Integration of Multiple Annotators by Aggregating Experts and Filtering Novices

Ping Zhang and Zoran Obradovic
Center for Data Analytics and Biomedical Informatics, Temple University, Philadelphia, PA 19122, USA
{ping, zoran.obradovic}@temple.edu

Abstract—**Learning from noisy labels obtained from multiple annotators and without access to any true labels is an increasingly important problem in bioinformatics and biomedicine. In our method, this challenge is addressed by iteratively filtering low-quality annotators and estimating the consensus labels based only on the remaining experts that provide higher-quality annotations. Experiments on biomedical text classification and CASP9 protein disorder prediction tasks provide evidence that the proposed algorithm is more accurate than the majority voting and previously developed multi-annotator approaches. The benefit of using the new method is particularly large when low-quality annotators dominate. Moreover, the new algorithm also suggests the most relevant annotators for each instance, thus paving the way for understanding the behaviors of each annotator and building more reliable predictive models for bioinformatics applications.**

Keywords-crowdsourcing; multiple noisy annotators; data curation; protein disorder prediction

I. INTRODUCTION

Data annotation (manual data curation) tasks are at the very heart of modern biology. This work was motivated by the following problem: There is considerable interest in the development of automated programs for assigning topics to each article that is registered in the PubMed database. Machine learning is a natural approach for the development of such programs, but machine learning tools require the input of large labeled samples. Thus, we would need large amounts of texts classified according to the MESH terminology [1]. However, it is very difficult and expensive to obtain such classifications from top human experts. Alternatively, one could have biomedical students (i.e., novices with different levels of expertise) label the texts, which are available both in larger numbers and at a much lower cost. The labeling provided by students may be erroneous though, especially for texts which are difficult to classify. Similar scenarios of utilizing such novice but readily available annotators in the process of learning arise in many other practical domains. For instance, weak annotations from novices might be generated using crowdsourcing services (e.g., Amazon's Mechanical Turk[1]).

Also annotators need not be human beings: they can also be, for instance, automatic classifiers. Then, another motivation problem is how to select and/or integrate multiple unreliable biomedical prediction models. For example, experimental characterizations (e.g., X-ray crystallography, NMR spectroscopy) of intrinsically disordered proteins (IDPs) are very expensive and time consuming, but they provide expert annotations. While a number of predictors for identification have been developed over the last decade, none can be taken as a fully reliable one on its own (i.e., the predictors are novice annotators) because all of them were developed on small, biased, annotated data.

In this paper, we address the question of whether it is possible to learn a classification model when provided with multiple noisy labels instead of a single golden standard. We are especially interested in the situation when the annotations are dominated by novices. Here, we propose a novel algorithm to integrate multiple annotators by **A**ggregating **E**xperts and **F**iltering **N**ovices, which we call AEFN. Our method iteratively evaluates annotators, filters the low-quality annotators, and re-estimates the labels based only on information obtained from the good annotators. The noisy annotations we integrate are from any combination of human and previously existing machine-based classifiers, and thus AEFN can be applied to many bioinformatics problems.

The rest of the paper is organized as follows. Section II presents the related work. Section III describes our AEFN algorithm. Section IV presents the application in a biomedical text classification task. Section V presents the application in the CASP9 protein disorder prediction task. Finally, section VI concludes the paper.

II. RELATED WORK

The problem of how to make the best use of the labeling information provided by multiple annotators to approximate the hidden true concept has been receiving increasing attention. One research direction focuses on annotator filtering (i.e., identifying low-performing annotators and excluding them). One of the commonly used strategies is to inject some items with known true labels into the annotations and use them to evaluate the annotators and thus filter the novices[2]. Also, without using any true labels, low-quality annotators can be pruned out via Z-score [2], via compactness analysis of the annotator clusters [3], and via a model trained from the entire dataset with all annotators [4].

Another research direction focuses on aggregating labels from multiple annotators for a given example in order to arrive at a single consensus label. A commonly used strategy is simple Majority Voting (MV) [5, 6]. In the MV method, the annotation that receives the maximum number of votes is treated as the final aggregated label, with ties broken randomly. A limitation of MV is that the consensus label for an

[1] Amazon Mechanical Turk found at http://www.mturk.com/.

[2] This is the strategy used by CrowdFlower found at http://crowdflower.com/docs/gold/.

example is estimated locally, considering only the labels assigned to that example (without regard to accuracy of the annotators involved in other examples). One could address this problem by introducing weighted voting. Weighted voting assigns greater weights to more accurate annotators, where accuracy is estimated by agreement with known expert labels. For example, Snow et al. [7] adopted a supervised Naive Bayes method to estimate the consensus labels from the reserved gold standard examples. However, supervised integration might be hard to control: if we reserve many gold standard examples, the cost will be high in expert annotation (why we are doing multiple-annotation in the first place); if we only reserve few gold standard examples, the weights calculated are unreliable. To better model data that have been processed by multiple annotators and without access to any true labels, a group of multi-annotator learning algorithms [8-11] that correct the annotator biases by estimating the annotator accuracy and the actual true label jointly has been proposed. They showed significantly improved performance in performed experiments when compared to MV and weighted voting.

In bioinformatics, there has also already been some literature for dealing with multi-annotation settings. In the application of manual data curation, [12-14] proposed couples of mathematical models to estimate annotator-specific correctness from different forms of computation of inter-annotator agreement, and facilitated the effective use of all annotated data. In the application of protein disorder prediction, meta approaches (e.g., metaPrDos [15], MFDp [16], and GSMetaDisorder [17]) which integrate the prediction results of several methods are widely used. Most of the meta predictors are supervised learning based (i.e., they are developed relying on disorder/order labeled training datasets that contain a very small number of proteins that have not already been used for development of the component predictors), and are prone to over-optimization in their integration processes. To address the problem, a meta predictor which is constructed in a completely unsupervised process (i.e., without using any confirmed disorder/order annotations) was developed by our group [18].

This paper differs from the related studies in the following aspects: (1) Unlike all previous approaches, we combined annotator filtering to the process of consensus labeling. The proposed AEFN algorithm filters novice annotators without using any true labels and estimates the ground truth based only on the good annotators. (2) Unlike all previous approaches, the proposed AEFN algorithm not only detects novice annotators but also provides more accurate estimates of each good annotator's performance at each Gaussian component. (3) Unlike most previous approaches, with the exception of [10] and [11], the proposed AEFN algorithm is data-dependent by addressing situations when annotators may not consistently accurate across the entire data.

III. METHOD

In III.A we introduce a score to rank annotators and in III.B we review a data-dependent algorithm (GMM-MAPML) to model data that has been processed by multiple annotators. Based on these works, in III.C we propose a new algorithm named AEFN, which filters the low-quality annotators without using any true labels and estimates the ground truth based only on the good annotators.

A. Annotator Model and Ranking Evaluation Score

In the multiple-annotation setting, an annotator provides a noisy version of the true label. Let $y_i^j \in \{0,1\}$ be the label assigned to the i-th instance by the j-th annotator, and let y_i be the hidden true label. Following the previous studies [8, 9] we model the accuracy of the annotator separately on the positive and the negative instances. If the true label is one, the sensitivity (true positive rate) α^j for the j-th annotator is the probability that the annotator labels it as one: $\alpha^j = \Pr[y_i^j = 1 | y_i = 1]$. On the other hand, if the true label is zero, the specificity (1-false positive rate) β^j is the probability that annotator labels as zero: $\beta^j = \Pr[y_i^j = 0 | y_i = 0]$.

Often we have low-quality annotators who make much more mistakes than high-quality annotators. In the extreme cases, we have spammers who assign labels randomly (e.g., without actually looking at the instances, doesn't understand the labeling criteria). More precisely an annotator is a spammer if the probability of observed label y_i^j being one given the true label y_i is independent of the true label, i.e.,

$$\Pr[y_i^j = 1 | y_i] = \Pr[y_i^j = 1]. \quad (1)$$

This means that the annotator is assigning labels randomly without actually looking at the data or understanding the data correctly. Equivalently (1) can be written as

$$\Pr[y_i^j = 1 | y_i = 1] = \Pr[y_i^j = 1 | y_i = 0]. \quad (2)$$

In other words, $\alpha^j = 1 - \beta^j$. Hence, in the annotator model, a spammer is an annotator meets the equation $\alpha^j + \beta^j - 1 = 0$. This corresponds to the diagonal on the ROC curve. In this study, we don't consider adversarial annotators (i.e., who has discriminatory power but flips the labels on purpose). Then a spammer is the annotator with the lowest quality. Hence, to rank annotators we define the ranking evaluation score as

$$S^j = |\alpha^j + \beta^j - 1|.$$

Then, an annotator is a spammer if S^j is close to zero, while a perfect annotator has S^j=1. In this study, we favor good annotators who have S^j>0.5. Also we call annotators who have S^j<0.5 as novices, and filter them during the integration process. The evaluation score S^j was first used by [9] in its experiments, and was formalized and proved recently by [19].

B. Data-dependent Model and Review of the GMM-MAPML Algorithm

In the annotator model introduced in III.A, it is implicitly assumed that α and β (i.e., the performance of the annotators) don't depend on the instances. To relax this impractical assumption, a GMM-MAPML algorithm [11] is developed to estimate the true labels for learning from multiple annotators of unknown quality. The algorithm takes into account that the annotators are not only unreliable, but may also be inconsistently accurate depending on the data. Given a dataset $D=\{x_i, y_i^1, \ldots, y_i^R\}$ (where x_i is an instance, $y_i^j \in \{0,1\}$ is the corresponding binary label assigned to the instance x_i by the j-th annotator and R is the number of the annotators), GMM-

MAPML algorithm uses EM algorithm and Bayesian information criterion (BIC) to get parameters of the fittest Gaussian mixture model (GMM) and its mixture components' responsibilities (τ_{ik}) for each instance. Based on the intuition that real world annotators have different sensitivity and specificity for different groups of instances, the sensitivity α_k^j and specificity β_k^j are defined as:

$\alpha_k^j = \Pr(y_i^j = 1 \mid y_i = 1$, k-th Gaussian component generates $x_i)$

$\beta_k^j = \Pr(y_i^j = 0 \mid y_i = 0$, k-th Gaussian component generates $x_i)$

where j=1,…,R; k=1,…,K. Therefore, the algorithm models the annotators to generate labels as follows: given an instance x_i to label, the annotators find the Gaussian mixture component which most probably generates that instance. Then the annotators generate labels with their sensitivities and specificities at the most probable component.

By following the modified annotator model, GMM-MAPML uses majority voting to initialize the probabilistic labels z_i (i.e., the probability when the hidden true label is 1). Then, the algorithm alternately carries out the maximum-likelihood (ML) estimation and the maximum a posterior (MAP) estimation: given the current estimates of probabilistic labels z_i, the ML estimation measures annotators' performance (i.e., their sensitivity α and specificity β) at each mixture component and learns a classifier with parameter w; given the estimated sensitivity α, specificity β, and the prior probability which is provided by the learned classifier, the MAP estimation gets the updated probabilistic labels z_i based on the Bayesian rule. After the two estimations converge, the GMM-MAPML algorithm outputs both the probabilistic labels z_i and the model parameters $\phi = \{w, \alpha, \beta\}$. The GMM-MAPML estimations of the hidden true labels depend both on observations and on the labels from all annotators. A brief summary of the GMM-MAPML algorithm is shown at algorithm 1. For details please refer to [11].

C. The AEFN Algorithm

Both data-independent and data-dependent algorithms aggregate all annotators by estimating the annotator accuracy and the actual true label jointly. Experiments conducted in previous studies [8-11] show the performance improvement of these algorithms as compared to the majority voting baseline. Recent experiments show that it is not the case that employing more annotators regardless of their expertise will result in improved highest aggregating performance [18]. In some cases, a consensus labeling of couples of experts will achieve a better performance. However, the authors didn't discuss the method to select high-quality annotators and/or remove low-quality annotators; instead they tried all possible combinations of annotators in a brute-force way.

Inspired by the experiments and the ranking evaluation score defined in the III.A, we propose an AEFN algorithm by extending the GMM-MAPML described in the III.B. In each iteration, ML estimation measures annotators' performance at each mixture component (i.e., their sensitivity α_k^j and specificity β_k^j). Then, we add a step to filter the low-quality annotators at each Gaussian component according to the score S_k^j (i.e., the ranking evaluation score of j-th annotator at k-th Gaussian component): if S_k^j is smaller than a pruning threshold we filter the j-th annotator from the pool of annotators at the k-th Gaussian component. Thus, we refit the MAP estimation with only the good annotators and get the updated probabilistic labels z_i based on the Bayesian rule. Note that the annotator filtering step is added after the first iteration, because z_i is initialized by majority voting and doesn't depend on observation in the first iteration. The AEFN algorithm is summarized at algorithm 2.

The AEFN Algorithm has following properties: (1) It is unsupervised. The inputs of algorithm 2 are instances and their noisy labels obtained from multiple annotators. The integration process doesn't involve any true labels. (2) It is data-dependent. The annotator evaluation and filtering processes are based on each Gaussian component of the entire data. (3) It is explainable. The output of algorithm 2 contains the sets of low-quality and high-quality annotators at each Gaussian component, thus paving the way for understanding the behaviors of each annotator.

Algorithm 1: GMM-MAPML Algorithm

1. Find the fittest K-mixture-component GMM for the instances, and get the corresponding GMM parameters and components responsibilities τ_{ik} for each instance.
2. Initialize $z_i = (1/R)\sum_{j=1}^{R} y_i^j$ based on majority voting.
3. (ML estimation) Given z_i, estimate the sensitivity and specificity of j-th annotator at k-th component as follows.

$$\alpha_k^j = \sum_{i=1}^{N} z_{ik} y_i^j \Big/ \sum_{i=1}^{N} z_{ik}$$
$$\beta_k^j = \sum_{i=1}^{N} (\tau_{ik} - z_{ik})(1 - y_i^j) \Big/ \sum_{i=1}^{N} (\tau_{ik} - z_{ik}) \quad (3)$$

Also learn a logistic regression classifier by the Newton-Raphson update for optimizing w. Then we can calculate the prior probability p_i for the positive class as

$$p_i = \Pr[y_i = 1 \mid \boldsymbol{x}_i, \boldsymbol{w}] = \sigma(\boldsymbol{w}^T \boldsymbol{x}_i) . \quad (4)$$

4. (MAP estimation) Given the sensitivity and specificity of each annotator at each component and the classifier parameter, update z_i as follows.

$$z_i = \frac{a_i p_i}{a_i p_i + b_i (1 - p_i)} \quad (5)$$

where

$$p_i = \Pr[y_i = 1 \mid \boldsymbol{x}_i, \boldsymbol{w}] = \sigma(\boldsymbol{w}^T \boldsymbol{x}_i)$$

$$a_i = \prod_{j=1}^{R} [\alpha_q^j]^{y_i^j} [1 - \alpha_q^j]^{1 - y_i^j}$$

$$b_i = \prod_{j=1}^{R} [1 - \beta_q^j]^{y_i^j} [\beta_q^j]^{1 - y_i^j}$$

$$q = \arg\max_{k=1,\ldots,K} (\tau_{ik})$$

Iterate 3 and 4 till convergence.

Algorithm 2: AEFN Algorithm

Input: Dataset $D = \{x_i, y_i^1, ..., y_i^R\}_{i=1}^N$ containing N instances. Each instances has binary labels $y_i^j \in \{0,1\}$ from R annotators.

1: Find the fittest K-mixture-component GMM for the instances, and get the corresponding GMM parameters and components responsibilities τ_{ik} for each instance.

2: Initialize $\Lambda_k = \{1,\ldots,R\}$ the sets of good annotators for each Gaussian component k=1,…,K.

3: Initialize $z_i = (1/R)\sum_{j=1}^{R} y_i^j$ based on a majority voting.

4: Initialize iteration indication iter $\leftarrow$ 0.

5: **repeat**

6: (ML estimation)

7: Update the sensitivity α_k^j and specificity β_k^j based on (3), $\forall j \in \Lambda_k$.

8: Update p_i based on (4).

9: (Low-quality annotators filtering)

10: **if** iter>0 (check from the second iteration)

11: **for all** k=1,…,K (all Gaussian components) **do**

12: **for all** $j \in \Lambda_k$ **do**

13: Update $S_k^j = |\alpha_k^j + \beta_k^j - 1|$.

14: **if** $S_k^j < \xi$ (the pruning threshold) **then**

15: $\Lambda_k \leftarrow \Lambda_k - \{j\}$

16: **end if**

17: **end for**

18: **end for**

19: **end if**

20: (MAP estimation)

21: Estimate z_i by using (5), $\forall$ i=1,…,N restricted to the annotators in the set Λ_k instead of integrating all R annotators.

22: iter $\leftarrow$ iter+1(update the number of iteration)

23: **until** change of z_i (5) between two successive iterations< ε.

24: Estimate the hidden true label y_i by applying a threshold γ on z_i. That is, y_i=1 if $z_i > \gamma$ and y_i=0 otherwise.

Output:

- Detected low-quality annotators of all Gaussian components in set $\{1,\ldots,R\} - \Lambda_k$.
- Good quality annotators of all Gaussian components in Λ_k with sensitivity α_k^j and specificity β_k^j, for $j \in \Lambda_k$, k=1,…,K.
- The probabilistic labels z_i and the estimation of the hidden true label y_i, $\forall$ i=1,…,N.

In the experiment, we set the convergence tolerance $\varepsilon = 10^{-3}$, the pruning threshold ξ =0.5, and the prediction threshold γ =0.5.

IV. Biomedical Text Classification Experiment

In this section we experimentally validate the proposed AEFN algorithm on a biomedical text classification task.

A. Data Description and Experiment Setup

Much current research in biomedical text mining is concerned with serving biologists by extracting certain information from scientific text. To identify information-bearing text fragments (high-utility fragments) within scientific text, Rzhetsky et al. [12] prepared a publicly available corpus of 10,000 sentences from scientific texts (PubMed and GeneWays corpus). At their first annotation cycle, each sentence in that corpus was annotated by three out of eight experts along five dimensions (i.e., *Focus*, *Polarity*, *Certainty*, *Evidence*, and *Direction/Trend*). At their second annotation cycle, they randomly sampled a subset of 1,000 sentences (out of the original 10,000) to re-annotate by five new experts. In our experiment, we used that 1,000-sentence subset from the second annotation cycle of their corpus[3]. In this dataset, 1,444 text fragments (out of 1,000 sentences) were labeled by eight experts (three-experts-per-fragment in the first cycle, and five-experts-per-fragment in the second cycle). We used these 1,444 text fragments as input instances. Because the five experts from the second annotation cycle annotated all 1,444 text fragments, we used them as the independent annotators. In conducted experiments we assumed that majority vote of eight experts (three providing labels at the first cycle and five at the second cycle) provided at [12] is a fair approximation to the true label of the fragment as in general opinions of more labelers is beneficial if they are experts. After text preprocessing (e.g., rare terms removal, stemming, stop word removal), we converted each text fragment by recording the tf-idf (term frequency-inverse document frequency) weight [20] of the most common words in the corpus, resulting in a 417-element feature vector for each instance.

In our experiment, we used the *Evidence* and *Focus* labels to define two separate binary classification tasks. For *Evidence* labels, {E3} was assigned to class 1 as direct evidence present in that instance, while {E0, E1, E2} were assigned to class 0 as no stated or no explicit evidence. Similarly, *Focus* labels {S, MS, SG} were assigned to class 1 as corresponding to instances describing findings and discovery, and {G, M, MG} were assigned to class 0 as corresponding to instances describing general knowledge or a method.

B. Experimental Results

In our comparisons, we considered three multiple-annotator methods: (1) **Majority Voting** that uses the average of five annotators' votes as the estimation of the hidden true label; (2) **MAP-ML** that estimates the hidden true labels and annotators' constant accuracy across all the input data using a data-independent model [8, 9]; (3) **GMM-MAPML** that uses a data-dependent model [11] to jointly estimate the hidden true labels and annotators' varying accuracy at each Gaussian component as described in section III.B (algorithm 1); and (4) the **AEFN** algorithm proposed in this paper that filters low-quality annotators and estimates the hidden true labels based only on the good annotators as described in section III.C (algorithm 2). For further comparisons, we also

[3] Available at http://www.ploscompbiol.org/article/fetchSingleRepresentation.action?uri=info:doi/10.1371/journal.pcbi.1000391.s002.

learned an additional logistic regression classifier: (5) **LR Concatenation** that considers every pair (instance, label) provided by each annotator as a separate example and builds a classifier in the traditional single annotator manner by concatenating all annotators' labels as a training set.

The ROC results for four multiple-annotator methods and the LR classifier on the biomedical *evidence* classification based on text annotations from 5 experts are shown at Fig. 1. On this comparison our AEFN algorithm had an AUC score of 0.837. The score was superior to the Majority Voting, MAP-ML and GMM-MAPML methods. In addition, Fig. 1 also shows that building a classifier in the traditional single annotator manner (simply concatenates all annotators' labels as LR Concatenation) without regard for the annotator properties may not be effective for the multi-annotator problems. Using the same measures and procedures, the ROC results of five methods on *focus* classification task are shown at Fig. 2. Similar to Fig. 1, the AUC score of our AEFN was better than those obtained by all the competitor methods in the analysis.

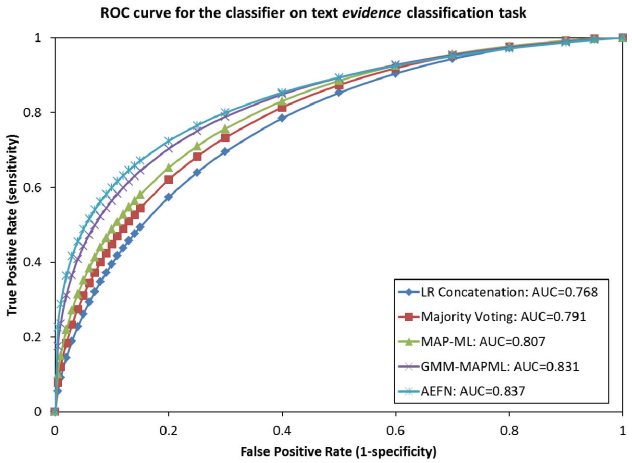

Figure 1. The ROC comparison of our AEFN vs. 3 alternative multiple-annotator methods and the LR Concatenation logistic regression classifier on the biomedical *evidence* classification using text annotations from 5 experts. Methods are sorted in legend of the figure according to their AUC value.

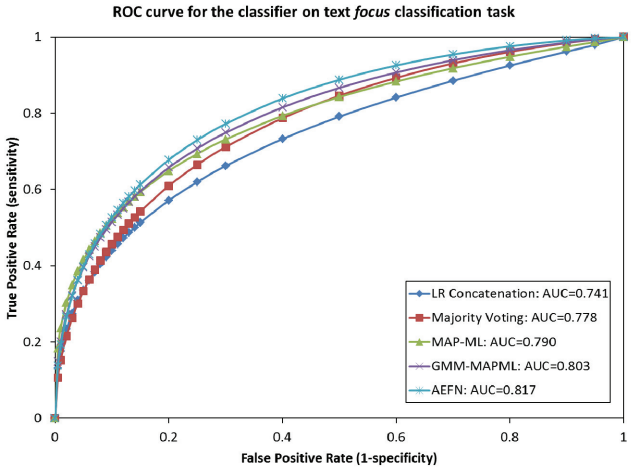

Figure 2. The ROC comparison of our AEFN vs. 3 alternative multiple-annotator methods and the LR Concatenation logistic regression classifier on the biomedical *focus* classification using text annotations from 5 experts. Methods are sorted in legend of the figure according to their AUC value.

C. *The Relationship Between the Number of Low-quality Annotators and the Prediction Performance*

In the biomedical text data, all the 5 experts had comparable good quality and did not strongly reflect the diversity of annotators and other issues that we aim to address (e.g. the situation when low-quality annotators dominate). To further characterize our proposed AEFN algorithm, we simulated low-quality annotators that introduced a greater diversity of annotations. The sum of random sensitivity and specificity for each simulated annotator is selected to be between 1.1 and 1.3. We kept all the 5 expert annotators used in the experiments reported at the previous section and added more low quality annotators until 45 simulated annotators have been added (resulting in annotations by 50 sources). AUCs of the estimated ground truth vs. the number of low-quality annotators added for both *evidence* and *focus* classification tasks are shown in Fig. 3 and Fig. 4. All reported results are

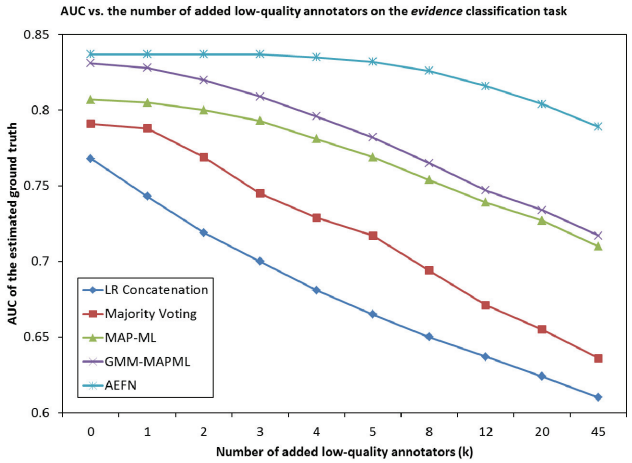

Figure 3. The AUC comparison of five integration methods vs. the number of added low-quality annotators on the *evidence* classification in biomedical texts. In the experiment annotations were provided by five experts and k random low-quality annotators where k is 0, 1, … 45 as shown at the figure. All plotted results are averages over 100 repetitions.

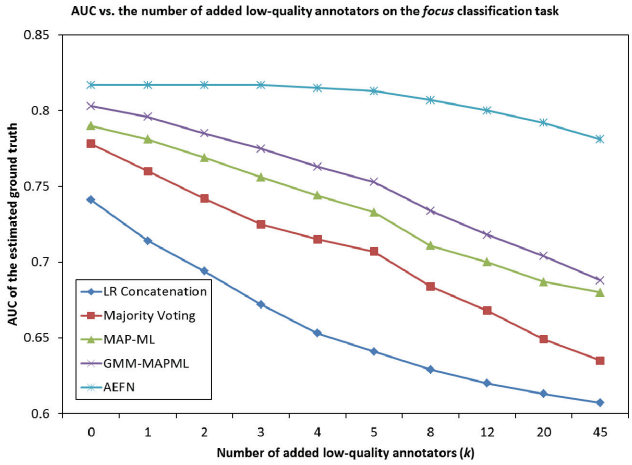

Figure 4. The AUC comparison of five integration methods vs. the number of added low-quality annotators on the *focus* classification in biomedical texts. In the experiment annotations were provided by five experts and k random low-quality annotators where k is 0, 1, … 45 as shown at the figure. All plotted results are averages over 100 repetitions.

averages over 100 repetitions, each with a random ordering of annotators to prevent bias due to the order in which simulated annotators are added.

Fig. 3 and Fig. 4 clearly provide evidence in support of our proposed AEFN algorithm: in both sets of the experiments, the AEFN has much better AUCs than all the competitor methods especially when the low-quality annotators dominate (when the number of low-quality annotators is greater than 5). Fig. 3 and Fig. 4 also show that the performance of LR Concatenation and Majority Voting degrades drastically as compared to the MAP-ML, GMM-MAPML and AEFN algorithms. The reason is that LR Concatenation and Majority Voting assume all annotators are equally good; they are very sensitive to low-quality annotators. MAP-ML and GMM-MAPML assign more weights to good annotators during the estimation as shown in the analyses of [9] and [11]. Our proposed AEFN goes even further: it filters low-quality annotators and estimate the ground truth based only on the good annotators. Thus, AEFN outperforms all competitor methods especially when the low-quality annotators dominate (AEFN has a good performance even with 90% of low-quality annotators, based on labels obtained by 5 experts and 45 low-quality annotators).

V. CASP9 PROTEIN DISORDER PREDICTION EXPERIMENT

Computational characterization of disorder in proteins is appealing due to the difficulties and high cost involved in experimental characterization of disorders. Treating an individual machine-based predictor as an annotator, the multiple-annotator methods can be used to build meta-predictors for protein disorder prediction. In this study we also experimentally validated the proposed algorithm on the CASP9 protein disorder prediction task. The results are reported in a supplement[4] due to lack of space.

VI. CONCLUSION

In this paper, we proposed a probabilistic algorithm for classification when given labels obtained by multiple noisy annotators but without any gold standard annotation. By combing annotator filtering to the process of consensus labeling, the proposed AEFN algorithm eliminates annotations provided by novices without using any true labels and estimates the consensus ground truth based only on higher quality annotations provided by experts. Experimental studies are conducted on biomedical text classification and CASP9 protein disorder prediction. The results provide evidence that AEFN works robustly and outperforms the majority voting baseline and previous multi-annotator methods (GMM-MAPML and MAP-ML) significantly, especially when low-quality annotators dominate. Moreover, AEFN also suggests competent annotators for each instance group (i.e., Gaussian component), which can be used for building more reliable biomedical predictive models.

ACKNOWLEDGEMENT

This project was funded in part under a grant with the GlaxoSmithKline LLC.

[4] Available at http://astro.temple.edu/~tua87106/BIBM_Suppl.pdf.

REFERENCES

[1] W.S. Cobb, R.M. Peindl, M. Zerey, A.M. Carbonell, and B.T. Heniford, "Mesh terminology 101", Hernia, 2009, 13(1), pp. 1-6.

[2] J.J. Hyun, and M. Lease, "Improving Consensus Accuracy via Z-score and Weighted Voting", in Proc. Human Computation Workshop, 2011, pp. 88-90.

[3] S. Chen, J. Zhang, G. Chen, and C. Zhang, "What if the irresponsible teachers are dominating", in Proc. AAAI Conference on Artificial Intelligence, 2010, pp. 419-424.

[4] O. Dekel, and O. Shamir, "Vox populi: Collecting high-quality labels from a crowd", in Proc. Conference on Learning Theory, 2009.

[5] V.S. Sheng, F.J. Provost, and P.G. Ipeirotis, "Get another label? Improving data quality and data mining using multiple, noisy labelers", in Proc. ACM Conference on Knowledge Discovery and Data Mining, 2008, pp. 614-622.

[6] H. Yang, A. Mityagin, K. Svore, and S. Markov, "Collecting High Quality Overlapping Labels at Low Cost", in Proc. ACM Conference on Information Retrieval, 2010, pp. 459-466.

[7] R. Snow, B. O'Connor, D. Jurafsky, and A.Y. Ng, "Cheap and fast - but is it good? Evaluating non-expert annotations for natural language tasks", in Proc. Conference on Empirical Methods on Natural Language Processing, 2008, pp. 254-263.

[8] V.C. Raykar, S. Yu, L.H. Zhao, A.K. Jerebko, C. Florin, G.H. Valadez, L. Bogoni, and L. Moy, "Supervised learning from multiple experts: whom to trust when everyone lies a bit", in Proc. International Conference on Machine Learning, 2009, pp. 889-896.

[9] P. Zhang, and Z. Obradovic, "Unsupervised integration of multiple protein disorder predictors", in Proc. IEEE Conference on Bioinformatics and Biomedicine, 2010, pp. 49-52.

[10] Y. Yan, R. Rosales, G. Fung, M.W. Schmidt, G.H. Valadez, L. Bogoni, L. Moy, and J.G. Dy, "Modeling annotator expertise: Learning when everybody knows a bit of something", in Proc. International Conference on Artificial Intelligence and Statistics, 2010, 932-939.

[11] P. Zhang, and Z. Obradovic, "Learning from Inconsistent and Unreliable Annotators by a Gaussian Mixture Model and Bayesian Information Criteria", in Proc. European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, 2011, 553-568.

[12] A. Rzhetsky, H. Shatkay, and W.J. Wilbur, "How to get the most out of your curation effort", PLoS. Comput. Biol, 2009, 5(5):e1000391.

[13] W.J. Wilbur, and W. Kim, "Improving a gold standard: treating human relevance judgments of MEDLINE document pairs", BMC Bioinformatics, 2011, 12(S3):S5.

[14] S. Mavandadi, S. Dimitrov, S. Feng, F. Yu, U. Sikora, O. Yaglidere, S. Padmanabhan, K. Nielsen, and A. Ozcan, "Distributed Medical Image Analysis and Diagnosis through Crowd-Sourced Games: A Malaria Case Study", PLoS ONE, 2012, 7(5): e37245.

[15] T. Ishida, and K. Kinoshita, "Prediction of disordered regions in proteins based on the meta approach", Bioinformatics, 2008, 24(11), pp. 1344-1348.

[16] M.J. Mizianty, W. Stach, K. Chen, K.D. Kedarisetti, F.M. Disfani, and L. Kurgan, "Improved sequence-based prediction of disordered regions with multilayer fusion of multiple information sources", Bioinformatics, 2010, 26(18), pp. i489-i496.

[17] L.P. Kozlowski, and J.M. Bujnicki, "MetaDisorder: a meta-server for the prediction of intrinsic disorder in proteins", BMC Bioinformatics, 2012, 13(1):111.

[18] P. Zhang, and Z. Obradovic, "Unsupervised Integration of Multiple Protein Disorder Predictors: The Method and Evaluation on CASP7, CASP8 and CASP9 Data", Proteome Science, 2011, 9(S1):S12.

[19] V.C. Raykar, and S. Yu, "Ranking annotators for crowdsourced labeling tasks", in Proc. Advances in Neural Information Processing Systems, 2011.

[20] G. Salton, and C. Buckley, Term-weighting approaches in automatic text retrieval", Information Processing & Management, 1988, 24(5), pp. 513-523.