Can models learned from a dataset reflect acquisition of procedural knowledge? An experiment with automatic measurement of online review quality

Martina Megasari
Pandu Wicaksono
Chiao Yun Li
Clément Chaussade
Shibo Cheng
University of Tours, France
firstname.lastname@etu.univ-tours.fr

Nicolas Labroche
Patrick Marcel
Verónika Peralta
University of Tours, France
firstname.lastname@univ-tours.fr

ABSTRACT

Can models learned from a dataset reflect how good are humans at mastering a particular skill? This paper studies this question in the context of online reviews writing, where the skill corresponds to the procedural knowledge needed to write helpful reviews. To this end, we model the quality of a review by a combination of various metrics stemming from text analysis (like readability, polarity, spelling errors or length) and we use customer declared helpfulness as a ground truth for constructing the model. We use Knowledge Tracing, a popular model of skill acquisition, to measure the evolution of the ability to write reviews of good quality over a period of time. While recent studies have tried to measure the quality of a review and correlate it to helpfulness, to the best of our knowledge, our work is the first to address this question as the exercise of a reviewer's skill over a sequence of reviews. Our experiments on a set of 41,681 Amazon book reviews show that it is possible to accurately assess the individual skill acquisition of writing a helpful review, based on a statistical model of the procedural knowledge at hand rather than human evaluations prone to subjectivity and variations over time.

1 INTRODUCTION

In today's era of big and open data, plenty of datasets are analyzed to derive models mimicking humans by using machine learning techniques. The representation and assessment of user knowledge opens new possibilities for big data analytics, as differentiating among novice and expert users, taking advantage of user experience for recommending (e.g. products or actions), calculating advanced scores (e.g. credibility), assessing the quality of users' analysis, etc. In this paper we focus on the assessment of procedural knowledge from large data collections.

Procedural knowledge is the knowledge about how to do something. Different from declarative knowledge, that is often verbalized, application of procedural knowledge may not be easily explained [3]. Models exists to evaluate procedural knowledge acquisition, like for instance the popular Bayesian Knowledge Tracing [6].

Many open datasets illustrate the application of procedural knowledge. For instance, Amazon review datasets like those provided by He and McAuley [12] contain customer written reviews, where the skill of writing helpful reviews is an example of application of procedural knowledge. However, this skill is difficult to define and assess. Reviews can be voted helpful or not by customers, but this assessment is subjective and as such subject to variations over time, and it is difficult to construct a model that accurately predicts helpfulness of a review [16].

In this paper, we show that it is possible to benefit from such very large datasets to learn an individual model of procedural knowledge acquisition. The resulting model of knowledge has several nice properties: (1) it is not prone to the usual bias caused by a single small set of evaluators that might be non representative or produce a subjective evaluation, (2) it avoids defining explicitly the procedural knowledge at hand that is replaced by a statistical model learned over the large dataset. As a consequence, the larger the dataset, the more accurate is the modeling of the procedural knowledge, and the better the evaluation of the skill for a user is.

To illustrate this, we experiment a use case with a dataset of the aforementioned Amazon on-line product reviews. We chose this use case because it is prototypical of how procedural knowledge influences decision making. For instance, Mayzlin and Chevalier studied the effects of on-line book reviews of Amazon.com and Barnesandnoble.com and found positive correlation between the reviews and the transactions of the book [4]. This means that the reviewers opinion play an important role in users' decision on the transaction. Automatic measurement of the reviewer skill may be beneficial to predict how helpful the review is. A skillful writer is assumed to be able to write a good review, which can help the customer to make a better decision on the transaction.

To motivate our approach, suppose that we want to determine whether a reviewer is assumed to master the skill of writing helpful reviews. This is preferable to trying to predict helpfulness of the reviews, because of the high variability of the reviewer profiles, reviews and votes received by reviews. However this skill corresponds to procedural knowledge and it is difficult to define. Therefore to evaluate the skill of each reviewer, we use the classical Knowledge Tracing model. But instead of using the Knowledge Tracing directly over the votes received by reviews, we apply it over a model of helpfulness learned from each review. Our research question is: can this model of helpfulness be used to assess the skill accurately? Consider the 4 curves displayed in Figure 1. These curves are related to the evolution over time of the skill of writing helpful reviews of a particular reviewer (randomly extracted from the Amazon book review dataset). The *helpfulness* curve is the normalized score of helpfulness received by the 20 reviews written by this reviewer. The *model* curve is the helpfulness score as predicted for this reviewer by a model learned over the entire dataset. The *KT helpfulness* curve predicts

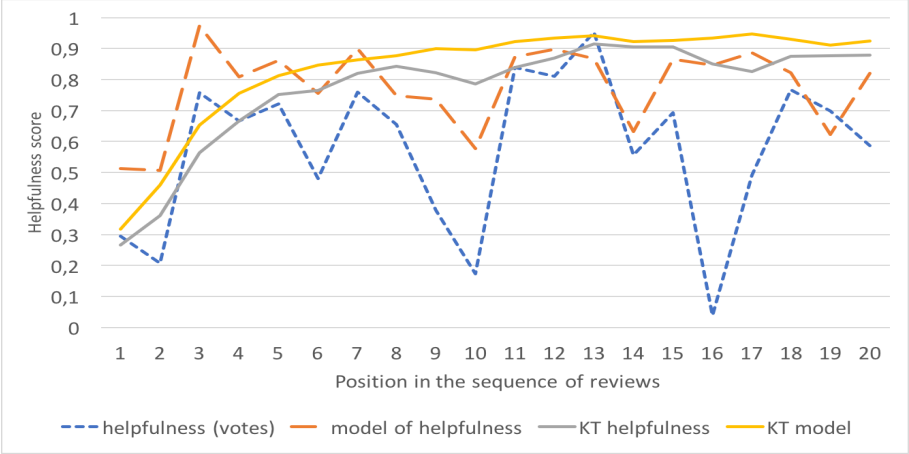

Figure 1: Evolution of helpfulness for a reviewer and different models of it

the probability that this reviewer has acquired the skill of writing helpful reviews, computed with the helpfulness score. The *KT model* curve is the same probability computed with the model. On this example, it is obvious that even though the skill can be considered acquired, helpfulness score is difficult to predict due to subjectivity of the voters. On the other hand, a model of helpfulness can be learned to predict if the skill has been acquired.

The contributions of this paper are the following: (1) assuming that writing helpful reviews is a hard to define skill, we propose a model for it. We use low level features of the on-line review such as rating, spelling error ratio or readability score to build the model that infers a high level and human-related feature which is helpfulness. This model is learned over the entire dataset and can be used to predict the helpfulness of future reviews for one particular reviewer. (2) Using Knowledge Tracing, we show that this model can be used to assess skill acquisition without relying on human entered votes. In particular, we show that this model, although learned over the entire dataset, is accurate enough to predict if the skill is acquired by each individual reviewer. To the best of our knowledge, this work is the first to evaluate a reviewer's skill over a sequences of reviews with Knowledge Tracing.

The remainder of the paper is organized as follows. Section 2 discusses related works. Section 3 defines the features used to build the model of helpfulness. Section 4 details our approach. Section 5 explains how the experiment is performed to build the model and exposes the results. Finally, Section 6 concludes the paper and discusses some possible future work.

2 RELATED WORKS AND BACKGROUND

We first review recent works on online review evaluation and then describe the Bayesian Knowledge Tracing model and some of its extensions.

2.1 Online review evaluation

Readability tests play an important role in online review evaluation. Various indexes have been proposed to quantify readability of an English text. Most of these indexes are related to the level of studies a person needs to understand the text at the first reading, according to American standard. They are computed considering the number of words, number of sentences, number of syllables or number of characters as components. The Gunning-Fog Index (FOG) [10] aims to estimate the years of formal education a person needs to understand the text during the first reading. The Flesch Reading Ease (FK) [15] indicates the difficulties of a text using the number of words, number of sentences and number of syllables. Higher values indicate better readability. The Automated Readability Index (ARI) [23] measures the approximate representation of the US grade level needed to understand the text. The Coleman-Liau Index (CLI) [5] is the approximation of US grade level needed to understand the text. More background on readability tests can be found in [16].

Previous works have studied the evaluation of online reviews due to the popularity of online marketing nowadays. Authors often pay attention to the influence of online reviews on helpfulness. Korfiatis et al. investigated the interplay between helpfulness, rating score and qualitative characteristics of the review text of 37,221 online reviews collected from Amazon UK during March to April in 2008 [16]. The authors theorize that helpfulness relates to a model with three aspects: conformity (relation between the review text and the rating), understandability (readability of the review text) and expressiveness (length of the review text). The authors formulate several hypotheses and perform linear regression to validate the relationship between the metrics derived from reviews and the helpfulness of the reviews. Regarding understandability, four common readability scores - indicating the education level the readers need to have in order to understand the content - are computed: FOG, FK, ARI and CLI. Their results indicate that helpfulness of a review is directionally affected by its qualitative characteristics and in particular by review text readability. Precisely, the relationship between reviews with average length and their readability scores holds for both moderate and extreme reviews. In addition, readability has more impact on the length of the reviews. In their work, metrics related to polarity, summary text of reviews and rating deviation (between the average rating and the reviewer's one) are not considered. Moreover, due to the purpose of the work, the books having special offers are not considered to avoid the price effect. In our work, such books are chosen due to the amount of reviews resulting from this price effect.

Based on the 7,659 book reviews on Amazon UK, Wu et al. explored whether a negative bias exists in terms of evaluating the helpfulness [27]. The assumption was that negative reviews may be more helpful than positive ones. After applying a regression

model controlling factors such as readability and length of the reviews, the result shows that the assumption is not yet readily applicable to online reviews.

Mudambi and Schuff analyzed 1,587 reviews from Amazon.com [19] to understand how review extremity, review depth and product type affect the perceived helpfulness of the review. Their helpfulness model is based on features rating, review text word count, total votes and product type. Product type is either Experience goods or Search goods, where Experience goods are products that require sampling or purchase in order to evaluate product quality. Books are examples of experience goods. They found that for experience goods, moderate reviews are more helpful than extreme reviews (whether they are strongly positive or negative). In contrast, it has been observed that reviews closer to the general opinion of people (average rating score) may be considered more helpful by the potential buyers [14].

Mc Auley and Leskovec [18] propose a latent-factor model for recommending products that may be preferred by the users according to their experience level at the moment. The model evaluates the evolution of users' experiences and is based on the rating that users give to products. Unlike other works on temporal dynamics, which rely on the hypothesis that two users rating a product at the same time will provide the same rating, Mc Auley and Leskovec's model takes users' personal development into consideration in order to evaluate the expertise degree of the reviewers. Experiments showed for example that experts' ratings are easier to predict and are more similar to each other. While close to our work in the idea of taking the evolution of the user into account, this work focuses on ratings and not helpfulness, and therefore does not consider the linguistic aspect of review text.

Liu et al. considered a complex model learned using non-linear regression, that combines the reviewer's expertise (based on the number of similar reviews written in the past), the writing style of the review (characterized with part of speech tagging and counting the number of words in each tag), and the timeliness of the review [17]. They showed that the three factors predict accurately helpfulness, over a dataset of 22,819 reviews collected from IMDB.

In [26], review helpfulness is considered through five features including user profile aspects (age, verified purchase) together with rating, text length and the rank of the review in the webpage. A model learned on 12,756 reviews was shown to be reasonably robust.

Agnihotri and Bhattacharya explored how the helpfulness of online reviews is affected by content readability (FK Index), sentiment analysis and the number of reviews written by a reviewer [1]. It was observed on 1608 Amazon reviews that the content readability and text sentiment of the reviews follow curvilinear relationship with review helpfulness. Reviews whose readability score are very high or sentiment are very good would be perceived less helpful.

Hong and Xu analyze the impact of review message and reviewer profile on the helpfulness of 2997 online reviews collected from Douban.com [13]. Using negative binomial regression, the authors show that reader participation is positively related to online review helpfulness; Reader participation fully mediates the effect of reviewer expertise history on online review helpfulness and partially mediates the effects of other three metrics: average rating, title depth and reviewer network centrality.

To the best of our knowledge, no work ever focused on the evolution of the quality of review text under the angle of skill acquisition, with a model learned only on the review content.

2.2 Knowledge Tracing Models

The Bayesian Knowledge Tracing model was proposed by Corbett and Anderson, using Bayesian network to assess people's procedural knowledge acquisition or simply put "skill level" [6]. An individual's grasp of the procedural knowledge is expressed as a binary variable expressing whether the corresponding skill has been mastered or not. The knowledge of an individual cannot be directly observed, but it can be induced by observing the individuals' answers to a series of questions (or opportunities to exercise the skill) in order to guess probability distribution of knowledge mastering. Observation variables are also binary: the answer to the question is either correct or wrong.

Specifically, the Knowledge Tracing model has four parameters, namely, two learning parameters, $P(L_0)$ and $P(T)$, and two performance parameters, $P(G)$ and $P(S)$. $P(L_0)$ is the probability that the skill has been mastered before answering the questions. $P(T)$ is the knowledge transformation probability: the probability that the skill will be learned at each opportunity to use the skill (i.e., the transition from not mastered to mastered). $P(G)$ is the probability of guess: in the case of knowledge not mastered, the probability that the individual can still answer correctly. $P(S)$ is the probability of slip, i.e. to fail while the skill is already mastered. The model uses these parameters to calculate the learning probability after each question to monitor individual's knowledge status and predict their future learning probability of knowledge acquisition using a Bayesian Network.

The probability that a skill L at question $i+1$ is mastered, denoted $P(L_{i+1})$ is the sum of two probabilities: (1) the posterior probability that the skill was already learned, contingent on the evidence at time i, i.e. the i^{th} opportunity to evaluate the skill, that can either be *Correct* or *Incorrect*, and (2) the probability that the knowledge changes from not mastered to mastered at the i^{th} opportunity. It can be shown in the following formula:

$$P(L_{i+1}) = P(L_i|Evidence_i) + (1 - P(L_i|Evidence_i)) * P(T) \quad (1)$$

where:

$$P(L_i|Evidence_i = Correct) = \frac{P(L_i) * P(\neg S)}{P(L_i) * P(\neg S) + P(\neg L_i) * P(G)}$$

$$P(L_i|Evidence_i = Incorrect) = \frac{P(L_i) * P(S)}{P(L_i) * P(S) + P(\neg L_i) * P(\neg G)}$$

Due to its predictive accuracy, Corbett and Anderson's Bayesian Knowledge Tracing is one of the most popular models. However, several challenges, including local minimum, degenerate parameters and computational costs during fitting, still exist. Hawkins et al. proposed a fitting method avoiding these problems while achieving a similar predictive accuracy, and evaluated it against one of the most popular fitting methods: Expectation-Maximization [11]. In this extension, the parameters are fitted by estimating the most likely opportunity at which each individual learned the skill. Learner's performance is thus annotated with an estimate of when the skill is learned, assuming that a known state can never be followed by an unknown state. This annotation is used to construct knowledge sequences, that when

compared with the actual performance sequence allows to empirically derivate the model's four parameters.

As aforementioned, traditionally, the performance of an individual is presented in binary value, correct or wrong, which does not account for all the cases of skill learning situation. Wang et al. proposed to extend the Knowledge Tracing model by replacing the discrete binary performance node with continuous partial credit node [25]. In this extension, it is assumed that $P(G)$ and $P(S)$ follow two Gaussian distributions, that are described respectively by their means and standard deviations. Prediction of the performance node also follows a Gaussian distribution, in which the mean value is used for the prediction. Noticeably, the standard deviation contains the information of how good the prediction is. Experiments with this extension show that by relaxing the assumption of binary correctness, the predictions of an individual's performance can be improved.

These two improvements of the Knowledge Tracing model (in the fitting method and the use of partial credits) were used successfully in sequencing educational content to students [7]. We conclude this section by noting that other models exist for predicting a learner's skill. Specifically, Performance Factor Analysis [20] uses standard logistic regression with the student performance as dependent variable. Interestingly, it is shown in [9] that Knowledge Tracing can achieve comparable predictive accuracy as Performance Factor analysis. Finally, Deep Knowledge Tracing [22] uses Recurrent Neural Networks to model student learning, with the advantage of not having to set explicit probabilities for slip and guess. However these models need very large datasets to learn the latent state from sequences, and most importantly, the encoding of the input vectors depends on an upper bound on the number of exercises which does not directly fit our context.

3 FEATURES AND METRICS

Consistently with the previous work of Korfiatis et al. [16], our model of helpfulness is based on features that are grouped in three categories: Conformity, Understandability and Extensiveness, with additional features compared to [16]. We derive metrics, i.e., numerical attributes to be used in the definition of our model, from these features. Conformity expresses the consistency of a review being written. In addition to the classical rating, we add two metrics in this category: Polarity and Deviation. Understandability measures how good is the quality of the written text in terms of readability. We derived five metrics to measure the score: Spelling Error Ratio and 4 readability metrics (FOG, FK, ARI, and CLI). Finally extensiveness refers to the length of the review. In total, 16 metrics are defined, since length and readability metrics apply both to the review text and summary. We detail them below, a summary of the features used in the experiments with their name, category, theoretical and empirical range is provided in Tables 1 and 2.

Feature name	Category	Applies to	Range
rating	Conformity	all	[1, 5]
polarityReviewText	Conformity	text	[-1,1]
polaritySummary	Conformity	summary	[-1,1]
deviation	Conformity	all	[0,5]
reviewTextSER	Readability	text	[0,1]
summarySER	Readability	summary	[0,1]
reviewTextFOG	Readability	text	$\mathbb{R}^+$
summaryFOG	Readability	summary	$\mathbb{R}^+$
reviewTextFK	Readability	text	$\mathbb{R}$
summaryFK	Readability	summary	$\mathbb{R}$
reviewTextARI	Readability	text	$\mathbb{R}$
summaryARI	Readability	summary	$\mathbb{R}$
reviewTextCLI	Readability	text	$\mathbb{R}$
summaryCLI	Readability	summary	$\mathbb{R}$
reviewTextLength	Extensiveness	text	$\mathbb{N}^+$
summaryLength	Extensiveness	summary	$\mathbb{N}^+$

Table 1: Summary of the main features

Feature name	Min	Max	Mean	Std Dev.
rating	1	5	4.112	1.183
polarityReviewText	-0.875	0.875	0.027	0.052
polaritySummary	-0.875	1	0.029	0.137
deviation	0	3.786	0.452	0.615
reviewTextSER	0	0.5	0.009	0.008
summarySER	0	1	0.014	0.038
reviewTextFOG	0	740.8	13.983	8.45
summaryFOG	0	42.4	9.524	10.038
reviewTextFK	-1788.235	121.22	58.96	24.407
summaryFK	-1824.58	121.728	59.537	51.228
reviewTextARI	-6.837	919.088	11.41	10.374
summaryARI	-16.22	261.67	5.162	7.769
reviewTextCLI	-22.24	39.133	8.64	2.549
summaryCLI	-58.13	307.6	5.387	9.417
reviewTextLength	0	32669	1152.094	1261.787
summaryLength	1	257	28.875	16.786

Table 2: Empirical Values of Metrics

3.1 Conformity

Metrics in this category relate to the consistency of the review. As the content of a review consists in a rating and a written text, we can derive a relation between them. A rating should correspond to the written review and vice versa, hence difference between these two contents might indicate that the review is inconsistent. For example, a review having 5 stars rating and very negatively written is inconsistent. Needless to say, inconsistent reviews may lead to lower score of helpfulness due to the confusion it brings. From this perspective, we consider Polarity of the text, which indicates the positiveness or negativeness of a review as a metric. Besides, the extremity of the rating given by the reviewer may indicate that the reviewer is biased and has a subjective point of view on the product being reviewed. Extremely high and low rating is associated with lower levels of helpfulness than reviews with moderate rating [19]. In contrast, reviews closer to the general opinion of people (average rating score) may be considered more helpful by the potential buyers [14]. From this perspective, we derived the Deviation score, quantifying how much different the rating given by the reviewer is to the average rating.

Rating. The Rating of a review is the user input quantitative indicator of the quality of the item reviewed (e.g., rating is from 1 to 5 for Amazon Book Reviews).

Polarity. Polarity of a text is measured by using a word list that indicates the positivity, negativity and objectivity of each synset. Polarity score of a word with the part of speech is calculated as the score of the positivity subtracted by the score of negativity. The range of the value of polarity is between -1 and 1, -1 indicates that the written text is very negative and 1 indicates that the written text is very positive.

Deviation. Deviation is calculated as the absolute difference between the rating of a review and the average rating of the item reviewed.

3.2 Readability

Metrics in this category relate to the effort needed to understand the text of the review. This is measured based on the number of spelling errors in the written text, which is expected to be negatively correlated to helpfulness [8], and with various readability measures.

Spelling Error Ratio (SER). Spelling Error Ratio is the number of spelling errors divided by the text length.

Gunning-Fog Index (FOG). The FOG [10] aims to estimate the years of formal education (according to the American system) a person needs to understand the text during the first reading. This index uses the number of words, the number of sentences and the number of complex words to measure the years. A word is considered as a complex word if the word is using more than two syllables.

$$FOG = 0.4[(\frac{nbWords}{nbSentences}) + 100(\frac{nbComplexWords}{nbWords})] \quad (2)$$

Flesch Reading Ease (FK). The FK index [15] indicates the difficulties of a text using the number of words, number of sentences and number of syllables.

$$FK = 206.835 - 1.015(\frac{nbWords}{nbSentences}) - 84.6(\frac{nbSyllables}{nbWords}) \quad (3)$$

Automated Readability Index (ARI). The ARI [23] approximates the US grade level needed to understand the text. This index uses number of characters, number of words and number of sentences.

$$ARI = 4.71(\frac{nbCharacters}{nbWords}) + 0.5(\frac{nbWords}{nbSentences}) - 21.43 \quad (4)$$

Coleman-Liau Index (CLI). The CLI [5], Like ARI, is the approximation of US grade level needed to understand the text. This index also uses number of characters, number of words and number of sentences as components.

$$CLI = 5.89(\frac{nbCharacters}{nbWords}) - 0.3(\frac{nbSentences}{nbWords}) - 15.8 \quad (5)$$

3.3 Extensiveness

The textual part of the review consists of a text and a summary of this text. For both we measure the length in characters, respectively called Review Text Length and Summary Length.

4 METHODOLOGY

Our approach is divided into three phases: metric extraction, model construction and skill evaluation. These phases are detailed below.

4.1 Metric extraction and feature selection

In the first phase, we calculate for each review the scores for the metrics presented in Section 3, that we use to build the model of helpfulness. Then we apply feature selection to reduce the set of metrics by removing redundant ones, while avoiding losing too much information on the data set. We use a heuristic greedy method by calculating all the pairwise correlations between metrics. For those metrics that are highly correlated, only the ones highly correlated with the helpfulness score will be kept, the others being discarded. Finally, we normalize the scores in order to be independent of attribute ranges and units and highlight the actual importance of each attribute. We use Min-Max Scaling normalization strategy.

4.2 Model construction

We build our model to measure the quality of a review, where quality is defined by the helpfulness ratio of the review:

$$helpfulness = \frac{nbHelpfulVotes}{nbVotes} \quad (6)$$

where $nbHelpfulVotes$ is is the number of positive votes received by the review and $nbVotes$ is the total number of votes received by the review. This constitutes the class attribute value of a supervised machine learning method to build our simple model of helpfulness as a linear combination of the metrics. Thus, our predicted output variable $y \in \mathbb{R}$ will be expressed as a weighted sum of input features $x_i, \forall i \in [1, m]$, m being the number of features:

$$y = \sum_{i=1}^{m} \omega_i \times x_i + b \quad (7)$$

where $\omega_i \in \mathbb{R}$ is the weight reflecting the contribution of feature i to the overall decision and $b \in \mathbb{R}$ stands for the bias.

The intuition behind restricting our study to linear models is mainly for two reasons. First, these models are more simple and can be calculated more efficiently. Second, they allow for a direct interpretation of the contribution of each feature to the final helpfulness decision. To this end, we try a variety of methods and keep the one best fitting the dataset.

In our tests, error measurement is done using classical correlation coefficient, Efron's R^2, MAE and RMSE scores.

4.3 Skill evaluation

In this last phase, we apply Knowledge Tracing (KT) to sequences of reviews in order to estimate reviewers' skills. We proceed as follows: We group the reviews by reviewers, obtaining one sequence of reviews per reviewer. Each review is considered as an opportunity to learn the skill (i.e. being able to write useful reviews) and is graded with a score, representing the reviewer's performance (i.e. how useful is the review). We compute two KT scores: (i) directly from helpfulness ratings, and (ii) from the learned helpfulness model. In the former, the reviewer's performance is calculated as the helpfulness score of the review. In the latter, it is predicted by the helpfulness model. In both cases, the final score, output by KT model, expresses the probability that the skill is mastered by the reviewer.

We use the continuous version of KT described in [25] since the scores we will consider are continuous. In this extension of KT, $P(G)$ and $P(S)$ are assumed to follow a Gaussian distribution, and as such, they are represented by a mean value and a standard deviation. As a consequence, and opposed to binary KT, the prediction $P(L_n)$ also follows a Gaussian distribution, whose mean expresses the value of the prediction and whose standard deviation expresses the confidence attached to this prediction. To learn the 6 parameters of continuous KT, we extend the approach proposed by Hawkins et al. [11] so that it outputs estimates of $P(G)$ and $P(S)$ described by a mean and a standard deviation. Then, based on these 6 parameters, the estimation of each skill acquisition $P(L_n)$ is performed by running 100 tests, with randomly generated values for $P(G)$ and $P(S)$ following their respective distribution. From these 100 $P(L_n)$ estimates, we compute a mean and a standard deviation following the normal hypothesis.

However, the KT efficiency is known to be dependent on the granularity of skills that are fed to the model: generally, the more focused the skills, the better the prediction of skill acquisition. In this respect, it is possible to consider that each of the features that fed our linear predictive model of helpfulness can be considered as a sub-skill related to helpfulness. For this reason, we define two distinct tests to evaluate the learned model of helpfulness: In the first we simply use the output of the linear regression model as the predicted helpfulness for a review. In the second, we consider each feature metric as a possible sub-skill evaluation of the reviewer. We then learn as many KT models as there are features. In the end, we have the probabilities that sub-skills corresponding to each feature are acquired. These sub-skills scores are then aggregated into one single skill acquisition probability.

The global validation of our proposal is given by measuring the error between the KT based on real ratings, the KT based on the general linear model and the KT based on aggregated feature-based models. This error is evaluated by RMSE, which has been shown to be the strongest performance indicator for binary KT with significantly higher correlation than Log Likelihood and Area Under Curve [21].

5 EXPERIMENTS

Our implementation is done in Java 8, with Weka 3.8 for model learning. We used our own implementation of the knowledge tracing, whose code has been made available through Github[1] as one contribution of this paper. For polarity extraction, we use SentiWordNet [2], that lists the positivity, negativity and objectivity of each synset (set of synonyms). SentiWordNet provides the score of each word with the part-of-speech, hence we do POS tagging for each word using Stanford POS tagging library [24].

5.1 Dataset description

The dataset we use for experiments is Amazon Book Review Data provided by Julian McAuley from UCSD [12]. We select the book category in this dataset resulting in 22,507,155 total reviews.

As one of our goals is to measure the evolution of the ability to write reviews of good quality, we need to obtain for each reviewer a sequence of reviews long enough to observe that evolution. Therefore, we define reviewers with less than 30 reviews as not so active reviewers and filter them out. In addition, we only consider the reviews that have been scored by customers by means of votes (helpful review or not).

To confirm the hypothesis that few reviewers have written many reviews and that many reviewers have written few reviews, we plotted on Figure 2 the number of reviewers (on a logarithmic scale) by number of reviews, for reviewers with more than 30 reviews. Each points (x, y) in this figure represents that x reviewers have written y reviews. Furthermore, we found reviewers writing so much reviews that are dubious and possibly bias their reviews. For instance, reviewer of ID A14OJS0VWMOSWO wrote 43,201 reviews with an average score of 4.9991 out of 5. The reviewer received 240,262 votes, of which 199,573 are helpful. In our opinion, such reviewers introduce a bias in the dataset. Hence we limited our experiment and selected reviewers that have 30 to 50 reviews.

We calculate the score of each feature from the dataset and calculate their standard deviations, reported in the last column of Table 2. The standard deviation of the helpfulness, that varies in [0,1], is 0.32, which indicates that the score is quite spread out and the dataset has a wide enough variety, from helpful reviews and not helpful reviews. Moreover, the standard deviations of the features indicate that creating a model from this dataset is difficult.

[1] https://github.com/Cubiccl/Continuous-Knowledge-Tracing/releases/tag/1.0

5.2 Model construction

We now describe how the model of helpfulness is learned from the dataset. Consistently with [16] our model of helpfulness is constructed as a linear combination of the metrics extracted from the review text and summary. More precisely, as explained in Section 4, we use a linear classifier to learn a weight for each of the features introduced in the previous section, in order to understand its contribution to the helpfulness score. We tested three different approaches to learn the feature's weights: Linear Regression, Perceptron and Support Vector Machine with linear kernel. We used out-of-the-shelf Weka algorithms with 10-fold cross validation. Table 5 summarizes the results of those tests, for various size of dataset selected according to minimum number of votes for the reviews (918 reviews with number of votes being at least 200, up to 522804 reviews with at least 1 vote). Results for Perceptron and SVM are not reported for the largest dataset due to too much time consumption. The results show that linear regression achieves a good compromise of accuracy and computation time, with better accuracy on smaller datasets and better at handling larger datasets with no significant drop in accuracy. We therefore chose to work with linear regression in what follows.

5.2.1 Preprocessing. We recall that our definition of helpfulness is the number of helpful votes divided by the total number of votes, hence, a review with large number of votes is a genuine representation of helpfulness from a customer's point of view. But a review with only one vote, being a helpful one, can still obtain a maximum helpfulness score, which is not desirable. Filtering the dataset by number of votes becomes necessary. In order to find the appropriate minimum number of votes for each review, we iterated this parameter from 1 to 25 for the most important features of our model (i.e., after feature selection), and checked the results in terms of correlation and expressiveness (contribution of each metric), reported in Table 3. We decided to choose 2 datasets among those tested, based on, first, expressiveness (determined by the non zero value of coefficient in the linear model), and second, correlation coefficient (that indicates to which extent the model matches the dataset), for more than 10,000 reviews. The best interestingness and correlation coefficients were obtained for, respectively at least 12 votes and at least 23 votes. In this phase, we are not sure about the effect of these parameters on knowledge tracing model. Therefore, we keep two data sets, to see which can give a better result in knowledge tracing model. In what follows, the first dataset is called $minVotes = 12$ and consists of 41,681 reviews while the second dataset is called $minVotes = 23$ and consists of 11,083 reviews.

Using linear regression on the two datasets $minVotes = 12$ and $minVotes = 23$ results in the models described in Tables 6 and 7 respectively. The models constructed are evaluated with correlation coefficient, Efron's R^2, MAE and RMSE scores, reported in Table 8.

5.2.2 Feature selection impact. We then proceed to feature selection, as described in Section 4.1. As shown in table 8, our models before and after feature selection achieve very similar accuracy results. If efficiency in learning the model is an issue, or if the model should remain as simple as possible, one can

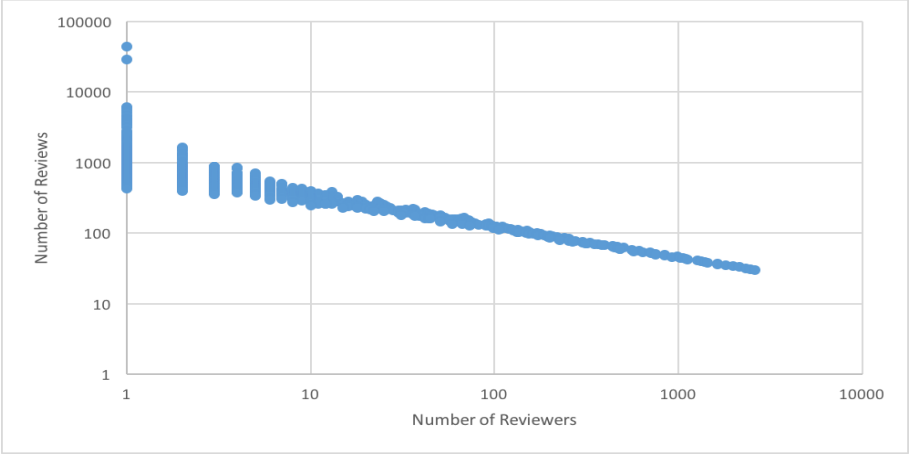

Figure 2: Number of reviews by number of reviewers

minVotes	Number of reviewers	Number of reviews	Correlation coefficient	Number of zero coefficients
1	13820	522801	0.3352	0
2	13556	350158	0.3917	0
3	11312	247394	0.4351	0
4	9060	184092	0.4649	0
5	7408	142572	0.4946	0
6	6304	115295	0.5174	0
7	5349	94482	0.5373	0
8	4586	78416	0.5544	0
9	3972	66216	0.5716	0
10	3453	56446	0.5834	0
11	3004	48278	0.5975	1
12	**2643**	**41681**	**0.6065**	**0**
13	2320	36058	0.6154	2
14	2041	31385	0.6245	2
15	1822	27596	0.6355	2
16	1616	24270	0.6415	2
17	1451	21519	0.6465	2
18	1302	19131	0.6551	2
19	1192	17274	0.657	2
20	1079	15489	0.6625	2
21	980	13899	0.6698	2
22	886	12451	0.6772	2
23	**793**	**11083**	**0.6804**	**2**
24	714	9940	0.684	2
25	655	9069	0.6897	2

Table 3: Correlation Coefficient for various minVotes

then safely decide to use the model learned on only the selected features. In what follow, we report the results for both sets of features.

A second lesson learned with our feature selection step is that, interestingly, for both datasets, the features selected include features that were not present in [16], namely spelling Error Ratio, polarity and deviation. With the notable exception of Summary Spelling Error Ratio, these features' weights remain steady, and in some cases relatively important, after feature selection. Quite surprisingly, ReviewTextSER has no impact on helpfulness, while as expected deviation highly contributes negatively to it.

5.2.3 Comparison with the state-of-the-art. As to model accuracy, Table 8 shows that the results we obtained are notably comparable, and in some cases slightly better than those reported

Metrics	$minVotes = 12$	$minVotes = 23$
rating	0.31117594	0.37121056
polarityReviewText	0.36708846	0.27873667
polaritySummary	0.05166703	0.08006764
deviation	-0.20847153	-0.1951008
reviewTextSER	0	0
summarySER	-0.28603436	-0.25242002
reviewTextFOG	-1.10263702	-0.40678142
summaryFOG	0	0.02506183
reviewTextFK	4.37638627	2.3020136
summaryFK	0.12251469	0.13780373
reviewTextARI	5.01873535	2.47411051
summaryARI	-0.4099729	-0.10677126
reviewTextCLI	0.31215745	0.21371702
summaryCLI	0.79694206	0.28470061
reviewTextLength	0.30807426	0.3431656
summaryLength	0	0.03837902
bias	-4.26391009	-2.21159487

Table 4: Coefficient of Linear Regression Model for $minVotes = 12$ and $minVotes = 23$

Algorithm	Dataset size	Exec. time	Correlation coefficient	RMSE score
Linear Regression	918	0.01	0.6455	0.2005
Perceptron	918	0.12	0.5071	0.2635
SVM	918	0.25	0.6352	0.218
Linear Regression	3414	0.02	0.7226	0.1957
Perceptron	3414	0.44	0.5135	0.2569
SVM	3414	6.39	0.7199	0.1992
Linear Regression	10971	0.02	0.6888	0.2023
Perceptron	10971	1.39	0.5349	0.2658
SVM	10971	101.87	0.6846	0.2062
Linear Regression	29808	0.04	0.6303	0.2064
Perceptron	29808	3.81	0.499	0.2401
SVM	29808	829.65	0.627	0.2119
Linear Regression	522801	0.67	0.3352	0.3028

Table 5: Test of 3 linear model algorithms on various datasets

$minVotes = 12$	Before	After
rating	0.31117594	0.31312877
polarityReviewText	0.36708846	0.3655654
polaritySummary	0.05166703	0.05351795
deviation	-0.20847153	-0.20951913
reviewTextSER	0	-0.03361242
summarySER	-0.28603436	-0.31027976
reviewTextFOG	-1.10263702	N.A
summaryFOG	0	-0.04014441
reviewTextFK	4.37638627	0.4228708
summaryFK	0.12251469	N.A
reviewTextARI	5.01873535	N.A
summaryARI	-0.4099729	N.A
reviewTextCLI	0.31215745	0.04970302
summaryCLI	0.79694206	0.40990694
reviewTextLength	0.30807426	0.3077809
summaryLength	0	0.03922442
bias	-4.26391009	-0.12802418

Table 6: Models of helpfulness before and after feature selection for $minVotes = 12$

$minVotes = 23$	Before	After
rating	0.37121056	0.37369313
polarityReviewText	0.27873667	0.28253483
polaritySummary	0.08006764	0.0821465
deviation	-0.1951008	-0.19656865
reviewTextSER	0	0
summarySER	-0.25242002	-0.29930955
reviewTextFOG	-0.40678142	N.A
summaryFOG	0.02506183	-0.021072
reviewTextFK	2.3020136	0.13767929
summaryFK	0.13780373	N.A
reviewTextARI	2.47411051	N.A
summaryARI	-0.10677126	N.A
reviewTextCLI	0.21371702	0
summaryCLI	0.28470061	0.13525824
reviewTextLength	0.3431656	0.34368253
summaryLength	0.03837902	0.07231388
bias	-2.21159487	0.13145667

Table 7: Models of helpfulness before and after feature selection for $minVotes = 23$

Evaluation Metrics	$minVotes = 12$	$minVotes = 23$
Total Number of Reviews	41681	11083
|Before feature selection|		
Correlation Coefficient	0.608	0.682
Efron's R^2	0.3697	0.4651
Mean Absolute Error	0.1521	0.1494
Root Mean Squared Error	0.2014	0.201
|After feature selection|		
Correlation Coefficient	0.6065	0.6804
Efron's R^2	0.3678	0.4629
Mean Absolute Error	0.1526	0.15
Root Mean Squared Error	0.2017	0.2014

Table 8: Evaluation of the models

in [16] on datasets of similar size (37,221 Amazon UK reviews were analyzed in that work). In that work, 3 models were constructed, and their fitness to the dataset was reported in terms of Efron's R^2 scores. Their three models obtained respectively 0.316, 0.354 and 0.451 while ours scores at 0.3697 for $minVotes = 12$ and 0.4651 for $minVotes = 23$ (the higher the better for the Efron's R^2). Importantly, their models incorporate the features number of votes and number of helpful votes, which we have deliberately not included in ours, since we aim at predicting helpfulness when no such socres are available.

Finally, the two datasets $minVotes = 12$ and $minVotes = 23$ achieve comparable MAE and RMSE, even though $minVotes = 23$ shows a better correlation coefficient or Efron's R^2. This illustrates the robustness of our model construction approach to larger but more skewed datasets.

5.3 Skill evaluation

In this section, we show that the model obtained can be used to accurately predict the learning of the skill of writing helpful reviews.

After training the Knowledge Tracing (KT) model as explained in Section 4.3 using a 10 fold cross validation, we acquire the average of the six parameters and the KT model RMSE scores. We also learn one KT per sub-skill and aggregate them to obtain a single probability, as explained in Section 4.3. To be consistent with the learning of the linear regression model, this aggregation is done with the weights learned for this model. The results are reported in Table 9 and Table 10. Each table shows the average skill acquisition probability ($mean(L_n)$) for the actual helpfulness skill, the helpfulness model and the aggregation of the sub-skills. We also report the parameters learned for the KT of the model.

	Scores	minVotes = 12	minVotes = 23
Actual skill	mean(L_n)	0.968337	0.960511
	variation(L_n)	0.025213	0.033238
Model	$P(L_0)$	0.007504	0.033457
	$P(T)$	0.030262	0.077669
	mean($P(G)$)	0.349992	0.369982
	variation($P(G)$)	0.007067	0.0147
	mean($P(S)$)	0.412574	0.412882
	variation($P(S)$)	0.016212	0.025905
	mean(L_n)	0.783885	0.800687
	variation(L_n)	0.090915	0.090820
Aggregated	mean(L_n)	0.999943	0.999991
	variation(L_n)	0.000584	0.000122
	a-mKRMSE	0.164619	0.156373
	a-AggKRMSE	0.064818	0.081964

Table 9: KT parameters, prediction and predictive accuracy before feature selection

For the sake of readability, we recall that RMSE scores are generated in three ways:

- RMSE as reported in table 8 represents the error between the helpfulness model scores and the actual helpfulness scores, without KT involved at that point.
- actual-model Knowledge RMSE (a-mKRMSE) represents the error between the KT of the actual helpfulness scores and the KT of the helpfulness as computed with the model.
- actual-Aggregated Knowledge RMSE (a-AggKRMSE) represents the error between the KT of the actual helpfulness scores and the aggregation of the KT scores of each feature taken independently (i.e., each sub-skill).

Before commenting the results of the tests, it is important to note that the average value of the helpfulness skill acquisition

	Scores	minVotes = 12	minVotes = 23
Actual skill	mean(L_n)	0.968026	0.961189
	variation(L_n)	0.02536	0.032609
Model	$P(L_0)$	0.004666	0.030936
	$P(T)$	0.026656	0.075609
	mean($P(G)$)	0.348688	0.369544
	variation($P(G)$)	0.007853	0.015195
	mean($P(S)$)	0.414712	0.406663
	variation($P(S)$)	0.014588	0.027189
	mean(L_n)	0.774606	0.804482
	variation(L_n)	0.090247	0.093722
Aggregated	mean(L_n)	0.96117	0.900541
	variation(L_n)	0.048756	0.055234
	a-mKRMSE	0.170865	0.156404
	a-AggKRMSE	0.062204	0.050704

Table 10: KT parameters, prediction and predictive accuracy after feature selection

probability (i.e., the value to be predicted) is high. We conjecture that this is due to the importance of the filtering, in terms of number of reviews per reviewer and number of votes, applied over the dataset.

5.3.1 Accuracy of the two KT models. The key observation is that switching to KT achieves very good to excellent RMSE scores, whatever the dataset considered. Notably, predicting the skill of writing helpful reviews is done much more accurately than predicting helpfulness. This allows to answer positively to the question expressed at the beginning of this paper: a model constructed on a large dataset can be used to assess procedural knowledge acquisition. Interestingly, predicting each sub-skill (corresponding to each feature) and combining these predictions to infer the global skill of writing helpful reviews is significantly better than predicting the skill at the coarse level of the model. In our test, this combination was naively done with the weights learned using the linear regression algorithm, normalized, bias included, to build the model of helpfulness. It is left as future works to determine more sophisticated weight combination.

Scores	minVotes = 12	minVotes = 23
$P(L_0)$	0	0
$P(T)$	0.014637	0.016329
mean($P(G)$)	0.346532	0.345426
variation($P(G)$)	0.007084	0.014245
mean($P((S))$	0.409464	0.409721
variation($P(S)$)	0.016246	0.029494
mean(L_n)	0.518769	0.518702
variation(L_n)	0.110546	0.106100
a-mKRMSE	0.673532	0.669856

Table 11: KT parameters, prediction and predictive accuracy for random sequences of helpfulness

5.3.2 Comparison with random sequences of helpfulness scores. The small RMSE indicates that the KT model is good at predicting the learning of the writing skill of the reviewers. However, in order to validate the hypothesis that these good results do not come from an intrinsic smoothing behavior of the KT model, we ran the model on random sequences of helpfulness score. To this end, we generated as many sequences as the original data set has and faked the helpfulness scores with generated random numbers between 0 and 1. The result, reported in Table 11 confirms that for both datasets the RMSE values are bad. It infers that for random sequence of numbers as the score of helpfulness, the model fails to predict the skill of the reviewers (that in this case is expectedly close to 0.5).

6 CONCLUSION

In this paper, we experimented with a large dataset of Amazon book reviews to show that a model of review helpfulness can be used to assess the acquisition of the skill of writing helpful reviews. Learning such an individual model of procedural knowledge acquisition has the advantages to be less prone to human variation and subjectivity (e.g., in judging the helpfulness of a review) and to not have to define precisely a hard to define skill, that is replaced by a model learned over the dataset. In our experiments, we modeled the quality of a review by a linear combination of metrics stemming from text analysis (like readability, polarity, spelling errors or length) and we use customer declared helpfulness as a ground truth for constructing the model. This model achieves comparable to slightly better accuracy results when compared to a state-of-the-art approach. We used Bayesian Knowledge Tracing (KT), a popular model of skill acquisition, to measure the evolution of the ability to write reviews of good quality over a period of time. Our tests validated our hypothesis, showing that the model of skill acquisition achieves a very good to near perfect accuracy score.

Our short term future works include the revision of both the helpfulness model and the skill acquisition model. In particular, the helpfulness model can be extended with advanced features like sentiment analysis or reviewer profiles features, while Deep Knowledge Tracing could be used instead of classical Knowledge Tracing. We also want to better understand the relation between the linear coefficient learned for the helpfulness model and the KT parameters of the corresponding sub-skills. Long term goals include the generalization of our approach to other datasets and skills. We are particularly interested in better understanding in what contexts skill acquisition with model building is more relevant than only building the model.

REFERENCES

[1] Arpita Agnihotri and Saurabh Bhattacharya. 2016. Online Review Helpfulness: Role of Qualitative Factors. *Psychology & Marketing* 33, 11 (Dec 2016), 1006–1017.

[2] Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. 2010. SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. In *LREC*.

[3] Kathleen M. Cauley. 1986. Studying Knowledge Acquisition: Distinctions among Procedural, Conceptual and Logical Knowledge. In *67th Annual Meeting of the American Educational Research Association.*

[4] Judith A Chevalier and Dina Mayzlin. 2006. The effect of word of mouth on sales: Online book reviews. *Journal of marketing research* 43, 3 (2006), 345–354.

[5] Meri Coleman and Ta Lin Liau. 1975. A computer readability formula designed for machine scoring. *Journal of Applied Psychology* 60, 2 (1975), 283.

[6] Albert T Corbett and John R Anderson. 1994. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User modeling and user-adapted interaction* 4, 4 (1994), 253–278.

[7] Yossi Ben David, Avi Segal, and Ya'akov (Kobi) Gal. 2016. Sequencing educational content in classrooms using Bayesian knowledge tracing. In *LAK*. 354–363.

[8] Anindya Ghose and Panagiotis Ipeirotis. 2009. The EconoMining project at NYU: Studying the economic value of user-generated content on the internet. *Journal of Revenue and Pricing Management* 8, 2-3 (2009), 241–246.

[9] Yue Gong, Joseph E. Beck, and Neil T. Heffernan. 2010. Comparing Knowledge Tracing and Performance Factor Analysis by Using Multiple Model Fitting Procedures. In *ITS*. 35–44.

[10] Robert Gunning. 1952. The technique of clear writing. (1952).

[11] William J. Hawkins, Neil T. Heffernan, and Ryan Shaun Joazeiro de Baker. 2014. Learning Bayesian Knowledge Tracing Parameters with a Knowledge Heuristic and Empirical Probabilities. In *ITS*. 150–155.

[12] Ruining He and Julian McAuley. 2016. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In *WWW*. 507–517.

[13] Hong Hong and Di Xu. 2015. Research of online review helpfulness based on negative binary regress model the mediator role of reader participation. In *2015 12th International Conference on Service Systems and Service Management (ICSSSM)*. 1–5.

[14] Jingxian Jiang, Ulrike Gretzel, and Rob Law. 2010. Do Negative Experiences Always Lead to Dissatisfaction? - Testing Attribution Theory in the Context of Online Travel Reviews. In *ENTER*. 297–308.

[15] J Peter Kincaid, Robert P Fishburne Jr, Richard L Rogers, and Brad S Chissom. 1975. *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.

[16] Nikolaos Korfiatis, Elena García-Bariocanal, and Salvador Sánchez-Alonso. 2012. Evaluating content quality and helpfulness of online product reviews: The interplay of review helpfulness vs. review content. *Electronic Commerce Research and Applications* 11, 3 (2012), 205–217.

[17] Yang Liu, Xiangji Huang, Aijun An, and Xiaohui Yu. 2008. Modeling and Predicting the Helpfulness of Online Reviews. In *ICDM*. 443–452.

[18] Julian John McAuley and Jure Leskovec. 2013. From amateurs to connoisseurs: modeling the evolution of user expertise through online reviews. In *WWW*. 897–908.

[19] Susan M. Mudambi and David Schuff. 2010. What Makes a Helpful Online Review? A Study of Customer Reviews on Amazon.com. *MIS Quarterly* 34, 1 (2010), 185–200.

[20] Philip I. Pavlik, Hao Cen, and Kenneth R. Koedinger. 2009. Performance Factors Analysis - A New Alternative to Knowledge Tracing. In *AIED*. 531–538.

[21] Radek Pelánek. 2015. Metrics for Evaluation of Student Models. In *EDM*. 19.

[22] Chris Piech, Jonathan Bassen, Jonathan Huang, Surya Ganguli, Mehran Sahami, Leonidas J. Guibas, and Jascha Sohl-Dickstein. 2015. Deep Knowledge Tracing. In *NIPS*. 505–513.

[23] RJ Senter and Edgar A Smith. 1967. *Automated readability index.* Technical Report. Univ. Cincinati.

[24] Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. 2003. Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network. In *HLT-NAACL*.

[25] Yutao Wang and Neil T. Heffernan. 2013. Extending Knowledge Tracing to Allow Partial Credit: Using Continuous versus Binary Nodes. In *AIED*. 181–188.

[26] Jianan Wu. 2017. Review popularity and review helpfulness: A model for user review effectiveness. *Decision Support Systems* 97 (2017), 92–103.

[27] Philip Fei Wu, Hans van der Heijden, and Nikolaos Korfiatis. 2011. The Influences of Negativity and Review Quality on the Helpfulness of Online Reviews. In *ICIS*.