A Discriminative Approach to Predicting Assessor Accuracy

Hyun Joon Jung and Matthew Lease

School of Information
University of Texas at Austin, USA
{hyunJoon,ml}@utexas.edu

Abstract. Modeling changes in individual relevance assessor performance over time offers new ways to improve the quality of relevance judgments, such as by dynamically routing judging tasks to assessors more likely to produce reliable judgments. Whereas prior assessor models have typically adopted a single generative approach, we formulate a discriminative, flexible feature-based model. This allows us to combine multiple generative models and integrate additional behavioral evidence, enabling better adaptation to temporal variance in assessor accuracy. Experiments using crowd assessor data from the NIST TREC 2011 Crowdsourcing Track show our model improves prediction accuracy by 26-36% across assessors, enabling 29-47% improved quality of relevance judgments to be collected at 17-45% lower cost.

Keywords: search evaluation, crowdsourcing, machine learning and modeling

1 Introduction

Recent efforts in efficiently collecting relevance judgments at scale have focused on how to collect high-quality relevance judgments with crowdsourcing [1] [2] [3]. Since quality of relevance judgments critically influences the results of IR system evaluation [4], a great deal of research has focused quality improvement of relevance judgments via various approaches: multiple labeling and aggregation [5], behavioral effects investigation [6], letting assessors select which tasks to work on [7], and efficient HIT (Human Intelligence Tasks) design [8].

Predicting the quality of judgments represents another opportunity to improve quality of crowdsourced relevance judgments. For instance, task routing in crowdsourcing [7] requires a method to match a worker to a task. One can route a specific judgment task to a specific assessor based on the prediction of a probability of an assessor's next judgment correctness, and expect improved quality of relevance judgments.

Prior work in predicting assessors' annotation performance has typically assumed that an assessor's judgments are independent and identically distributed (i.i.d) over time [9]. In other words, prior work has not considered temporal effects among judgments. To solve this problem, Donmez et al. [10] and Jung et al [11] proposed time-series models. However, while one could imagine many features characterizing an assessor's behavior, their models still rely upon a single generative model at time t.

To address this problem, we build a Generalizable feature-based Assessor Model (GAM) that allows us to flexibly capture a wider range of assessor behaviors by incorporating features which model different aspects of this behavior. We integrate various features from prior studies which were used mainly or only for the estimation of crowd assessor's annotation performance [11] or judgment simulation [4]. In addition, we devise several new behavioral features indicating an assessor's annotation performance over time and integrate them with the existing features selected from prior studies.

We investigate this predictive model with the public NIST TREC 2011 Crowdsourcing Track dataset[1]. Firstly, we evaluate prediction quality, both in terms of hard prediction (binary correct or not) and soft prediction (probability of making a correct label). In particular, we study the effect of a *decision reject option*, which improves prediction accuracy by sacrificing prediction coverage, providing a tuning parameter for aggressive vs. conservative prediction given model confidence. In the second experiment, we conduct an in-depth feature analysis in order to compare the relative importance of each feature. Finally, we evaluate the effectiveness of our predictive model for crowdsourced judgment quality improvement under a realistic scenario assuming task routing and label aggregation. Our empirical evaluation demonstrates that our model improves prediction accuracy by 26-36% across 54 assessors. In addition, our experiments show that the quality of relevance judgments by our prediction model-based task routing improves its accuracy by 29-47% with lower cost (17-45%). Our research questions are:

RQ1: Feature Design for Prediction Model *When we build a discriminative, feature-based learning framework for predicting work quality, what features are useful to include, and what is their relative importance?*

RQ2: Prediction Performance Improvement *Does our prediction model improve prediction performance? How does decision rejection trade-off coverage vs. accuracy of prediction model in comparison to other baselines?*

RQ3: Impact on Judgment Quality and Cost. *Can our prediction model improve the quality of relevance judgments and/or decrease cost of collecting judgments?*

2 Problem

Estimating and predicting crowd assessors' performance has gained relatively little attention in IR system evaluation. Most prior work in crowd assessor modeling has focused on simple estimation of assessors' performance via metrics such as accuracy and F1 [12] [13]. Unlike other studies, Caterette and Soboroff presented several assessor models based on Bayesian-style accuracy with various types of Beta priors [4]. Recently, Ipeirotis and Gabrilovich presented a similar type of Bayesian style accuracy with a different Beta prior in order to measure assessors' performance [8]. However, neither investigated prediction of an assessor's judgment quality.

Figure 1 shows two real examples of failures of existing assessor models in predicting assessor's judgment correctness. The more accurate left assessor (a) begins with very strong accuracy (0.8) which continually degrades over time, whereas accuracy of the right assessor (b) hovers steadily around 0.5. Suppose that a crowd worker's next

[1] https://sites.google.com/site/treccrowd/

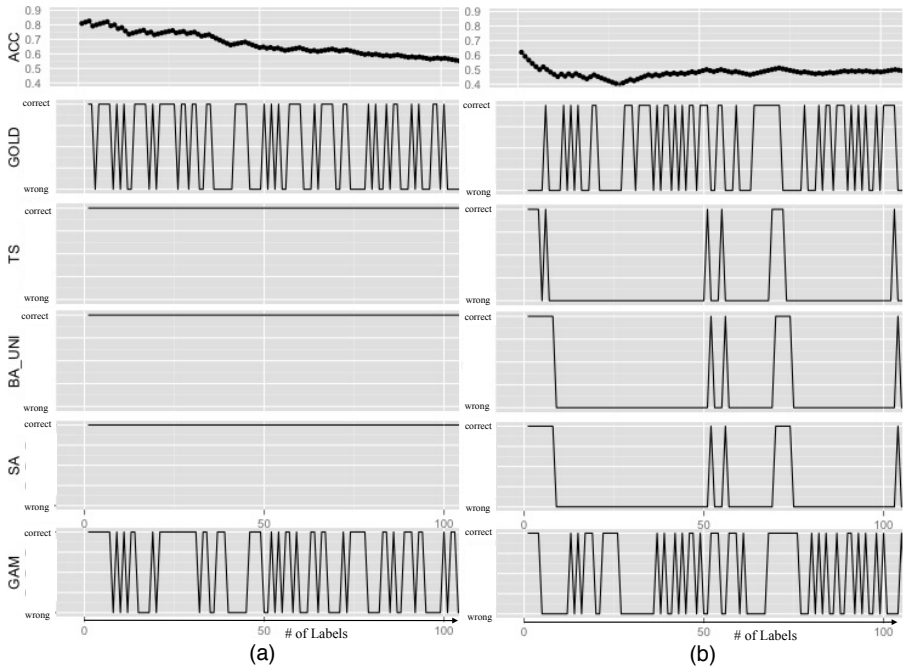

Fig. 1. Two examples of failures of existing assessor models and success of our proposed model, GAM in predicting the correctness of assessors' next label ((a) high accuracy assessor and (b) low accuracy assessor). While the agreement of a crowd assessor's judgments with that of the original NIST topic authority (GOLD) oscillates over time, the existing assessor models (Time-series (TS) [11]), Sample Running Accuracy (SA), Bayesian uniform beta prior (BA-UNI [8]) do not follow the temporal variation of the assessors' agreement with the gold labels. On the contrary, GAM is sensitive to such dynamics of labels over time for higher quality prediction.

label quality (y_t) is binary (correct/wrong) with respect to ground truth. While y_t oscillates over time, the existing models are not able to capture such temporal dynamics and thus prediction based on these models is almost always wrong. In particular, when an assessor's labeling accuracy is greater than 0.5 (eg., average accuracy = 0.67 in Figure 1 (a)), the prediction based on the existing models are always 1 (correct) even though the actual assessor's next label quality oscillates over time. A similar problem happens in Figure 1 (b) with another worker whose average accuracy is below 0.5.

In crowdsourcing and human computation, significant research has focused on the estimation or prediction of crowd workers' behavior or performance [14] [15]. However, most studies assumed that each annotation is independent and identically distributed (i.i.d) over time even though crowd worker behavior can have temporal dynamics as shown in Figure 1. Donmez et al. [10] was the first to propose a time-series model. Jung et al. [11] presented a temporal model to estimate asymptotic worker accuracy. However, while there exist many features characterizing a crowd assessor's behavior, these models only rely on the observation of labels [10] or labels' correctness [11]. For this reason, existing time-series models remain limited in terms of predicting an assessor's next judgment correctness as shown in Figure 1.

Problem Setting. Suppose that an assessor has completed n relevance judgments and each judgment has NIST expert labels available to judge an assessor's judgment correctness. In this work, we assume that NIST expert labels represent objective ground truth from which deviation is assumed to represent error, rather than valid, subjective disagreement. However, in practice, some level of disagreement is expected and common, even with simplified topical relevance [16]. We leave relaxing this assumption for future work.

The correctness of the ith judgment is denoted as $y_i \in \{0, 1\}$, where 1 and 0 represent correct or not. Thus, the performance of an assessor can be represented as a sequence of binary observations, $\mathbf{y} = \begin{bmatrix} y_1 \; y_2 \; \dots \; y_n \end{bmatrix}$. For example, if an assessor completed five relevance judgments and erred on the first and third respectively, then his *binary performance sequence* is encoded as $\mathbf{y} = \begin{bmatrix} 0 \; 1 \; 0 \; 1 \; 1 \end{bmatrix}$. **GOLD** in Figure 1 indicates $\mathbf{y}$ of each assessor.

For this problem, we propose a generalizable feature-based assessor model (GAM) that allows us to flexibly capture a wider range of assessors' behaviors by incorporating features which model different aspects of this behavior. Based on this model, we predict whether or not an assessor's next judgment will be correct, as defined by agreement with the NIST expert who developed and judged the topic originally. By this ability to flexibly model more aspects of assessor behavior, we expect greater predictive power and an opportunity for more accurate predictions.

We generate a multi-dimensional feature vector, $x_i = \begin{bmatrix} x_{1i} \; x_{2i} \; \dots \; x_{mi} \end{bmatrix}$ per time i and use x_i as an input of a prediction function f. Prior assessor models only consider a simple feature measure x_i by a single metric, accuracy, and then use this feature as an input of simple link function y_{i+1} = *roundOff*(x_i). Instead, our proposed model incorporates a multi-dimensional feature vector x_i and uses this feature vector with a learning framework $f(x_i, y_i) = y_{i+1}$. The bottom plot of Figure 1 shows how GAM is able to track the assessor's varying correctness with greater fidelity.

3 Method: Generalized Time-Varying Assessor Model (GAM)

In this section, we present a generalizable feature-based assessor model that incorporates various observable and latent features modeling different aspects of assessors' behavior. We first examine feature generation and integration, and then discuss learning a predictive model with the generated features.

3.1 Feature Generation and Integration

An assessor's behavior and annotation performance may be captured by various types of features. In this study, we generate and integrate two types of features shown in Table 1: observable and latent features. Bayesian-style features have various forms in prior work according to different Beta prior settings. Among them, we adopt $optimistic$ (a Beta prior $\alpha = 16, \beta = 1$) and $pessimistic$ (a Beta prior $\alpha = 1, \beta = 16$) assessor models from Carterette and Soboroff's study [4]. In addition, we adopt a Bayesian style accuracy from Ipeirotis and Gabrilovich's study which assumes a Beta prior ($\alpha = 0.5, \beta = 0.5$), referred to here as the $uniform$ assessor model. In these assessor models, each Beta prior characterizes each assessor's annotation performance.

	Feature Name	Description
Observable	Bayesian Optimistic Accuracy (BA_{opt}) [4]	a Bayesian style accuracy with a prior $Beta$ (16,1) $BA_{opt} = (x_t + 16)/(n_t + 17)$
	Bayesian Pessimistic Accuracy (BA_{pes}) [4]	a Bayesian style accuracy with a prior $Beta$ (1,16) $BA_{pes} = (x_t + 1)/(n_t + 17)$
	Bayesian Uniform Accuracy (BA_{uni}) [8]	a Bayesian style accuracy with a prior $Beta$ (0.5,0.5) $BA_{uni} = (x_t + 0.5/(n_t + 1)$
	Sample Running Accuracy (SA)	$SA_t = x_t/n_t$
	CurrentLabelQuality	a binary value indicating whether a current label is correct or wrong.
	TaskTime	time to spend in completing this judgment task. (ms)
	AccuracyChangeDirection (ACD)	a binary value indicating the absolute difference between $SA_{t-1} - SA_t$.
	TopicChange	a binary value indicating a topic change between time $t-1$ and time t.
	NumLabels	a cumulative number of completed relevance judgments at time t.
	TopicEverSeen	a real value [0$\sim$1] indicating the familiarity of a topic. $\frac{1}{a\ number\ of\ judgments\ on\ topic\ k\ at\ time\ t}$
Latent	Asymptotic Accuracy (AA) [11]	a time-series accuracy estimated by latent time-series model proposed by Jung et al. $\frac{c}{1-\phi}$.
	ϕ [11]	a temporal correlation indicating how frequently a sequence of correct/wrong observations has changed over time.
	c [11]	a variable indicating the direction of judgments between correct and wrong.

Table 1. Features of generalized assessor model (GAM). n is the number of total judgments and x is the number of relevance judgments at time t.

For instance, the $optimistic$ assessor model indicates that an assessor is likely to make a relevance judgment in a permissive fashion, while the $pessimistic$ model tends to make more non-relevant judgments than relevant judgments. The $uniform$ model has an equal chance of making a relevant or non-relevant judgment. Note that Bayesian style accuracies ($BA_{opt}, BA_{pes}, BA_{uni}$) were only used as a way of simulating judgments or estimating an assessor's performance in the original studies. In this study, we instead used these accuracies as a feature of estimating an assessor's annotation performance as well as predicting an assessor's next judgment's correctness. Other observable features include measurable features from a sequence of relevance judgments from an assessor. Among them, $TaskTime$ and $NumLabels$ are designed to capture an assessor's behavioral transition over time. $TopicChange$ checks the sensitivity of an assessor to topic variation over time. The $TopicEverSeen$ feature is designed to consider the effect of growing topic familiarity over time. The value is discounted by increased exposure to topic k.

Latent features are adopted from Jung et al's [11] model of temporal dynamics of assessor behavior (ϕ and c). While they only used $asymptotic\ accuracy$ (AA) as an indicator of an assessor's annotation performance, we integrate all three features (AA, ϕ, and c) into our generalized assessor model. Our intuition is that each feature may capture a different aspect of an assessor's annotation performance and thus the integration of various features enabling greater predictive power for more accurate predictions.

3.2 Predicting Judgments Quality

To select a learning model, we adopt **L1-regularized logistic regression** due to several reasons. Firstly, it supports probabilistic classification as well as binary prediction

by logistic function. In our problem setting, we conflate graded relevance judgments into binary values (0 or 1), and thus logistic regression is the best fit in order to handle such a binary classification problem. In addition, a logistic regression model allows us obtain the odds ratio, defined as the ratio of the probability of correct over incorrect relevance judgments. Secondly, L1-regularized logistic regression prevents over-fitting in learning models due to either co-linearity of the covariates or high-dimensionality. The regularized regression shrinks the estimates of the regression coefficients towards zero relative to the maximum likelihood estimate. Finally, logistic regression is relatively simple and fast. In practice, one of the challenging issues to run learning algorithms is that it takes too much time to update parameters and predict output values once a new label comes. However, this model is quite efficient.

In prediction, we consider a supervised learning task where we are given N training instances $\{(x_i, y_i), i = 1, ..., N\}$. Here, each $x_i \in \mathbb{R}^M$ is an M-dimensional feature vector, and $y_i \in 0, 1$ is a class label indicating whether an assessor's next judgment is correct (1) or wrong (0). Before fitting a model to our feature and target labels, we first normalize our features in order to ensure that normalized feature values implicitly weight all features equally in a model learning process. Logistic regression models the probability distribution of the class label y given a feature vector X as follows:

$$p(y = 1|x; \theta) = \sigma(\theta^T x) = \frac{1}{1 + exp(-\theta^T x)} \tag{1}$$

Here $\theta = \{\beta_0, \beta_1^T, ..., \beta_M^T\}$ are the parameters of the logistic regression model; $\sigma(\cdot)$ is the sigmoid function, defined by the second equality. The following function attempts to maximize the log-likelihood in order to fit a model to a given training data.

$$\max_{\theta}\{\sum_{i=1}^{N}[y_i(\beta_0 + \beta^T x_i) - log(1 + e^{\beta_0 + \beta^T x_i})] - \lambda \sum_{j=1}^{M} |\beta_j|\}. \tag{2}$$

3.3 Prediction with Decision Reject Option

Our predictive model can generate two types of outputs: a binary value predicting the correctness of an assessor's judgment (0 or 1) and a continuous value ($y_{i+1} \in [0, 1]$) indicating the probability of making a correct judgment. While a binary predictive value (*hard prediction*) can be used as it is, a probabilistic predicted value (*soft prediction*) can be used after a transformation, such as rounding-off. For instance, if an original predicted value is 0.76, we could round this to a binary predictive value of 1.

In term of soft prediction, there exists room for improving its quality by taking account of prediction confidence. For instance, if a value of soft prediction is close to 0.5, it fundamentally indicates very low confidence. Therefore, we may avoid the risk of getting noisy predictions by adopting a *decision rejection option* [17]. In this study, we round off a probabilistic predictive value with a decision reject option as follows. If $y_{i+1} < 0.5 - \delta$ or $y_{i+1} \geq 0.5 + \delta$ then y_{i+1} does not need any transformation and use its original value. If $y_{i+1} \geq 0.5 - \delta$ or $y_{i+1} < 0.5 + \delta$ then y_{i+1} is $null$, indicating the reject of decision. δ is a parameter to control the limits of decision reject option $\in [0, 0.5]$. High δ indicates a conservative prediction which increases the range of decision rejection while sacrificing coverage. On the other hand, low δ allows prediction

in a permissive manner, decreasing the threshold of decision rejection and increasing coverage.

4 Evaluation

Experimental Settings

Dataset. Data from the NIST TREC 2011 Crowdsourcing Track Task 2 is used. The dataset contains 89,624 *graded relevance judgments (2: strongly relevant, 1: relevant, 0: non-relevant)* collected from 762 workers rating the relevance of different Webpages to different search queries [18]. We conflate judgments into a binary scale (relevant / non-relevant), leaving prediction of graded judgment accuracy for future work. We processed this dataset to extract the original temporal order of the assessor's relevance judgments. We include 3,275 query-document pairs which have expert judgments labeled by NIST assessors, and we exclude workers making < 20 judgments to ensure stable estimation. Moreover, since the goal of our work is to predict assessors' next judgment quality, we intentionally focus on prolific workers who will continue to do this work in the future, for whom such predictions will be useful. 54 sequential relevance judgment sets are obtained, one per crowd worker. The average number of labels (i.e., sequence length) per worker is 154.

Metrics. Prior to measurement, we collect **gold** labels for each assessor by computing the agreement of a crowd assessor's judgments with that of the original NIST topic authority. We evaluate the performance of our prediction model with two metrics. Firstly, we measure the prediction performance with accuracy and *Mean Absolute Error* (MAE). Predicted probabilistic values (soft prediction) produced by our model are measured with MAE, indicating the absolute difference between a predicted value vs. original binary value indicating the correctness of an assessor's judgment: $MAE = \frac{1}{n}\sum_{i=1}^{n}|pred_i - gold_i|$, where n is the number of judgments. Rounded binary labels (hard labels) are evaluated by accuracy. Secondly, accuracy is used for measuring the prediction performance of the binary probabilistic values from our prediction method. Since our extracted dataset is well-balanced in terms of a ratio between relevant vs. non-relevant judgments, use of accuracy is appropriate.

Models. We evaluate our proposed Generalized Assessor Model (GAM) under various conditions of *decision reject options* with two metrics. Our initial model uses no decision reject option, setting $\delta = 0$. In order to examine the effect of *decision reject options*, we vary $\delta \in [0, 0.25]$ by 0.05 step-size. Since we have 54 workers, we build 54 different predictive models and evaluate their prediction performance and final judgment quality improvement.

Our model works in a sequential manner that updates the model parameter θ once a new binary observation value (correct/wrong) comes. We use each worker's first 20 binary observation values as an initial training set. For instance, suppose a worker has 50 sequential labels. We first collect a sequence of binary observation values (correct/wrong) by comparing a worker's label with a corresponding ground truth judged by NIST experts. Next, our prediction model takes the first 20 binary observation values and then predicts the 21st label's quality (correct/wrong) of this worker. Once actual 21st label comes from this worker, we measure the accuracy and MAE by comparing

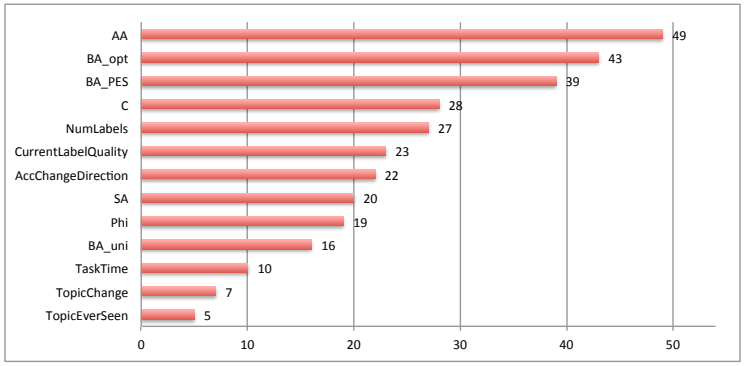

Fig. 2. Summary of relative feature importance across 54 regression models.

the label with a corresponding ground truth from NIST experts. For the following 29 judgments we repeat the same process in a sequential manner, predicting the quality of each label one-by-one.

To learn our logistic regression model, we choose the regularization parameter λ as 0.01 after the investigation of prediction performance with varying parameter values $\{0.1, 0.01, 0.001\}$ over the initial training set of each worker. For feature normalization, we apply standard min-max normalization to the 13 features defined in Section 3.1. Note that λ is the only model parameter we tune, and all settings of decision-reject parameter are reported in results.

As a baseline, we consider several assessor models proposed by prior studies [4] [8] [11] (Section 3.1). We adopt two assessor models from Carterette and Soboroff's study, $optimistic$ assessor (BA_{opt}) and $pessimistic$ assessor (BA_{pes}), and one assessor model of Bayesian accuracy (BA_{uni}) used in Ipeirotis and Gabrilovich's study (see Table 1). In addition, we test the performance of a time-series model (TS) proposed by Jung et al [11] and sample running accuracy (SA) as defined by Table 1. All of the baseline methods predict the binary correctness of the next judgment y_{i+1} by rounding off the worker's estimated accuracy at time i. *Decision reject options* are equally applied to all of the baseline methods.

4.1 Experiment 1 (RQ1): Feature Selection & Importance

Our first experiment is to figure out which features are relatively more important than others. Intuitively, having more features leads to more predictive power. However, in practice, excessive features may lead to over-fitting. Thus, we investigate relative feature importance by evaluating feature subsets.

We adopt the $bestglm$ r package[2] and run the BICg model in order to find the best subset regression models. Since we have 54 assessors, we run this method for all of the 54 original regression models. Next, we observe the selected features of each subset model, and count the cumulative selection of each feature across 54 regression models. Figure 2 shows the relative feature importance across 54 regression models for all of the assessors. Asymptotic accuracy (AA) is selected in 49 of 54 models, followed by BA_{opt}

[2] http://cran.r-project.org/web/packages/bestglm/vignettes/bestglm.pdf

Metric	GAM	TS	BA_{uni}	BA_{opt}	BA_{pes}	SA
Accuracy	0.802*	0.621	0.599	0.601	0.522	0.599
% Improvement	NA	29.1	33.9	33.4	53.6	33.9
# of Wins	NA	50	52	50	54	52
# of Ties	NA	3	1	3	0	1
# of Losses	NA	1	1	1	0	1
MAE	0.340*	0.444	0.459	0.448	0.488	0.458
% Improvement	NA	23.4	25.9	24.1	33.0	25.8
# of Wins	NA	53	53	53	54	53
# of Losses	NA	1	1	1	0	1

Table 2. Prediction performance (Accuracy and Mean Average Error) of different predictive models. *% Improvement* indicates an improvement in prediction performance between GAM vs. each baseline ($\frac{(GAM-baseline)}{baseline}$). *# of Wins* indicates the number of assessors that GAM outperforms a baseline method while *# of Losses* indicates the opposite of *# of Wins*. *# of Ties* indicates the number of assessors that both a method and GAM show the same prediction performance for an assessor. (*) indicates that GAM prediction outperforms the other six methods with a high statistical significance ($p<0.01$).

and BA_{pes} at 43 and 39, respectively. $Numlabels$ is selected in the half of the cases (27), which implicitly indicates that the increase in the quantity of the given tasks affects an assessor's next judgment correctness. On the contrary, the quality of next judgments of the 54 assessors in our dataset does not appear to be sensitive to topic change and topic familiarity. In addition, sample accuracy (SA) appears relatively less important than the other accuracy-based metrics such as AA, BA_{opt} and BA_{pes}. Interestingly, GAM model with only the top five features still shows little degraded performance (7-10% less) vs. the original regression models and outperforms all baselines.

4.2 Experiment 2 (RQ2): Prediction Performance Improvement

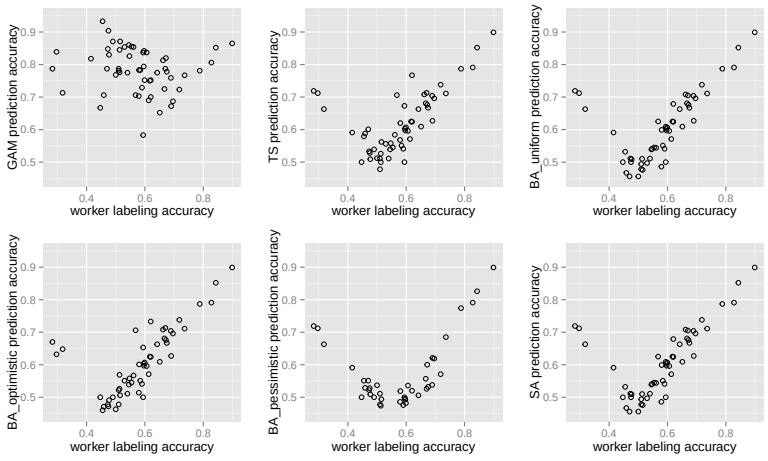

Fig. 3. Prediction accuracy of workers' next label by different methods ($\delta = 0$). While other methods show low accuracy against assessors with labeling accuracy near 0.5, the proposed model (GAM) shows significant improvement in predicting the correctness of workers' next judgments.

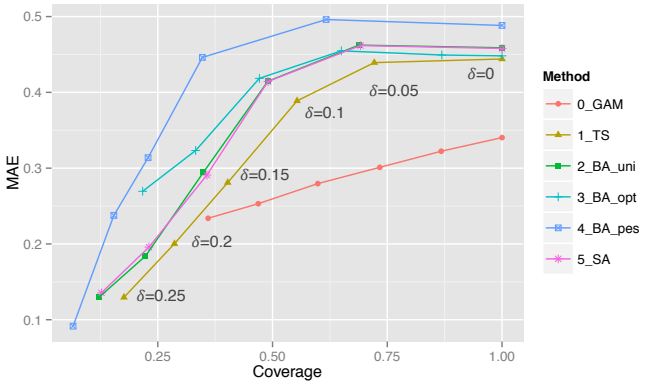

Fig. 4. Prediction performance (MAE) of assessors' next judgments and corresponding coverage across varying decision rejection options (δ=[0-0.25] by 0.05). While the other methods show a significant decrease in coverage, under all of the given reject options, GAM shows better coverage as well as prediction performance.

To answer our second research question, we first compare the overall prediction performance (Accuracy, MAE) of GAM with the baseline models across 54 crowd assessors. Table 2 shows that GAM prediction performance outperforms all of the baseline methods across 50-54 assessors in accuracy and 53-54 assessors in MAE. GAM improves the prediction accuracy (hard label) and MAE (soft label) by 26-36% on average. GAM prediction errs for only one assessor vs. the baselines. However, even for this assessor, GAM only made one or two more prediction errors in comparison to the other baselines.

Figure 3 shows the relationship between assessors' labeling accuracy (sample running accuracy) vs. prediction accuracy of GAM and the baseline models. While the baseline models show low accuracy against assessors whose labeling accuracy is near 0.5, GAM significantly improves prediction error for those assessors in particular.

Lastly, we examine the effects of *decision reject options* on GAM prediction. Figure 4 demonstrates that the baseline models show sharp decline of coverage in prediction in order to significantly improve their prediction accuracies. However, the coverage of GAM prediction only gently decreases; even with the second strongest reject option ($\delta = 0.2$), it still covers almost the half of prediction. In sum, GAM prediction not only outperforms the baseline models in terms of prediction accuracy, but it also shows less sensitivity to the increase of the decision reject option.

4.3 Experiment 3 (RQ3): Impact on judgment quality and cost

Our last experiment is to examine quality effects on relevance judgments via the proposed prediction model. We conduct an experiment based on task routing. For instance, if the prediction of an assessor's next judgment indicates that the assessor is expected to be correct, we route the given topic-document pair to this assessor and measure actual judgment quality against ground truth labeled by NIST. From our dataset, we only use 826 topic-document pairs that have more than three judgments per topic-document pair. Since the average number of judges per query is about 3.7, we test the cost saving effect with varying three task routing scenarios (*Number of Judges* = $\{1, 2, 3\}$). Judgment

	Prediction Models for Task routing							No Routing
Number of Judges	GAM	TS	BA_{uni}	BA_{opt}	BA_{pes}	SA	**Random**	**All labels**
1	0.786*	0.604	0.578	0.582	0.558	0.569	0.556	0.595
% Improvement	NA	30.1	36.0	35.1	40.9	38.1	41.4	
2	0.816**	0.617	0.592	0.595	0.574	0.582	0.572	
% Improvement	NA	32.3	37.8	37.1	42.2	40.2	42.7	
3	0.880*	0.647	0.608	0.623	0.598	0.608	0.581	
% Improvement	NA	36.0	44.7	41.3	47.2	44.7	51.5	

Table 3. Accuracy of relevance judgments via predictive models. *Number of Judges* indicates the number of judges per query-document pair. When the *Number of Judges* > 1, majority voting is used for label aggregation. Accuracy is measured against NIST expert gold labels. *% Improvement* indicates an improvement in label accuracy between GAM vs. each baseline ($\frac{(GAM-baseline)}{baseline}$). The average number of judges per query-document pair is 3.7. (*) indicates that GAM prediction outperforms the other six methods with high statistical significance ($p<0.01$).

quality is measured with accuracy, and a paired t-test is conducted to check whether quality improvement is statistically significant.

Table 3 shows the results of judgment quality via predictive model-based task routing. GAM substantially outperforms the other baselines across three task routing cases. The improvement of final judgment quality grows with the increase of the number of judges per query-document pair (*Number of Judges*) from 29-32% to 36-47%. Notice that GAM with only two routed judges achieves 29% quality improvement. Moreover, GAM provides high-quality relevance judgments (accuracy > 0.8) with only 54% = ($\frac{2}{3.7}$) of the original assessment cost. In contrast, we see that task routing with baselines alone (BA_{uni},BA_{pes},SA) may not be any better than random assignment.

5 Conclusion and Future Work

Despite recent efforts of quality improvement in crowdsourced relevance judgment, prior work in crowd assessor modeling cannot adequately predict an assessor's next judgment quality since it simply measures assessor performance via a single generative model without considering temporal effects among relevance judgments. We present a general discriminative learning framework for integrating arbitrary and diverse evidence for temporal modeling and prediction of crowd work accuracy. Our experiments demonstrate that the proposed model improves prediction performance by 26-36% as well as crowdsourced relevance judgment quality by 29-47% at 17-45% lower cost.

As a next step, we plan to relax our restrictive assumption of the existence of NIST expert labels to judge the correctness of an assessor's judgments. In addition, we want to examine how to evaluate the correctness of judgments in recognition that even topical judgments are still subjective. Beyond that, we plan to further investigate how to use this model for different applications of quality assurance in crowdsourcing, such as weighted label aggregation and spam worker filtering.

Acknowledgments. We thank the anonymous reviewers for their feedback. This work is supported in part by DARPA YFA Award N66001-12-1-4256, IMLS Early Career grant RE-04-13-0042-13, and NSF CAREER grant 1253413. Any opinions, findings, and conclusions or recommendations expressed by the authors do not express the views of the supporting funding agencies.

References

1. Alonso, O., Rose, D.E., Stewart, B.: Crowdsourcing for relevance evaluation. ACM SIGIR Forum **42** (2008) 9–15
2. Vuurens, J.B., de Vries, A.P.: Obtaining High-Quality Relevance Judgments Using Crowdsourcing. IEEE Internet Computing **16** (2012) 20–27
3. Lease, M., Kazai, G.: Overview of the TREC 2011 Crowdsourcing Track (Conference Notebook). In: 20th Text Retrieval Conference (TREC). (2011)
4. Carterette, B., Soboroff, I.: The effect of assessor error on IR system evaluation. In: Proceedings of the 33rd international ACM SIGIR conference on Research and Development in Information Retrieval. SIGIR '10 (2010) 539–546
5. Hosseini, M., Cox, I.J., Milić-frayling, N.: On aggregating labels from multiple crowd. In: Proceedings of the 34th European Conference on Advances in Information Retrieval. ECIR '12 (2012) 182–194
6. Kazai, G., Kamps, J., Milic-Frayling, N.: The Face of Quality in Crowdsourcing Relevance Labels: Demographics, Personality and Labeling Accuracy. In: Proceedings of the 21st ACM International Conference on Information and Knowledge Management. CIKM '12 (2012) 2583–2586
7. Law, E., Bennett, P., Horvitz, E.: The effects of choice in routing relevance judgments. In: Proceedings of the 34th ACM SIGIR conference on Research and development in Information. SIGIR '11 (2011) 1127–1128
8. Ipeirotis, P.G., Gabrilovich, E.: Quizz: targeted crowdsourcing with a billion (potential) users. In: Proceedings of the 23rd international conference on World Wide Web. WWW '14 (2014) 143–154
9. Yuen, M., King, I., Leung, K.S.: Task recommendation in crowdsourcing systems. In: Proceedings of the First International Workshop on Crowdsourcing and Data Mining. (2012) 22–26
10. Donmez, P., Carbonell, J., Schneider, J.: A probabilistic framework to learn from multiple annotators with time-varying accuracy. In: Proceedings of the SIAM International Conference on Data Mining. (2010) 826–837
11. Jung, H.J., Park, Y., Lease, M.: Predicting Next Label Quality: A Time-Series Model of Crowdwork. In: Proceedings of the 2nd AAAI Conference on Human Computation. HCOMP '14 (2014) 87–95
12. Kazai, G.: In search of quality in crowdsourcing for search engine evaluation. In: Proceedings of the 30th European Conference on Advances in Information Retrieval. ECIR '11 (2011) 165–176
13. Smucker, M.D., Jethani, C.P.: Measuring assessor accuracy: a comparison of NIST assessors and user study participants. In: Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval. SIGIR '11 (2011) 1231–1232
14. Raykar, V., Yu, S.: Eliminating spammers and ranking annotators for crowdsourced labeling tasks. Journal of Machine Learning Research **13** (2012) 491–518
15. Rzeszotarski, J.M., Kittur, A.: Instrumenting the crowd: Using implicit behavioral measures to predict task performance. In: Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology. UIST '11 (2011) 13–22
16. Voorhees, E.M.: Variations in relevance judgments and the measurement of retrieval effectiveness. Information Processing and Management **36** (2000) 697–716
17. Pillai, I., Fumera, G., Roli, F.: Multi-label classification with a reject option. Pattern Recognition **46** (2013) 2256 – 2266
18. Buckley, C., Lease, M., Smucker, M.D.: Overview of the TREC 2010 Relevance Feedback Track (Notebook). In: 19th Text Retrieval Conference (TREC). (2010)