An Algorithm for Estimating Answerers' Performance and Improving Answer Quality Predictions in QA Forums

Yonas Demeke Woldemariam
Dept. Computing Science, Umeå University, Sweden

Keywords: Answerer Performance Estimation, Syntactic-semantic based Algorithm, Answer Quality Assessments.

Abstract: In this study, a multi-components algorithm is developed for estimating answerer performance, largely from a syntactic representation of answer content. The resulting algorithm has been integrated into semantic based answer quality prediction models, and appears to significantly improve all testsets' baseline results, in the best case scenario. Upto 86% accuracy and 84% F-measure are scored by these models. Also, answer quality classifiers yeild upto 100% recall and 98% precision. Following the transformation of joint syntactic-punctuation information into the identified expertise dimensions (e.g., authoritativeness, analytical, descriptiveness, completeness) that formally define answerer performance, extensive algorithm analyses have been carried on almost 142,246 answers extracted from diverse sets of 13 different QA forums. The analyses prove that incorporating competence information into answer quality models certainly leads to nearly perfect models. Moreover, we found out that the syntactic based algorithm with semantic based models yield better results than answer quality prediction modles built on shallow linguistic or meta-features presented in related works.

1 INTRODUCTION

Despite the fact that textual content constitute the core part of most QA forums, *non-textual meta-features* available on the surface of these forums seem to direct and dominate several analytic works to soley rely on them (Harper et al., 2008; Shah and Pomerantz, 2010; Cai, 2013). As a result, many potential problems within QA forums remain unsolved because they are too complex to be caputured by surface meta-features.

Essentially, one of potential decisions within QA forums that requires deep (psycho)linguistic analysis is an **answer quality assessment**. Nevertheless, due the aforementioned reason, existing studies on answer quality limitied to make use either shallow linguistic information or simple meta-features. Moreover, those potential information (e.g., **answerer performance**) hidden in the actual answer content, and play a powerful role in estimating answer quality are not considered yet.

Within QA forums answer quality is always determined by askers. Obvioulsy, that raises many potential concerns, mainly due to the subjectivity among askers. Regardless of the subjectivity, **askers' satisfaction** seems to drives them to choose the *best answer* from other alternative answers. Apparently, behind every best answer there is a **compentent answerer** whose answer quality could be estimated through relevant text analysis methods. Yet, answerer competence is not explored to assess answer quality.

Thus, the joint effort of considering both the quality of linguistic constructions of answers and answerer performance, might help accurately determine *high quality answers*. To address the former deep analysis of answers, for example, via syntactic and semantic representations of answers is required. And the later could be acheived through identifying potential expertise dimensions that determine answerer performance from their answers.

2 RELATED WORK

A casual link between authors' proficiency in certain tasks and their associated text quality is explored by a number of studies (Chen et al., 2014; Woldemariam et al., 2017; Bryan and Robert, 2000). Most of these studies, however, considered various types of text (e.g., medical notes, pilot speech transcriptions) written by professionals, which are not directly related with QA content. Yet, there are a few studies (Tausczik and Pennebaker, 2011; Woldemariam, 2020; Woldemariam, 2021) which attempted to estimate user trustworthiness and contribution within QA

Woldemariam, Y.
An Algorithm for Estimating Answerers' Performance and Improving Answer Quality Predictions in QA Forums.
DOI: 10.5220/0010783100003116
In *Proceedings of the 14th International Conference on Agents and Artificial Intelligence (ICAART 2022) - Volume 3*, pages 106-113
ISBN: 978-989-758-547-0; ISSN: 2184-433X

forums from user-related features (e.g., reputation).

Chen et al. in (Chen et al., 2014) evaluate medical practitioners' competence from the bag-of-words (BoW) representation of their clinical portfolio. And the trained classifiers resulted in different outcomes across various competence dimensions. A syntactic based method together with a BoW is presented by Woldemariam et al. in (Woldemariam et al., 2017). Authors in (Woldemariam et al., 2017) aim to capture both structural and BoW information from user text to predict user proficiency. The predictions give interesting insights on how syntactic approaches produce better results than the BoW approach.

Tausczik and Pennebaker in (Tausczik and Pennebaker, 2010) studied the same problem with a psycholinguistic perspective. However, their method is completely based on word counting and assumptions that low ranked experts tend to be more self-focused (to be evidenced by use of first person singular pronouns) than competent experts. Tausczik and Pennebaker provide some empirical evidence to argue that levels of competenece and authority influence their language use. Quite a similar perspective is applied by authors in (Bryan and Robert, 2000). And the authors found out that pilots and crew member authoritativeness and inquisitiveness get reflected in their language.

Several studies on answer quality assessments agree that among those factors affecting answer quality, user-related attributes are dominant (Li et al., 2015; Molino et al., 2016; Suggu et al., 2016; Shah and Pomerantz, 2010; Tausczik and Pennebaker, 2011). For instance, Suggu et al. showed how reputation plays a key role in answer quality prediction. Also, Shah and Pomerantz, added expertise as information/answer quality measure among 13 manual criteria originally suggested in (Zhu et al., 2009).

3 A SYNTAX BASED ALGORITHM

In prior to the actual development of the competence estimation algorithm, an extensive **preliminarly analysis** with *fully annotated syntax-(seed) competence* data has been carried out with the StackOverflow dataset. Moreover, it makes use of evidence from psycho-linguistic text analysis (Tausczik and Pennebaker, 2010; Tausczik and Pennebaker, 2011), manual criteria used to judge answer/information quality (Zhu et al., 2009; Shah and Pomerantz, 2010), and syntactic patterns characterizing crowdsourced text (Woldemariam, 2021). Here, *reputation* has been used as **a seed-competence score**. That aims to group related syntactic units together and map into relevant expertise dimensions that could formally define answerer competence within QA forums. Following the joint syntax-punctuation annotation, users are divided into three groups (High, Middle, Low) based on seed competence. The resulted joint syntax-punctuation patterns of each group have been compared with others. And observable differences between such groups are noticed in terms of syntax usage. Moreover, the correlation coefficient(CC) of 0.62 has been found between the computed competence and number of answers. That is quite a good indicator of the strength between competence related features and expertise dimensions estimated from join syntax-punctuation information.

3.1 Transforming Syntax Trees into Expertise Dimensions and User-performance

Given that a question-answers pair **(QA)**, Algorithm **1** carries out 4 main tasks to compute core competence components. Firstly, it parses an answer content **(A)** and generates a parse tree **(answerSynTree)**. Secondly, it builds a map **(syntacticCatMap)** with **(syntacticUnit, synCatCount)** pairs by iterating through the resulting syntactic trees (a forest). Thirdly, the map has been further analyzed and supported with a predefined list of syntactic categories **(syntacticCatSet[])**. That is to keep track of each leaf and nonleaf node into a list **(synCatCount)**. Fourthly, it computes each expertise dimension (e.g., **comScore**) by iterating the resulting list. Finally, the remaining competence components **(e.g., questionComp)** are calaculated from meta-data information e.g., (reputation, badge). Among the identified expertise dimensions, we provide a breif description for some of them:

Descriptiveness: defines and quantifies the understandability as well as the clarity and simplicity of answer content. This expertise dimension has been formed using the composition of selected clauses (e.g., *simple declarative clauses (S)*), phrases (e.g., *adverb phrases (ADVP)*) and determiners. Moreover, this dimension is supported with important related infomation, for instance, **URLs** provided in answers and the **relevance** of the answers.

Analytical: measures answerer competence (e.g., in comparing items and ascertain with facts) and analytical skills reflected in their answers. Higher rates of syntactic categories (e.g., *quantifiers (QT)* and *list markers (LST)* might help identify potential contributors among answerers. Such expertise dimension is also enriched with other important syntactic units (e.g., *comparative adjectives (JJR)*, *cardinal numbers*

(CD)).

Coherency: measures answerers' quality in presenting coherent messages and connecting ideas. To acheive quantifying such dimension conjunctions (e.g., *coordinating conjunctions (CC), conjunctions phrases (e.g., conjunction phrases (CONJP)* and *prepositional phrases (PP)* are considered.

Inquisitiveness: refers how frequent answerers ask other users (askers) to further explain their questions. This expertise dimension has been made part of the overall user competence, because that helps provide good quality answers for answerers. Question marks with other relevant syntactic tags (e.g. SQ, WHNP) have been used to define such expertise category.

Focus Construction: roughly measures how well answerers make efforts to provide focused answers. To capture such information from parsed answers the syntactic tag *SINV (inverted sentences)* has been taken into account, though that does not completely address the selected dimension.

Completeness: measures how well answerers construct complete expressions at various levels (e.g., phrase, clause, sentence). Completeness, in other related studies, for example in (Blooma et al., 2008) has been found to be a good predictor of answer quality, though it was done manually (by human judges). Our algorithm mainly checks completeness at a sentence level, sentences containing at least *noun phrases (NP)* and *verb phrases (VP)*, along with periods, are considered to be complelete. To further catputre such information larger units (e.g., *subordinate clauses (SBAR)*) and those syntactic units closely related with VP and NP (e.g., *nouns (NN)*) are included. Moreover, the occurence of *fragments (FRAG)* and *reduced relative clauses (RRC)* are taken to penalize answerers due to their contribution towards forming *in-complete expressions*.

Complexity: measures the rate at which answerers construct unclear expressions from syntactic parsers point of view. Those syntactic tags that signal complex syntactic structures (e.g., *X*) or unknown grammatical constructions are considered to contexually define the *complexity* dimension, in a sense that answerers are writing in-complete experessions (Zhu et al., 2013).

Authoritativeness or Clout: assesses how confidently answerers provided their answers. That is measured by looking at the rate at which such answerers use **first person plural**, and the inverse of **first person singular**. This dimension is originally suggested in (Tausczik and Pennebaker, 2011).

Cognition: congnition words are those words which are helpful in building arguments (reasons). Congition has been estimated from the count of insight words (e.g., consider, think) and causal words (e.g., because, effect). Although cognition is not directly related with user-performance as other expertise dimensions as well as difficult to completely capture from answers content, it has been used in this study in its loose sense.

3.2 Integrating the User-performance Algorithm into Answer Quality Prediction Models

The strategy used to integrate the competence estimation algorithm into the answer quality assessments method is, to incorparate the computed competence components returned by the algorithm into answer quality prediction models as features characterizing answer quality. Afterwards, the impact of these components is evaluated. The evaluations are set up in such a way that, whether adding competence information or not has a significance impacts in the prediction of answer quality.

4 EXPERIMENTAL SETUP

4.1 Data

While the StackExchange (SE) QA forum is largest and oldest, among available forums under the SE network, we target more profession-oriented forums which reflect some sort of competence. A total of 14 forums' data-dumps have been directly collected from the SE (particularily from the **technology** forum) repository[1].

These datasets are used for the following three different tasks:joint syntax-competence analysis with StackOverflow, answer quality models building and model evaluation and algorithm analysis. The ServerFault dataset (54,710 answers) has been split into training, validation, and evaluation sets, 70%, 10% and 20%, respectively. For algorithms analysis and model evaluation tasks 13 independent QA forums have been selected. Four of them, ServerFault (SF), SuperUser (SU), WebApplications (WA) and Apple (AP), can be good examples of closely related domains, from the remaining 8 datasests, four of them, CraftCMS (CR), WordPress (WP), Drupal (DR), Magento (MA), SharePoint (SP), are examples of loosely related and, Software Engineering (SE), Game (GA), Blender (BL) and WebMaster (WM), un-related domains.

[1] https://archive.org/details/stackexchange

Algorithm 1: An Answerer Performance Estimation Algorithm from QA pairs.

```
Input: A Question-Answer (QA) Pair
Output: Multicomponent Answerer
        Performance (AC,QC,UC,CC)
initialization
$synCatSet[] \leftarrow synCat_1 \ldots synCat_n$
$answerSynTree \leftarrow buildParseTree(A)$
$list \leftarrow answerSynTree.nodeList()$
 $Iterator < Tree > it \leftarrow list.iterator()$
$synCatMap \leftarrow null\ synCatCount \leftarrow null$
while (it.hasNext()) do
    $syntacticUnit \leftarrow it.next()$
    if $syntacticUnit! = isLeaf()$ then
        $tag \leftarrow syntacticUnit.label()$
         $tag \leftarrow tag.split(``-")[0]$
        if $synCategoryMap.has(tag)$ then
            $count \leftarrow synCatMap.get(tag)$
             $int\, j \leftarrow count.intValue()$
             $j \leftarrow j+1$
             $synCatMap.put(tag, newInteger(j))$
        else
            $synCatMap.put(tag, newInteger(1))$
    if $syntacticUnit.isLeaf()$ then
        $leafTag \leftarrow syntacticUnit.label()$
         $leafTag \leftarrow tag.split(``-")[0]$
        if $isFirstPSingular(leafTag)$ then
            $firstPS \leftarrow firstPS + 1$
        if $isFirstPPlural(leafTag)$ then
            $firstPP \leftarrow firstPP + 1$
for k = 0 to synCategorySet.length do
    $occur \leftarrow synCatMap.get(synCatSet[k])$
     $tagName = syntacticCatSet[k]\ Zero \leftarrow 0$
     if (occur != null) then
        $synCatCount.add(occur.intValue())$
    else
        $synCatCount.add(Zero)$
$compScore \leftarrow \sum_{i=1}^{x} countCompletenessTag_i$
$descScore \leftarrow \sum_{i=1}^{y} countDescriptivenessTag_i$
...// compute the remaining
   expertise dimensions similarly
$authScore \leftarrow firstPP + (2/(firstPS+1))$
 $answerCom(AC) \leftarrow$
 $compScore + \cdots + authScore$
 $userCom(UC) \leftarrow$
 $answeringRate + \cdots + repScore$
 $questionCom(QC) \leftarrow$
 $qScore + \cdots + answCount$
$communityComp(CC) \leftarrow$
 $ansView + \cdots + qView$
return performanceScore<AC,UC,QC,CC>
```

4.2 Extracting Linguistic and Non-linguistic Features

4.2.1 A Phrase-structured Feature Set

Once generating syntactic parse trees through constituency parsing, syntactic information important to define the identified expertise dimensions are extracted. The Stanford shift-reduce parser (SRP) (Zhu et al., 2013) along with the Stanford CoreNLP toolkit has been used for linguistic information annotation, constituency and dependency parsing. In each tree (possibly a forest) representing an input answer, leaf and non-leaf nodes form/constitute syntactic categories. Phrasal categories include larger syntactic units (e.g., NP (noun phrase), VP (verb phrase)) constructed from two or more smaller syntactic units (lexical categories). Lexical categories represent part-of-speech tags (POS) (e.g., VB (verb), NN (noun)), relatively smaller than phrasal categories. Functional categories consist smallest syntactic units (e.g., DT (determiner), IN (preposition)) which link other larger syntactic information together. To reinforce and enrich the syntactice features, punctuation marks along with special characters and character encodings are added in our linguistic information sets.

4.2.2 A Dependency Relations or Semantic Feature Set

To extract dependency relations (aka universal dependencies), dependency parsing is run on training and evaluation sets. Subsequently, *head words*, and other dependencies (e.g., *nsubj, case, dobj*) present in the generated parse trees (graphs), are extracted.

Each depenendency type gives interesting facts how relations between words in answers' content are contrstucted and influence answer quality. For instance, high occurrences of the dependency type "paraxis" in the dependency structure of answers' sentences, might signal that the parsed sentences are constructed with clauses (phrases) without being connected with linking words that coordinate them. Such types of grammatical constructions might be frequently observed in low (possiby high) quality parsed answers (Woldemariam, 2020).

4.3 Non-linguistic (QA Pairs Meta) Features

While almost all posssibe meta-features available in QA are extracted, we give a particular attention for those competence-oriented features that support the syntactic information to best define each competence component. The extracted features are grouped together into four competence components. **Answer related features** (e.g., answer scores) define the answer competence component (AC). **Question related features** characterize answer quality and define answerers' performance in terms of question quality, question related features, e.g., askers' reputation and badge. **User related features** include reputation, answerers' answering rate and answer acceptance rate and so on. Also, we attempted to turn users' descriptions (**About Me**) and their associated URLs to their proffessional pages, into an important aspect of user-competence information called **credibility**. **Community related features** (e.g., view/favorite counts) assess the community feedback provided for answerers and define the community competence component (CC).

4.4 Answer Quality Models Training, Validation and Evaluation

Following preliminarly experiments on machine learning classifiers, support vector based logistic regression (SVMLR) has been found to be suitable for both learning answer quality from the selected linguistic features and very much sensitive for the incorparated competence information. As a result, almost 28 statstically valid binary classifiers have been built and evaluated on 13 different testsets, which results in the total of 364 (28*13) evaluations.

Both standard and non-standard evaluation metrics have been used to measure the validitity and performance of the built logistic regression models. For the former case, ($p-value$), accuracy, F-measure, recall and precision are considered, for the later case, a number of measures (e.g., the number of testsets improved) important for filterning and ranking the models are computed. Furthermore, R-squared values are computed to measure the power of the selected model features to explain answer quality.

4.4.1 Model Training and Significance Tests

The selected SVM based logistic regression classifier has been iteratively trained on largely dependency relations (the semantic feature set). That results in the **semantic baseline (SB) model**. Following the validation and model optimization phase with the development set, on top of the SB model, various all possible combinations of competence information is added to build competence based answer quality models.

The validitity of all models is checked and found to be statstically significant. The validity measurement has been perfomed using the stanadard **null-hypothesis (H_0)) test** using **T-test** (Alexopoulos, 2010) as shown in **Equation 1**. That ensures that answer quality predictions performed by the trained models is not by a random chanance, and the selected syntactic and semantic information has significant relationship with answer quality. Given that there are two equally distributed classes of answer quality (best or true positive and non best or true negative answers) to be determined by the models for any input answer, a valid model is basically expected to exceed a 50% mean accuracy for both classes. Other relevant statistical parameters (e.g., standard deviation, degrees of freedom (df)) and the size of the development set are also considered by the T-test equation, which is defined in. The resulting t-scores, in turn, are transformed into the corresponding $p-values$, and compared with the set threshod alpha value, that is mostly 0.05.

$$t-score = \frac{meanAccuracy - \mu_0}{\frac{standardDev}{\sqrt{answerCountDevSet}}} \qquad (1)$$

Equation 1 computes $t-scores$ from four variables: the known (assumed) (μ_0) and actual mean accuracy (*meanAccuracy*), standard deviation (*standardDev*) and the size of the development set (*answerCountDevSet*).

4.4.2 Evaluation Results and Discussions

Results from 364 models' performance evaluations, are summarized and ranked. Nearly perfect answer quality prediction models (100% recall and 99% precision scoring), in terms of recall and precision, are filtered and presented in Table **1**. In addition to that, those models which have completely improved the baseline results across all testsets, in terms accuracy and F-measure, are shown in Table **2**. Integrating the **user-competence estimation algorithm** into the answer quality assessment models significantly enhanced their prediction accuracy and F-measure. Adding the computed user-competence on top of the baseline semantic model significantly improved the F-measure of **all testsets' results**. Most importantly, *true positives* (high-quality answers) have been almost perfectly (nearly 100%) detected in many test cases. That implies, no best-answer has been misclassifed as *false negative*, in such scenario.

Given that almost all possible combinations of competence components have been employed to generate various answer quality models, our evaluation aims to explore which competence components yeild significantly better improvements over the semantic baseline model.

Table 1: Ordered Lists of Competence Based Answer Quality Prediction Models.

Recall				**Precision**			
Model	**Max**	**Model**	**Mean**	**Model**	**Max**	**Model**	**Mean**
AV+4C[2]	**1.00**	QC+UC+CC	**0.98**	AV+AC+2C2[3]	**0.98**	AV+QC+UC	**0.79**
QC+UC+CC	**1.00**	AV+AC+CC	**0.98**	AV+QC+UC	0.89	SB	0.76
AV+QC+CC	**1.00**	AV+AC+2C2	**0.98**	QC+CC	0.87	AV+QC	0.74
AV+AC+2C[4]	**1.00**	AC+UC	**0.98**	SB	0.86	QC+CC	0.71
AV+AC+CC	**1.00**	AV+UC+CC	0.97	AV+QC	0.85	UC+CC	0.71
AV+AC+2C2	**1.00**	AV+AC+2C	0.97	UC+CC	0.84	AC+QC+UC	0.70
AC+UC	**1.00**	AV+AC+2C3[5]	0.96	AC+QC+UC	0.83	AC+CC	0.69
AV+UC+CC	**1.00**	AV+QC+CC	0.96	AC+CC	0.82	AC	0.69
AV+AC+QC	**1.00**	AV+AC+QC	0.96	AC	0.82	AV+CC	0.68
QC	0.99	AC+QC+CC	0.96	CC	0.81	CC	0.68
AV+AC+2C3	0.99	QC	0.95	AV+CC	0.81	AC+QC+2C3	0.64
AV+AC+UC	0.99	AV+AC+UC	0.93	AC+QC+2C3	0.78	AC+UC+CC	0.64
AC+QC+CC	0.99	AV+4C	0.93	AC+UC+CC	0.78	AC+QC	0.63

Table 2: Semantic Baseline and Competence Enhanced Models' Evaluation Results in Accuracy.

	QA Forum Test set with Accuracy Scores											
Model	BL	CR	WP	MA	GA	DR	SE	WM	AP	WA	SU	SP	SF
SB	0.62	0.61	0.61	0.61	0.74	0.53	0.85	0.66	0.57	0.57	0.69	0.46	0.57
SB+AV+QC	**0.64**	0.66	0.58	**0.62**	**0.76**	0.56	**0.86**	0.69	0.58	0.60	0.71	0.54	0.61
SB+UC+CC	**0.64**	0.65	0.63	**0.62**	0.75	0.57	0.85	0.68	0.61	0.59	0.70	0.55	0.61
SB+AC+CC	**0.64**	0.65	0.65	**0.62**	0.72	0.60	0.80	0.69	0.64	0.65	0.74	0.57	0.61
SB+AV+CC	**0.64**	**0.68**	0.68	0.61	0.72	0.71	0.83	**0.70**	0.66	0.65	0.75	0.65	0.67
SB+CC	0.63	0.67	0.66	0.61	0.71	0.65	0.78	**0.70**	0.64	0.63	0.75	0.61	0.65
SB+AC	0.63	0.64	0.63	0.61	0.73	0.58	0.83	0.67	0.61	0.63	0.70	0.54	0.60
SB+AC+QC	0.63	0.66	**0.69**	0.59	0.64	**0.72**	0.74	0.66	**0.71**	**0.71**	**0.76**	**0.69**	**0.71**

Among the competence added answer quality models, **SB+QC+UC** and **SB+AC+QC** with **SB+AV+AC** yeild the maximum accuracy shift of 24% and 23%, respectively. That shows, from any other competence components, the answer competence component appears to give better results, as it gets mixed with the other other possible combinations. On the other hand, the joint user-question competence component (**SB+QC+UC**) gives both the maximum F-measure shift (i.e., 52%) and the best F-measure score (i.e., 84%).

Besides the achieved results, looking into the methods and model construction details of many answer quality prediction classifiers, they are characterized by unbalanced (regading the ratio of best to non-best answers), limited amount of data and features have been used. In comparison with related works, while our results seem to be better than prediction accuracy scores acheived in (Shah and Pomerantz, 2010; Adamic et al., 2008; Burel et al., 2012), more works need to be done as comared to (Cai, 2013; Suggu et al., 2016). Although authors in (Calefato et al., 2016) acheived better results than ours, surpisingly enough, considering evaluations on out-domain datasets, we acheived upto 85% accuracy, that outperforms the answer quality models trained in (Calefato et al., 2016).

5 ALGORITHM ANALYSIS

5.1 Algorithm Analysis with Single Competence Components

To make a simple observation of how the answer component compares with others in terms of the power of improving the baseline results, each competence component has been added on top the semantic baseline and evaluated iteratively. From any other competence components, adding the answer component results in the best accuracy value of 83%. On average, it improves the mean accuracy provided by the baseline semantic i.e., 62% to 65%. Regarding the number of testsets' results improved, the answer component enhances all test sets' results, F-measure wise, and gives the maximum number of improvements accuracy wise (i.e., 9 out of 12) next to the community component.

Considering other important measures, for instance, the maximum and the net (mean) accuracy improvement, the average (**SB+AV**) and the community competence components, respectively outperform others. While the average competence component gives an improvement of 23% over the semantic baseline, the community component gives the net improvement of 6%. F-measure wise, the question component gives the maximum improvement of 53% and net harvest of 28%.

The results from such simple observations, particulary regarding the number of testsets' improved, of single competence component based models lead to choose the answer compenent due to its distinctive role of discriminating best answers from non-best answers and singnificantly enhancing the baseline semantic model. Yet, considering the average competence component, perhaps be a good preference to acheive the maximum accuracy shift.

5.2 Algorithm Analysis with Two Competence Components

We attempted to further understand the impact of the integrated competence components and clearly identify which combinations lead to better results. To acheive that we added one more competence component either on top of the answer or average component, as well as joining the remaining components together (e.g., QC with UC or CC). For every model trained on a pair of competence components, five different accuracy and F-measure related values have been calculated as measures of effectiveness in predicting answer quality. That followed by ranking the models based on the resulting values. For instance, looking at the mean accuracy difference gives an order list of the first best models, **SB+QC+UC** and **SB+AC+QC**, and the least two, **SB+QC+CC** and **SB+AC+UC**.

Accuracy-wise, as the average competence gets combined with the question competence component, it triumphs over all types of dual based competence combinations, as it yeilds the maximum accuracy value of 86%. Its impact continues in improving the largest number of testsets' results, with **SB+UC+CC**, **SB+AV+QC** equally transform the accuracy values of the 92% of the testsets. In turn, as the question and user components get joined together, they provide the maximum and the best mean accuracy differences, 24% and 12%, respectively, from the semantic baseline's result.

F-measure-wise, **SB+QC+UC** gives the best mean and maximum values, 84% and 72%, respectively. Also, 100% of the testsets' baseline's results have been equally improved by such competence combination and others (e.g., **SB+AC+CC**, **SB+AV+CC**). Surprsingly enough, as the answer and average compentce components are used together, they results in a largest F-measure difference of 52%. However, looking at the net F-measure difference, the answer competence component gives better results as it gets mixed with the question component than the average component.

5.3 Algorithm Analysis with Three or More Competence Components

The Recall and Precision table, Table **1**, cleary illustrates the impact of joining three or more competence components together in detecting true positives (best answers). The table summarizes the order list of top models selected based their maximum and mean recall and precision. For instance, looking at the ordering of the models based the maximum recall, while about 67% of the models give a recall of 100% and the remaining score 99% recall. Among such high recall scoring models, while just only two of them, **SB+AC+UC** and **SB+QC**, have been built on less than three competence components, all the five competence components have been combined together to generate the **SB+AV+4C** model. Although the semantic baseline model does not appear on the recall based order list, it takes the second and the fourth position in the mean and the maximun precision based ranking, respectively.

Noting that the majority cases of best answers being detected perfectly with *no false negatives* implies that, likely integrating multiple aspects of user-competence gives better results than single (few) competence component based models. Nevertheless, the competence models also seem to be challenged by false positives (non-best answers) to a certain extent in comparision to their ability to discriminate *true positives*. It has been also observed that some components have an exceptionally outstanding role than others as they constantly occur on the considered lists. For instance, the user component, **UC** seems to frequently appear to give best results. The implication is the user related features, particularly answerers-related attributes have great impacts on the overall estimated competence and the achieved answer quality prediction, as also evident in other many related studies on answer quality(Blooma et al., 2008; Shah and Pomerantz, 2010; Woldemariam, 2020).

6 CONCLUSIONS

The contribution of this study is threefolds: the development of an algorithm that estimates answerer performance, the development of answer quality prediction models and the integration of this algorithm into answer quality prediction models. As a result, we proved that incorporating answerer competence information and looking deeply into answer content than using meta-features significantly improves the performance of answer quality assessment methods. That is evident from our evaluation results, which yeild upto **86% accuracy** and **84% F-measure**. Also, answer quality classifiers yeild upto **100% recall** and **98% precision**. In the future, it would be interesting to customize expertise dimensions and extend our method to enhance question quality assessments.

REFERENCES

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

Alexopoulos, E. C. (2010). Introduction to multivariate regression analysis. *Hippokratia*, 14 Suppl 1:23–8.

Blooma, M. J., Chua, A. Y., and Goh, D. H.-L. (2008). A predictive framework for retrieving the best answer. In *Proceedings of the 2008 ACM symposium on Applied computing*, pages 1107–1111.

Bryan, S. and Robert, H. (2000). Analyzing cockpit communications: the links between language, performance, error, and workload. *Human Performance in Extreme Environments*, 5(1):63–68.

Burel, G., He, Y., and Alani, H. (2012). Automatic identification of best answers in online enquiry communities. In *Extended Semantic Web Conference*, pages 514–529. Springer.

Cai, Y. (2013). Answer quality prediction in q/a social networks by leveraging temporal features. *International Journal of Next-Generation Computing*, 4(1).

Calefato, F., Lanubile, F., and Novielli, N. (2016). Moving to stack overflow: Best-answer prediction in legacy developer forums. In *Proceedings of the 10th ACM/IEEE international symposium on empirical software engineering and measurement*, pages 1–10.

Chen, Y., Wrenn, J. O., Xu, H., Spickard, A., Habermann, R., Powers, J. S., and Denny, J. C. (2014). Automated assessment of medical students' clinical exposures according to aamc geriatric competencies. *AMIA ... Annual Symposium proceedings. AMIA Symposium*, 2014:375–84.

Harper, F. M., Raban, D., Rafaeli, S., and Konstan, J. A. (2008). Predictors of answer quality in online q&a sites. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 865–874.

Li, L., He, D., Jeng, W., Goodwin, S., and Zhang, C. (2015). Answer quality characteristics and prediction on an academic q&a site: A case study on researchgate. In *Proceedings of the 24th international conference on world wide web*, pages 1453–1458.

Molino, P., Aiello, L. M., and Lops, P. (2016). Social question answering: Textual, user, and network features for best answer prediction. *ACM Transactions on Information Systems (TOIS)*, 35(1):1–40.

Shah, C. and Pomerantz, J. (2010). 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.

Suggu, S. P., Goutham, K. N., Chinnakotla, M. K., and Shrivastava, M. (2016). Deep feature fusion network for answer quality prediction in community question answering. *arXiv preprint arXiv:1606.07103*.

Tausczik, Y. and Pennebaker, J. (2010). The psychological meaning of words: Liwc and computerized text analysis methods. *Journal of language and social psychology*, 29(1):24–54.

Tausczik, Y. and Pennebaker, J. (2011). Predicting the perceived quality of online mathematics contributions from users' reputations. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1885–1888.

Woldemariam, Y. (2020). Assessing user reputation from syntactic and semantic information in community question answering. In *Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020)*, pages 5385–5393.

Woldemariam, Y., Bensch, S., and Björklund, H. (2017). Predicting user competence from linguistic data. In *Proceedings of the 14th International Conference on Natural Language Processing (ICON-2017)*, pages 476–484.

Woldemariam, Y. D. (2021). Expertise detection in crowdsourcing forums using the composition of latent topics and joint syntactic–semantic cues. *SN Computer Science*, 2(6):1–28.

Zhu, M., Zhang, Y., Chen, W., Zhang, M., and Zhu, J. (2013). Fast and accurate shift-reduce constituent parsing. In *ACL*.

Zhu, Z., Bernhard, D., and Gurevych, I. (2009). A multi-dimensional model for assessing the quality of answers in social q&a sites. In *ICIQ*.