato, D.; Gionis, A.; Mishne, G. Finding high-quality content in social media. In Proceedings of the 2008 International Conference on Web Search and Data Mining, Alto, CA, USA, 11–12 February 2008; pp. 183–194.
37. Burel, G.; He, Y.; Alani, H. Automatic identification of best answers in online enquiry communities. In Proceedings of the Extended Semantic Web Conference, Heraklion, Greece, 27–31 May 2012; Springer: Berlin/Heidelberg, Germany, 2012; pp. 514–529.
38. Senter, R.; Smith, E.A. *Automated Readability Index*; Technical Report; Cincinnati University: Cincinnati, OH, USA, 1967.
39. Gunning, R. *Technique of Clear Writing*; McGraw-Hill: New York, NY, USA, 1952.
40. Kincaid, J.P.; Fishburne, R.P., Jr.; Rogers, R.L.; Chissom, B.S. *Derivation of New Readability Formulas (Automated Readability Index, Fog Count and Flesch Reading Ease Formula) for Navy Enlisted Personnel*; Technical Report; Naval Technical Training Command Millington TN Research Branch: Millington, TN, USA, 1975.
41. Posnett, D.; Warburg, E.; Devanbu, P.; Filkov, V. Mining stack exchange: Expertise is evident from initial contributions. In Proceedings of the 2012 International Conference on Social Informatics, Alexandria, VA, USA, 14–16 December 2012; IEEE: New York, NY, USA, 2012; pp. 199–204.
42. Barua, A.; Thomas, S.W.; Hassan, A.E. What are developers talking about? An analysis of topics and trends in stack overflow. *Empir. Softw. Eng.* **2014**, *19*, 619–654. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.