Water from Two Rocks: Maximizing the Mutual Information

YUQING KONG, University of Michigan, USA
GRANT SCHOENEBECK, University of Michigan, USA

We build a natural connection between the learning problem, co-training, and forecast elicitation without verification (related to peer-prediction) and address them simultaneously using the same information theoretic approach.[1]

In co-training/multiview learning [5] the goal is to aggregate two views of data into a prediction for a latent label. We show how to optimally combine two views of data by reducing the problem to an optimization problem. Our work gives a unified and rigorous approach to the general setting.

In forecast elicitation without verification we seek to design a mechanism that elicits high quality forecasts from agents in the setting where the mechanism does not have access to the ground truth. By assuming the agents' information is independent conditioning on the outcome, we propose mechanisms where truth-telling is a strict equilibrium for both the single-task and multi-task settings. Our multi-task mechanism additionally has the property that the truth-telling equilibrium pays better than any other strategy profile and strictly better than any other "non-permutation" strategy profile.

CCS Concepts: • **Theory of computation → Algorithmic mechanism design**; **Unsupervised learning and clustering**; • **Information systems → Incentive schemes**;

Additional Key Words and Phrases: Peer prediction, co-training, information theory

1 INTRODUCTION

Co-training/multiview learning is a problem that asks to aggregate two views of data into a prediction for the latent label, and was first proposed by Blum and Mitchell [5]. Although co-training is an important learning problem, it lacks a unified and rigorous approach to the general setting. The current paper will make an innovative connection between the co-training problem and a peer prediction style mechanism design problem: forecast elicitation without verification, and develop a unified theory for both of them via the same information theoretic approach.

We use "forecasting whether a startup company will succeed" as our running example. We have two possible sources of information for each startup: the features X_A (e.g. products, business idea, target customer) of the startup; and the survey feedback X_B, collected from the crowd (e.g. a survey of amateur investors). Sometimes we have access to both the sources, and sometimes we have access to only one of the sources. We want to learn how to forecast the result Y (succeed/fail) of a startup company, using both or one of the sources.

We are given a set predictor candidates $\{P_A\}$ (e.g. a set of hypotheses) such that each predictor candidate P_A maps the features X_A to a forecast for the result Y of the startup (e.g. succeed with 73%

[1] A full version of the paper is available at https://arxiv.org/abs/1802.08887.

This work is supported by the National Science Foundation, under grant CAREER#1452915, CCF#1618187 and AitF#1535912.
Authors' addresses: Yuqing Kong, University of Michigan, Ann Arbor, USA, yuqkong@umich.edu; Grant Schoenebeck, University of Michigan, Ann Arbor, USA, schoeneb@umich.edu.

ACM EC'18, June 18–22, 2018, Ithaca, NY, USA. ACM ISBN 978-1-4503-5829-3/18/06…$15.00
https://doi.org/10.1145/3219166.3219194

probability, fail with 27% probability). We are also given a set predictor candidates $\{P_B\}$ (e.g. a set of aggregation algorithms like majority vote/weighted average) such that each predictor candidate P_B maps the survey feedback X_B to a forecast for the result Y. Our goal is to evaluate the performance of a specific pair P_A, P_B. The learning problem, learning how to forecast, can be reduced to this goal since if we know how to evaluate the two candidates P_A, P_B's performance, we can select the two candidates P_A^*, P_B^* which have the highest performance and use them to forecast.

Given a batch of past startup data each with the features X_A, the crowdsourced feedback X_B, and the result Y, we can evaluate the performance of the predictors through many existing measurements (e.g. proper scoring rules, loss functions). This evaluation method is related to the supervised learning setting. However, there may be only very few data points about the startups with results Y.[2] When we only use a few labeled data points to train the predictor, the predictor will likely over-fit. Thus, we can boldly ask:

(*Learning) *Can we evaluate the performance of the predictor candidates, as well as learn how to forecast the ground truth Y, without access to any data labeled with Y?* (See Figure 1)

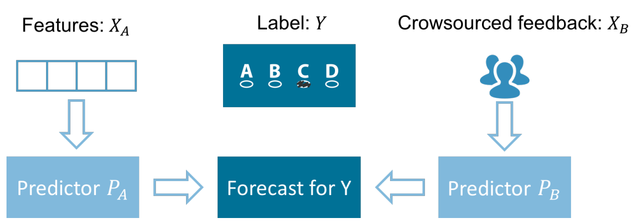

Fig. 1. Problem (*): Finding the common ground truth

It is impossible to solve this problem without making an additional assumption on the relationship between X_A, X_B and Y. However, it turns out we can solve this problem with a natural assumption, conditioning on Y, X_A and X_B are independent. This assumption states that Y contains all common information between X_A and X_B (see Section 3 for more discussion).

With this assumption, a naive approach is to learn the joint distribution of X_A and X_B using the past data, and then solve the relationship between Y and X_A, X_B by some calculations, using the fact that X_A and X_B are independent conditioning on Y. However, this naive approach will not work if either X_A or X_B has very high dimension. We will address this issue using learning methods. Before we go further on the learning problem, let's consider a corresponding mechanism design problem. In the scenario where the forecasts are provided by human beings, we want to ask a mechanism design problem:

(**Mechanism design) *Can we design proper* instant *reward schemes to incentivize high quality forecast for Y without instant access to Y?* (See Figure 2)

People will obtain instant payments from *instant* reward schemes. If we do not require the reward schemes to be instant, proper scoring rules will work by rewarding people in the future after Y is revealed. It turns out the above learning problem (*) and mechanism design problem (**) are essentially the same, since there is a natural correspondence between an evaluation of their performance and their rewards. The mechanism design applications still require the conditional independent assumption. To address the two problems, a first try would be rewarding the predictors according to their "agreement", since high quality predictors should have a lot of agreement with each other. However, if we train the predictors based on this criterion, then the output of the

[2]For example, if we focus on cryptographic or self-driving currencies, there are very few startups labeled with results.

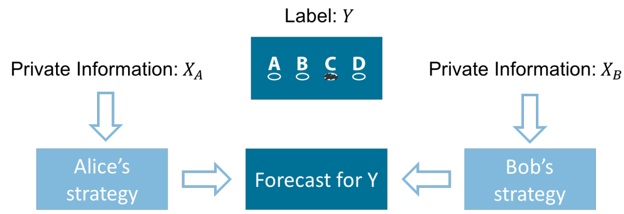

Fig. 2. Problem (**): Forecast elicitation

training process will be two meaningless constant predictors which perfectly agree with each other (e.g. always forecast 100% success). We call this problem the "naive agreement" issue.

Note that the mechanism design problem (**) is closely related to the peer prediction literature, incentivizing high quality information reports without verification. It is natural to leverage the techniques and insights from peer prediction to address problems (*) and (**). In fact, the peer prediction literature provides an information theoretic idea to address the "naive agreement" issue, that is, replacing "agreement" by mutual information. In the current paper, we will show that with a natural assumption, conditioning on Y, X_A, and X_B are independent, we can address problem (*) and (**) simultaneously via rewarding the predictors the mutual information between them and using the predictors' reward as the evaluation of their performance.

Our contribution. We build a natural connection between mechanism design and machine learning by simultaneously addressing a learning problem and a mechanism design problem in the context where ground truth is unknown, via the same information theoretic approach.

Learning We focus on the co-training problem [5]: learning how to forecast Y using two sources of information X_A and X_B, without access to any data labeled with ground truth Y (Section 3). By making a typical assumption in the co-training literature, conditioning on Y, X_A and X_B are independent, we reduce the learning problem to an optimization problem $\max_{P_A,P_B} MIG^f(P_A,P_B)$ such that solving the learning problem is equivalent to picking the P_A^*, P_B^* that maximize $MIG^f(P_A,P_B)$, i.e., the f-mutual information gain between P_A and P_B (Section 4). Formally, we define *the Bayesian posterior predictor* as the predictor that maps any input information $X = x$ to its Bayesian posterior forecast for $Y = y$, i.e., $Pr(Y = y|X = x)$. Then when both P_A, P_B are Bayesian posterior predictors, $MIG^f(P_A,P_B)$ is maximized and the maximal value is the f-mutual information between X_A and X_B. With an additional mild restriction on the prior, $MIG^f(P_A,P_B)$ is maximized if and only if both P_A, P_B are permuted versions of the Bayesian posterior predictor.
As is typical in related literature, we do not investigate the computation complexity or data requirement of the learning problem.
To the best of our knowledge, this is the first optimization goal in the co-training literature that guarantees that the maximizer corresponds to the Bayesian posterior predictor, without any additional assumption. Thus, our method optimally aggregates the two sources of information.

Mechanism design Consider the scenario where we elicit forecasts for ground truth Y from agents and pay agents immediately. Without access to Y, given the prior on the distribution of Y, i.e., $Pr[Y]$, [3] by assuming agents' private information are independent conditioning on Y, in the single-task setting (there is only a single forecasting task), we design a *strictly*

[3]This is not a very strong assumption since we do not need the knowledge of the joint distribution over the event and agents' private information.

truthful mechanism, the *common ground mechanism*, where truth-telling is a strict equilibrium (Section 5.2); in the multi-task (there are at least two a priori similar forecasting tasks) setting, we design a family of *focal* mechanisms, the *multi-task common ground mechanism MCG*(f)*s*, where the truth-telling equilibrium pays better than any other strategy profile and *strictly* higher than any non-permutation strategy profile (Section 5.1).

Technical contribution. Our main technical ingredient is a novel performance measurement, the f*-mutual information gain*, which is an unbiased estimator of the f-mutual information. To give a flavor of this measurement, we give an informal presentation here: both P_A and P_B are assigned a batch of forecasting tasks, the f-mutual information gain between P_A and P_B is

$$\begin{aligned}&\text{The agreements between } P_A\text{'s forecast and } P_B\text{'s forecast for the same task}\\ &- f^\star(\text{The agreements between } P_A\text{'s forecast and } P_B\text{'s forecast for different tasks})\end{aligned}$$

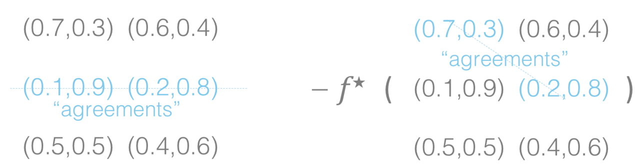

Fig. 3. An unbiased estimator of f-mutual information: f-mutual information gain. P_A and P_B are assigned three forecasting tasks. P_A's outputs are $(0.7, 0.3), (0.1, 0.9), (0.5, 0.5)$ and P_B's outputs are $(0.6, 0.4), (0.2, 0.8), (0.4, 0.6)$. To calculate the f-mutual information gain between them, we pick a task (e.g. Task no. 2) uniformly at random and calculate the "agreement" a_s between P_A and P_B's forecasts for this task; we also pick a pair of distinct tasks (i, j) uniformly at random (e.g. (Task no. 1, Task no. 2)) and calculate the "agreement" a_d between P_A's forecast for task i and P_B's forecast for this task j. The f-mutual information gain is then $a_s - f^\star(a_d)$. The formal definition (Section 4.1) actually uses the empirical expectations of a_s and $f^\star(a_d)$.

where $f^\star$ is the conjugate of the convex function f. With this measurement, two agreeing constant predictors have small gain since their outputs have large agreements for both the same task and different tasks. The formal definition will be introduced in Section 4.1 and the agreement measure is introduced in Definition 4.2.

The f-mutual information gain is conceptually similar to the correlation payment scheme proposed by Dasgupta and Ghosh [10] (in the binary choice setting), and Shnayder et al. [33] (in the multiple choice setting), which pays agents "the agreement for the same task *minus* the agreement for the distinct task". In Dasgupta and Ghosh [10] and Shnayder et al. [33], the payment scheme is designed for discrete signals and the measure of agreements is a simple indicator function. Kong and Schoenebeck [17] show that this correlation payment is related to a special f-mutual information. Thus, the f-mutual information gain can be seen as an extension of the correlation payment scheme that works for forecast reports.

1.1 Applications

In our startup running example, we consider the situation where one source of information is the features and another source of information is the crowdsourced feedback. In fact, our results apply to all kinds of information sources. For example, we can make both sources features or crowdsourced feedback. Different setups for the information sources and predictor candidates can bring different applications of our results.

Let's consider the "learning with noisy labels" problem where the labels in the training data are a noisy version of the ground truth labels Y and the noise is independent. We can map this problem into our framework by letting X_B be the noisy label of features X_A. That is, X_B is a noisy version of Y. Our framework guarantees that the Bayesian posterior predictor that forecasts Y using X_A must be part of a maximizer of the optimization problem. However, there are many other maximizers. For example, since X_A and X_B are independent conditioning X_B. The Bayesian posterior predictor that forecasts X_B using X_A is also part of a maximizer, since the scenario $Y = X_B$ also satisfies the conditional independence assumption. If X_B has much higher dimension than Y, we do not have this issue. But X_B has the same signal space with Y in the learning with noisy label problem. Thus, it's impossible to eliminate other maximizers without any side information here. With some side information (e.g. a candidate set $\mathcal{F}$, like linear regressions, that only contains our desired maximizer.), it's possible to obtain the Bayesian posterior predictor that forecasts Y using X_A. Note that our framework does not require a pre-estimation on the transition probability that transits the ground truth label Y to the noisy ground truth label X_B, since our framework has this transition probability, which corresponds to the predictor P_B, as parameters as well and learns the correct forecaster P_A and the transition probability P_B simultaneously.

Ratner et al. [29] propose a method to collect massive labels by asking the crowds to write heuristics to label the instances. Each instance is associated with many noisy labels outputted by the heuristics. In their setting, the crowds use a different source of information from the learning algorithm (e.g. the learning algorithm uses the biology description of the genes and the crowds use the scientific papers about the gene). Thus, the conditional independence assumption is natural here and we can map this setting's training problem into our framework. Ratner et al. [29] preprocess the collected labels to approximate ground truth by assuming a particular information structure model on the crowds. Our framework is model-free and does not need to preprocess the collected labels since we can learn the best forecaster (predictor P_A) and the best processing/aggregation algorithm (predictor P_B) simultaneously.

Moreover, since the highest evaluation value of the predictors P_A, P_B is the f-mutual information between X_A and X_B, our results provide a method to calculate the f-mutual information between any two sources of information X_A, X_B of any format. Kong and Schoenebeck [17] propose a framework for designing information elicitation mechanisms that reward truth-telling by paying each agent the f-mutual information between her report and her peers' report. Thus, the f-mutual information gain method can be combined with this framework to design information elicitation mechanisms when the information has a complicated format.

1.2 Related work

Learning. Co-training/multiview learning was first proposed by Blum and Mitchell [5] and explored by many works (e.g. Collins and Singer [7], Dasgupta et al. [11]). Li et al. [19], Xu et al. [37] give surveys on this literature. Although co-training is an important learning problem, it lacks a unified theory and a solid theoretic guarantee for the general model. Most traditional co-training methods require additional restrictions on the hypothesis space (e.g. weakly good hypotheses) to address the "naive agreement" issue and fail to deal with soft hypotheses. Soft hypotheses output a continuous signal (as opposed to hard hypothesis which output a discrete signal) and are typically required to fully aggregate the information from two sources. Becker [3] deals with a feature learning problem which is very similar to the co-training problem. Becker [3] seeks to maximize the Shannon mutual information between the output of two functions. However, their work only considers hard (not soft) hypotheses and lacks a solid theoretic analysis for the maximizer. Kakade and Foster [15] consider the multi-view regression and maximize the correlation between the two hypotheses. Their method captures the "mutual information" idea (in fact, correlation is a special f-mutual information [17]) but their model has a very specific set up and the analysis cannot be extended to other co-training problems.

In contrast, we propose a simple, powerful and general information theoretic framework, f-mutual information gain, that has a solid theoretic guarantee, works for soft hypothesis and addresses the "naive agreement" issue without any additional assumption.

Natarajan et al. [24], Sukhbaatar and Fergus [34] and many other works (e.g. [16, 32]) consider the learning with noisy labels problem. Natarajan et al. [24] consider binary labels and calibrate the original loss function such that the Bayesian posterior predictor that forecasts ground truth Y is a maximizer of the calibrated loss. Sukhbaatar and Fergus [34] extend this work to the multiclass setting. These works require additional estimation steps to learn the transition probability that transits the ground truth labels to the noisy labels and fix this transition probability in their calibration step. In contrast, by mapping this problem into our framework (Section 1.1), we do not need the additional estimation steps to make the calibrated forecaster part of a maximizer of our optimization problem, and can incorporate any kind of side information to learn the calibrated forecaster and true transition probability simultaneously.

Moreover, our results can handle more complicated setting where each instance is labeled by multiple labels. Rather than preprocessing the labels by a particular algorithm (e.g. majority vote, weighted average, spectral method) and assuming some information structure model among the crowds [29], our framework is model-free and can learn the best calibrated forecaster (predictor P_A) and the best processing algorithm (predictor P_B) simultaneously.

Raykar et al. [30] also *jointly* learn the calibrated forecaster and the distribution over the crowdsourced feedback and ground truth labels. Raykar et al. [30] uses the maximum likelihood estimator and assumes a simple generative model for the distribution over the crowdsourced feedback and the ground truth labels, which is conditioning the ground truth label, the crowdsourced feedback is drawn from a binomial distribution, while our framework is model-free.

Generative Adversarial Networks (GAN) [14] combine game theory and learning theory to make innovative progress. We also combine game theory and learning theory by proposing a peer prediction game between two predictors. The game in GAN is a zero-sum competitive game while the game in the current paper is collaborative.

Several learning problems (e.g. finding the pose of an object in an image [4], blind source separation [6]) use mutual information maximization (infomax) as their optimization goal. Some of these problems require data labeled with ground truth and some of them have a very different problem set up than our work.

We borrow the techniques about the duality of f-divergence from Nguyen et al. [25, 26]. Nguyen et al. [25] show a correspondence between the f-divergence and the surrogate loss in the *binary supervised learning* setting and Nguyen et al. [26] propose a way to estimate the f-divergence between two high dimensional random variables. We apply the duality of f-divergence to an unsupervised learning problem and not restricted to the binary setting.

We also differ from the crowdsourcing literature that infers ground truth answers from agents' reports (e.g. [38]) in the sense that their agents' reports are a simple choice (e.g. A, B, C, D) while in our setting, the report can come from a space larger than the space of ground truth answers, perhaps even a very high dimensional vector.

Mechanism design. Our mechanism design setting differ from the traditional peer prediction literature (e.g.[10, 17, 23, 28, 33]) since we are eliciting forecast rather than a simple signal. We can discretize the forecast report and apply the traditional peer prediction literature results. However, this will only provide approximated truthfulness and fail to design focal mechanisms which pay truth-telling *strictly* better than any other non-permutation equilibrium since the forecast is discretized, while our mechanisms are focal for ≥ 2 tasks setting.

Witkowski et al. [36] consider the forecast elicitation situation and assume that they have an unbiased estimator of the optimal forecast while we assume an additional conditional independence assumption but do not need the unbiased estimator.

Liu and Chen [20, 21] connect mechanism design with learning by using the learning methods to design peer prediction mechanisms. In the setting where several agents are asked to label a batch of instances, Liu and Chen [20] design a peer prediction mechanism where each agent is paid according to her answer and a reference answer generated by a classification algorithm using other agents' reports. Liu and Chen [21] also use surrogate loss functions as tools to develop a multi-task mechanism that achieves truthful elicitation in dominant strategy when the mechanism designer only has access to agents' reports. Instead of using learning methods to design the peer prediction mechanisms, our work uses peer prediction mechanism design techniques to address a learning problem. Moreover, our mechanism design problem has a very different set up from Liu and Chen [20, 21]. Agarwal and Agarwal [1] connect learning theory with information elicitation by showing the equivalence between the calibrated surrogate losses in *supervised* learning and the elicitation of certain properties of the underlying conditional label distribution. Both our learning problem and mechanism design problem have a very different set up from theirs.

Independent work. Like the current paper, McAllester [22] also uses Shannon mutual information to propose an information theoretic training objective that can deal with soft hypotheses/classifiers. However, the optimization functions from these two works are different. We also use a more general information measure, f-mutual information, which has Shannon mutual information as a special case, and provide a formal analysis for this general framework. Additionally, we propose an innovative connection between co-training and peer prediction.

2 PRELIMINARIES

Given a finite set $[N] := \{1, 2, ..., N\}$, for any function $\phi : [N] \mapsto \mathbb{R}$, we use $(\phi(y))_{y\in[N]}$ to represent the vector $(\phi(1), \phi(2), ..., \phi(N)) \in \mathbb{R}^N$. Given a finite set Σ, Δ_Σ is the set of all distributions over Σ.

2.1 f-divergence and Fenchel's duality

f-divergence [2, 9]. f-divergence $D_f : \Delta_\Sigma \times \Delta_\Sigma \mapsto \mathbb{R}$ is a non-symmetric measure of the difference between distribution $\mathbf{p} \in \Delta_\Sigma$ and distribution $\mathbf{q} \in \Delta_\Sigma$ and is defined to be

$$D_f(\mathbf{p}, \mathbf{q}) = \sum_{\sigma\in\Sigma} \mathbf{p}(\sigma) f\left(\frac{\mathbf{q}(\sigma)}{\mathbf{p}(\sigma)}\right)$$

where $f : \mathbb{R} \mapsto \mathbb{R}$ is a convex function and $f(1) = 0$.

Here we introduce two f-divergences in common use: KL divergence, and Total Variance Distance.

Example 2.1 (KL divergence). Choosing $-\log(x)$ as the convex function $f(x)$, f-divergence becomes KL divergence $D_{KL}(\mathbf{p}, \mathbf{q}) = \sum_\sigma \mathbf{p}(\sigma) \log \frac{\mathbf{p}(\sigma)}{\mathbf{q}(\sigma)}$

Example 2.2 (Total Variance Distance). Choosing $|x-1|$ as the convex function $f(x)$, f-divergence becomes Total Variance Distance $D_{tvd}(\mathbf{p}, \mathbf{q}) = \sum_\sigma |\mathbf{p}(\sigma) - \mathbf{q}(\sigma)|$

Definition 2.3 (Fenchel Duality [31]). Given any function $f : \mathbb{R} \mapsto \mathbb{R}$, we define its convex conjugate $f^\star$ as a function that also maps $\mathbb{R}$ to $\mathbb{R}$ such that

$$f^\star(x) = \sup_t tx - f(t).$$

Lemma 2.4 (Dual version of f-divergence [25, 26]).

$$D_f(\mathbf{p},\mathbf{q}) \geq \sup_{u\in\Sigma} \mathbb{E}_{\mathbf{p}} u - \mathbb{E}_{\mathbf{q}} f^{\star}(u) = \sup_{u\in\mathcal{G}} \sum_{\sigma} u(\sigma)\mathbf{p}(\sigma) - \sum_{\sigma} f^{\star}(u(\sigma))\mathbf{q}(\sigma)$$

where $\mathcal{G}$ is a set of functions that maps Σ to $\mathbb{R}$. The equality holds if and only if $u(\sigma) = u^*(\sigma) \in \partial f(\frac{\mathbf{p}(\sigma)}{\mathbf{q}(\sigma)})$, i.e., the subdifferential of f on value $\frac{\mathbf{p}(\sigma)}{\mathbf{q}(\sigma)}$.

We call $(u^*, f^{\star}(u^*))$ *a pair of best disinguishers*. This dual version of f-divergence is introduced by Nguyen et al. [25] and also plays a key role in the design of a type of generative adversarial networks, f-GANs [27].

2.2 f-mutual information

Given two random variables X, Y whose realization space are Σ_X and Σ_Y, let $\mathbf{U}_{X,Y}$ and $\mathbf{V}_{X,Y}$ be two probability measures where $\mathbf{U}_{X,Y}$ is the joint distribution of (X, Y) and $\mathbf{V}_{X,Y}$ is the product of the marginal distributions of X and Y. Formally, for every pair of $(x, y) \in \Sigma_X \times \Sigma_Y$,

$$\mathbf{U}_{X,Y}(X = x, Y = y) = \Pr[X = x, Y = y] \qquad \mathbf{V}_{X,Y}(X = x, Y = y) = \Pr[X = x]\Pr[Y = y].$$

If $\mathbf{U}_{X,Y}$ is very different from $\mathbf{V}_{X,Y}$, the mutual information between X and Y should be high since knowing X changes the belief for Y a lot. If $\mathbf{U}_{X,Y}$ equals to $\mathbf{V}_{X,Y}$, the mutual information between X and Y should be zero since X is independent with Y. Intuitively, the "distance" between $\mathbf{U}_{X,Y}$ and $\mathbf{V}_{X,Y}$ represents the mutual information between them.

Definition 2.5 (f-mutual information [17]). The f-mutual information between X and Y is defined as

$$MI^f(X;Y) = D_f(\mathbf{U}_{X,Y}, \mathbf{V}_{X,Y})$$

where D_f is f-divergence. f-mutual information is always non-negative [17].

f-mutual information is used in the peer prediction literature since if the information is measured by f-mutual information, any "data processing" on either of the random variables will decrease the amount of information crossing them. Thus, in peer prediction, if we pay agents according to the f-mutual information between her information and her peers' information, agents will be incentivized to report all information to maximize their payments[4].

Two examples of f-mutual information are Shannon mutual information [8] (Choosing f-divergence as KL divergence) and $MI^{tvd}(X;Y) := \sum_{x,y} |\Pr[X = x, Y = y] - \Pr[X = x]\Pr[Y = y]|$ (Choosing f-divergence as Total Variation Distance).

We define $K(X = x, Y = y)$ as the ratio between $U_{X,Y}(x, y)$ and $V_{X,Y}(x, y)$, i.e.,

$$K(X = x, Y = y) := \frac{\Pr[X = x, Y = y]}{\Pr[X = x]\Pr[Y = y]} = \frac{\Pr[Y = y|X = x]}{\Pr[Y = y]} = \frac{\Pr[X = x|Y = y]}{\Pr[X = x]}.$$

$K(X = x, Y = y)$ represents the "**pointwise mutual information**(PMI)" between $X = x$ and $Y = y$. Lemma 2.4 directly implies:

Lemma 2.6 (Dual version of f-mutual information).

$$MI^f(X;Y) \geq \sup_{u\in\mathcal{G}} \mathbb{E}_{U_{X,Y}} u - \mathbb{E}_{V_{X,Y}} f^{\star}(u)$$

where $\mathcal{G}$ is a set of functions that maps $\Sigma_X \times \Sigma_Y$ to $\mathbb{R}$.

The equality holds if and only if $u(x, y) = u^*(x, y) \in \partial f(K(X = x, Y = y))$.

[4]In the current paper, we do not directly use the data processing inequality of f-mutual information. Thus, we omit the formal introduction here. The interested reader is refer to Kong and Schoenebeck [17].

f-divergence	$f(t)$	$u^*(x,y) = \partial f(K(x,y))$	$f^\star(u^*(x,y))$
Total Variation Distance	$\|t-1\|$	$\text{sign}(\log K(x,y))$	$\text{sign}(\log K(x,y))$
KL divergence	$t\log t$	$1+\log K(x,y)$	$K(x,y)$
Reverse KL	$-\log t$	$-\frac{1}{K(x,y)}$	$-1+\log K(x,y))$
Pearson χ^2	$(t-1)^2$	$2(K(x,y)-1)$	$(K(x,y))^2-1$
Squared Hellinger	$(\sqrt{t}-1)^2$	$1-\sqrt{\frac{1}{K(x,y)}}$	$\sqrt{K(x,y)}-1$

Table 1. Reference for common f-divergences and corresponding pairs of best distinguishers $(u^*(x,y), f^\star(u^*(x,y)))$ of f-mutual information. $K(x,y) = K(X=x, Y=y)$ (PMI).

2.3 Proper scoring rules

A scoring rule $PS : \Sigma \times \Delta_\Sigma \mapsto \mathbb{R}$ [13, 35] takes in a signal $\sigma \in \Sigma$ and a distribution over signals $\mathbf{p} \in \Delta_\Sigma$ and outputs a real number. A scoring rule is *proper* if, whenever the first input is drawn from a distribution $\mathbf{p}$, then $\mathbf{p}$ will maximize the expectation of PS over all possible inputs in Δ_Σ to the second coordinate. A scoring rule is called *strictly proper* if this maximum is unique. We will assume throughout that the scoring rules we use are strictly proper. Slightly abusing notation, we can extend a scoring rule to be $PS : \Delta_\Sigma \times \Delta_\Sigma \mapsto \mathbb{R}$ by simply taking $PS(\mathbf{p}, \mathbf{q}) = \mathbb{E}_{\sigma \leftarrow \mathbf{p}}(\sigma, \mathbf{q})$. We note that this means that any proper scoring rule is linear in the first term.

Example 2.7 (Log Scoring Rule [13, 35]). Fix an outcome space Σ for a signal σ. Let $\mathbf{q} \in \Delta_\Sigma$ be a reported distribution. The Logarithmic Scoring Rule maps a signal and reported distribution to a payoff as follows:

$$LSR(\sigma, \mathbf{q}) = \log(\mathbf{q}(\sigma)).$$

Let the signal σ be drawn from some random process with distribution $\mathbf{p} \in \Delta_\Sigma$.
Then the expected payoff of the Logarithmic Scoring Rule

$$\mathbb{E}_{\sigma \leftarrow \mathbf{p}}[LSR(\sigma, \mathbf{q})] = \sum_\sigma \mathbf{p}(\sigma)\log \mathbf{q}(\sigma) = LSR(\mathbf{p}, \mathbf{q})$$

This value will be maximized if and only if $\mathbf{q} = \mathbf{p}$.

2.4 Property of the pointwise mutual information

We will introduce a simple property of the pointwise mutual information that we will use multiple times in the future. In addition to several different formats of the pointwise mutual information (e.g. joint distribution/product of the marginal distributions, posterior/prior), if there exists a latent random variable Y such that random variable X_A and random variable X_B are independent conditioning on Y, we can also represent the pointwise mutual information between X_A and X_B by the "agreement" between the "relationship" between X_A and Y, and the "relationship" between X_B and Y.

Claim 2.8. *When random variables X_A, X_B are independent conditioning on Y,*

$$\begin{aligned} K(X_A = x_A, X_B = x_B) &= \sum_y \Pr[Y=y] K(X_A = x_A, Y = y) K(X_B = x_B, Y = y) \\ &= \sum_y \Pr[Y = y | X_A = x_A] K(X_B = x_B, Y = y) \end{aligned}$$

$$= \sum_{y} \frac{\Pr[Y = y|X_A = x_A]\Pr[Y = y|X_B = x_B]}{\Pr[Y = y]}.$$

We defer the proof to the full version.

3 GENERAL MODEL AND ASSUMPTIONS

Let X_A, X_B, Y be three random variables and we define prior Q as the joint distribution over X_A, X_B, Y. We want to forecast the ground truth Y whose realization is a signal in a finite set Σ. X_A, X_B are two sources of information that are related to Y. X_A's realization is a signal in a finite set Σ_A. X_B's realization is a signal in a finite set Σ_B. We may have access to both of the realizations of X_A and X_B or only one of them. Thus, we need to learn the relationship between X_A, X_B and Y to forecast Y. It's impossible to learn by only accessing the samples of X_A, X_B without additional assumption. We make the following conditional independence assumption:

Assumption 3.1 (Conditional independence). *We assume that conditioning on Y, X_A, and X_B are independent.*

Intuitively, Y can be seen as the "intersection" between X_A and X_B. To better understand this assumption and its limitations we return to our running example where the variable Y is the success of a start-up. In this case, if both X_A and X_B contain the sex of the CEO (which we assume is independent of Y), then this assumption will not hold. To make it hold, either Y would need to be redefined to contain the sex of the CEO, or this information would need to be removed from either X_A or X_B. For the mechanism design application, if the assumption is violated, for example both agents are sexists and forecast using the sex of the CEO, then it is impossible to avoid paying them for this useless/harmful information.

3.1 Well-defined and stable prior

We call Z a *solution* if conditioning on Z, X_A, and X_B are independent. Y is a solution. However, there are a lot of solutions. For example, conditioning on X_A or X_B, X_A and X_B are independent, which means X_A and X_B are both solutions. Thus, we have an additional restriction on the prior: well-defined prior and stable prior.

We will need restrictions on the prior when we analyze the strictness of our learning algorithm/mechanism. Readers can skip this section without losing the core idea of our results.

To infer the relationship between Y and X_A, X_B with only samples of X_A, X_B, we cannot do better than to just solve the system of equations (1), given the joint distribution over X_A, X_B: Q. Our goal is to obtain the Bayesian posterior predictor. Thus, we list a system that the Bayesian posterior predictor satisfies. The system below equations involve variables $\{\mathbf{a}^{x_A}, \mathbf{b}^{x_B} \in \Delta_\Sigma\}_{x_A \in \Sigma_A, x_B \in \Sigma_B}$, and $\mathbf{r} \in \Delta_\Sigma$. We insist $a_y^{x_A} = \Pr[Y = y|X_A = x_A]$, $b_y^{x_B} = \Pr[Y = y|X_B = x_B]$ and $r_y = \Pr[Y = y]$ is a solution and we call it the *desired* solution.

$$\begin{aligned} &\mathcal{S}(\{\mathbf{a}^{x_A},\mathbf{b}^{x_B}\}_{x_A \in \Sigma_A, x_B \in \Sigma_B}, \mathbf{r}) \\ &:= \left\{ \sum_{y \in \Sigma} \frac{a_y^{x_A} b_y^{x_B}}{r_y} - K(X_A = x_A, X_B = x_B) \right\}_{x_A \in \Sigma_A, x_B \in \Sigma_B} = 0 \end{aligned} \tag{1}$$

Claim 2.8 shows the above system has the desired solution.

Note that any permutation of a solution is still a valid solution[5]. Since we cannot do better than to solve the above system, if the above system only has one "unique" solution, in the sense that any two solutions are permuted version of each other, we call the prior Q a well-defined prior. Formally,

Definition 3.2 (Well-defined). A prior Q is well-defined if for any two solutions $\{\mathbf{a}^{x_A}, \mathbf{b}^{x_B}\}_{x_A \in \Sigma_A, x_B \in \Sigma_B}$, $\mathbf{r}$ and $\{\mathbf{c}^{x_A}, \mathbf{d}^{x_B}\}_{x_A \in \Sigma_A, x_B \in \Sigma_B}$, $\mathbf{r}'$ of the system of equations (1), there exists a permutation $\pi : \Sigma \mapsto \Sigma$ such that $\mathbf{r} = \pi \mathbf{r}'$ for any x_A, x_B, $\mathbf{a}^{x_A} = \pi \mathbf{c}^{x_A}$, $\mathbf{b}^{x_B} = \pi \mathbf{d}^{x_B}$.

The well-defined prior exist since intuitively, if $|\Sigma_A|$ and $|\Sigma_B|$ are high and $|\Sigma|$ is low, it is likely Y is the "unique intersection" since the number of constraints of the system will be much greater than the number of variables.

We say a prior is stable if fixing part of the desired solution of the system (1), in order to make it still a solution of the system, other parts of the desired solution should also be fixed.

Definition 3.3 (Stable). A prior Q is stable if fixing $a_y^{x_A} = \Pr[Y = y|X_A = x_A]$ and $r_y = \Pr[Y = y]$, the system (1) $\mathcal{S}(\{\mathbf{a}^{x_A}, \mathbf{b}^{x_B}\}_{x_A \in \Sigma_A, x_B \in \Sigma_B}, \mathbf{r}) = 0$ has unique solution $\mathbf{b}^{x_A}$ such that $b_y^{x_B} = \Pr[Y = y|X_B = x_B]$; and fixing $b_y^{x_B} = \Pr[Y = y|X_B = x_B]$ and $r_y = \Pr[Y = y]$, the system (1) $\mathcal{S}(\{\mathbf{a}^{x_A}, \mathbf{b}^{x_B}\}_{x_A \in \Sigma_A, x_B \in \Sigma_B}, \mathbf{r}) = 0$ has unique solution $\mathbf{a}^{x_A}$ such that $a_y^{x_A} = \Pr[Y = y|X_A = x_A]$.

We require stable priors when we design *strictly* truthful mechanisms.

3.2 Predictors

This section gives the definition of predictors. We have two sets of samples $S_A := \{x_A^\ell\}_{\ell \in \mathcal{L}_A}$ and $S_B := \{x_B^\ell\}_{\ell \in \mathcal{L}_B}$ which are i.i.d samples of X_A and X_B respectively. For $\ell \in \mathcal{L}_A \cap \mathcal{L}_B$, (x_A^ℓ, x_B^ℓ)s are i.i.d samples of the joint random variable (X_A, X_B).

A predictor $P_A : \Sigma_A \mapsto \Delta_\Sigma$ for X_A maps $x_A \in \Sigma$ to a forecast $P_A(x_A)$ for ground truth Y. We similarly define the predictors for X_B. We define *the Bayesian posterior predictor* as the predictor that maps any input information $X = x$ to its Bayesian posterior forecast for $Y = y$, i.e., $Pr(Y = y|X = x)$.

With the conditional independence assumption, we have

$$\begin{aligned}
\Pr[Y|X_A, X_B] &= \frac{\Pr[Y, X_A, X_B]}{\Pr[X_A, X_B]} \\
&= \frac{\Pr[Y]\Pr[X_A|Y]\Pr[X_B|Y]}{\Pr[X_A, X_B]} && \text{(conditional independence)} \\
&= \frac{\Pr[Y|X_A]\Pr[Y|X_B]}{K(X_A, X_B)\Pr[Y]} && \text{($K(X_A, X_B)$ is the pointwise mutual information.)}
\end{aligned}$$

When we have access to both the sources where $X_A = x_A$ and $X_B = x_B$, given the prior of the ground truth Y, we can construct an aggregated forecast for $Y = y$ using P_A, P_B:

$$\frac{P_A(x_A)P_B(x_B)}{\Pr[Y = y]} \cdot \text{normalization}$$

In this case, if both P_A and P_B are the Bayesian posterior predictor, the aggregated forecast is the Bayesian posterior predictor as well. Thus, it's sufficient to only train P_A and P_B. In the rest sections, we will show how to train P_A and P_B (Section 4), given the two sets of samples S_A and S_B, as well as how to incentivize high quality predictors from the crowds (Section 5).

[5] We may be able to distinguish a solution with its permuted version if we have some side information (e.g. the prior of Y/a few (x_A, x_B, y) samples).

4 CO-TRAINING: FINDING THE COMMON GROUND TRUTH

We have a set of candidates $\mathcal{H}_A$ for the predictor for X_A and a set of candidates $\mathcal{H}_B$ for the predictor for X_B. We sometimes call each predictor candidate *a hypothesis*. Given the two sets of samples $S_A = \{x_A^\ell\}_{\ell\in\mathcal{L}_A}$ and $S_B = \{x_B^\ell\}_{\ell\in\mathcal{L}_B}$, our goal is to figure out the best hypothesis in $\mathcal{H}_A$ and the best hypothesis in $\mathcal{H}_B$ simultaneously. Thus, we need to design proper "loss function" such that the best hypotheses minimize the loss. In fact, we will show how to design a proper "reward function" such that the best hypotheses maximize the reward.

4.1 f-mutual information gain

f-mutual information gain $MIG^f(R)$ (Figure 3).

Hypothesis We are given $\mathcal{H}_A = \{h_A : \Sigma_A \mapsto \Delta_\Sigma\}$, $\mathcal{H}_B = \{h_B : \Sigma_B \mapsto \Delta_\Sigma\}$: the set of hypotheses/predictor candidates for X_A and X_B, respectively.

Gain Given reward function $R : \Delta_\Sigma \times \Delta_\Sigma \mapsto \mathbb{R}$,
for each $\ell \in \mathcal{L}_A \cap \mathcal{L}_B$, reward "the amount of agreement" between the two predictor candidates' predictions for task ℓ, i.e.,

$$R(h_A(x_A^\ell), h_B(x_B^\ell));$$

for each distinct pair (ℓ_A, ℓ_B), $\ell_A \in \mathcal{L}_A, \ell_B \in \mathcal{L}_B, \ell_A \neq \ell_B$, punish both predictor candidates "the amount of agreement" between their predictions for a pair of distinct tasks (ℓ_A, ℓ_B), i.e.,

$$f^\star(R(h_A(x_A^{\ell_A}), h_B(x_B^{\ell_B})).$$

The f-mutual information gain $MIG^f(R)$ that is corresponding to the reward function R is

$$MIG^f(R(h_A, h_B))_{|S_A,S_B} = \frac{1}{|\mathcal{L}_A \cap \mathcal{L}_B|} \sum_{\ell\in\mathcal{L}_A\cap\mathcal{L}_B} R(h_A(x_A^\ell), h_B(x_B^\ell))$$
$$- \frac{1}{|\mathcal{L}_A||\mathcal{L}_B| - |\mathcal{L}_A \cap \mathcal{L}_B|^2} \sum_{\ell_A\in\mathcal{L}_A,\ell_B\in\mathcal{L}_B,\ell_A\neq\ell_B} f^\star(R(h_A(x_A^{\ell_A}), h_B(x_B^{\ell_B})))$$

Lemma 4.1. *The expected total f-mutual information gain is maximized over all possible R, h_A, and h_B if and only if for any $(x_A, x_B) \in \Sigma_A \times \Sigma_B$,*

$$R(h_A(x_A), h_B(x_B)) \in \partial f(K(x_A, x_B)).$$

The maximum is $MI^f(X_A; X_B)$.

Proof. $(x_A^\ell, x_B^\ell)_\ell$ are i.i.d. realizations of (X_A, X_B). Therefore, the expected f-mutual information gain is $\mathbb{E}_{U_{X_A,X_B}} R - \mathbb{E}_{V_{X_A,X_B}} f^\star(R)$. The results follow from Lemma 2.6. □

Although any reward function corresponds to an f-mutual information gain function, we need to properly design the reward function R such that, fixing R, there exist hypotheses to maximize the corresponding f-mutual information gain $MIG^f(R)$ to the f-mutual information between the two sources. We will use the intuition from Lemma 4.1 to design such reward functions R in the next section.

4.2 Maximizing the f-mutual information gain

In this section, we will construct a special reward function R^f and then show that the maximizers of the corresponding f-mutual information gain $MIG^f(R^f)$ are the Bayesian posterior predictors.

Definition 4.2 (R^f). We define reward function R^f as a function that maps the two hypotheses' outputs $\mathbf{p}_1, \mathbf{p}_2 \in \Delta_\Sigma$ and the vector $\mathbf{p} \in \Delta_\Sigma$ to

$$R^f(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}) := g\left(\sum_y \frac{\mathbf{p}_1(y)\mathbf{p}_2(y)}{\mathbf{p}(y)}\right)$$

where $g(t) \in \partial f(t), \forall t$. When f is differentiable,

$$R^f(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}) := f'\left(\sum_y \frac{\mathbf{p}_1(y)\mathbf{p}_2(y)}{\mathbf{p}(y)}\right).$$

With this definition of the reward function, fixing $\mathbf{p} \in \Delta_\Sigma$ which can be seen as the prior over Y, the "amount of agreement" between two predictions $\mathbf{p}_1, \mathbf{p}_2$ are an increasing function g of

$$\sum_y \frac{\mathbf{p}_1(y)\mathbf{p}_2(y)}{\mathbf{p}(y)},$$

which is intuitive and reasonable. The increasing function g is the derivative of the convex function f. By carefully choosing convex function f, we can use any increasing function g here.

THEOREM 4.3. *With the conditional independent assumption on X_A, X_B, Y, given the samples S_A, S_B, given a convex function f, we define the optimization goal as the expected f-mutual information gain with reward function R^f, i.e.,*

$$MIG^f(h_A, h_B, \mathbf{p}) := \mathbb{E}_{X_A, X_B} MIG^f(R^f(h_A, h_B, \mathbf{p}))_{|S_A, S_B}$$

and optimize over all possible hypotheses $h_A : \Sigma_A \mapsto \Delta_\Sigma$, $h_B : \Sigma_B \mapsto \Delta_\Sigma$ and distribution vectors $\mathbf{p} \in \Delta_\Sigma$. We have

Solution→Maximizer: *any solution Z corresponds to a maximizer of $MIG^f(h_A, h_B, \mathbf{p})$[6]: for any solution Z,*

$$h_A^*(x_A) := (\Pr[Z = y | X_A = x_A])_y \qquad h_B^*(x_B) := (\Pr[Z = y | X_B = x_B])_y$$ [7]

and the prior over Z, $\Pr[Z = y]_y$, is the maximizer of $MIG^f(h_A, h_B, \mathbf{p})$ and the maximum is $MI^f(X_A; X_B)$;

Maximizer→(Permuted) Ground truth *when the prior is well-defined, f is differentiable, and f' is invertible, any maximizer of $MIG^f(h_A, h_B, \mathbf{p})$ corresponds to the (possibly permuted) ground truth Y: for any maximizer $(h_A^*(\cdot), h_B^*(\cdot), \mathbf{p}^*)$ of $MIG^f(h_A, h_B, \mathbf{p})$, there exists a permutation π such that*

$$h_A^*(x_A) := (\Pr[\pi(Y) = y | X_A = x_A])_y \qquad h_B^*(x_B) := (\Pr[\pi(Y) = y | X_B = x_B])_y$$

and $\mathbf{p}^* = \Pr[\pi(Y) = y]_y$.

The above theorem neither investigates computation complexity (which may be affected by the choice of f), data requirements, nor the choice of the hypothesis class for practical implementation (see Section 6 for more discussion).

[6]Given the prior over Y, we can fix $\mathbf{p}$ as the prior over Y. Without knowing the prior over Y, $\mathbf{p}$ becomes a variable of the optimization goal and helps us learn the prior over Y.

[7]Recall that we use $(\phi(y))_{y \in [N]}$ to represent the vector $(\phi(1), \phi(2), \ldots, \phi(N)) \in \mathbb{R}^N$.

Proof for Theorem 4.3. Lemma 4.1 shows that the expected f-mutual information gain is maximized if and only if for any (x_A, x_B),

$$R^f(h_A^*(x_A), h_B^*(x_B), \mathbf{p}^*) \in \partial f(K(x_A, x_B)).$$

(1) *Solution⟶Maximizer:* For any solution Z, we can construct

$$h_A^*(x_A) := (\Pr[Z = y | X_A = x_A])_y \qquad h_B^*(x_B) := (\Pr[Z = y | X_B = x_B])_y$$

and $\mathbf{p}^* = \Pr[Z = y]_y$. Then

$$\begin{aligned} R^f(h_A^*(x_A), h_B^*(x_B), \mathbf{p}^*) &\in \partial f\left(\sum_y \frac{\Pr[Z = y | X_A = x_A]\Pr[Z = y | X_B = x_B]}{\Pr[Z = y]}\right) \\ &= \partial f(K(x_A, x_B) && \text{(Claim 2.8)} \end{aligned}$$

Thus, based on Lemma 4.1, any solution Z corresponds to a maximizer of the optimization goal.

(2)*Maximizer⟶(Permuted) Ground truth:* For any maximizer $(h_A^*(\cdot), h_B^*(\cdot), \mathbf{p}^*)$ of the optimization goal, when f is differentiable, Lemma 4.1 shows that

$$R^f(h_A^*(x_A), h_B^*(x_B), \mathbf{p}^*) = f'(K(x_A, x_B)).$$

When f' is invertible, we have

$$\sum_y \frac{h_A^*(x_A)(y) h_B^*(x_B)(y)}{\mathbf{p}^*(y)} = K(x_A, x_B)$$

for all x_A, x_B.

Thus, $\{(h_A^*(x_A), h_B^*(x_B), \mathbf{p}^*)\}_{x_A, x_B}$ is actually the solution of the system (1). When the prior is well-defined, there exists a permutation π such that

$$h_A^*(x_A) := (\Pr[\pi(Y) = y | X_A = x_A])_y \qquad h_B^*(x_B) := (\Pr[\pi(Y) = y | X_B = x_B])_y$$

and $\mathbf{p}^* = \Pr[\pi(Y) = y]_y$ where Y is the ground truth.

□

5 FORECAST ELICITATION WITHOUT VERIFICATION

This section considers the setting where the forecasts are provided by the crowds and we want to incentivize high quality forecast by providing an instant reward without instant access to the ground truth.

There is a forecasting task. Alice and Bob have private information $X_A, X_B = x_A \in \Sigma_A, x_B \in \Sigma_B$ correspondingly and are asked to forecast the ground truth $Y = y$. We denote $(\Pr[Y = y | X_A = x_A])_y$, $(\Pr[Y = y | X_B = x_B])_y$ by $\mathbf{p}_{x_A}, \mathbf{p}_{x_B}$ correspondingly. Alice and Bob are asked to report their Bayesian forecast $\mathbf{p}_{x_A}, \mathbf{p}_{x_B}$. We denote their actual reports by $\hat{\mathbf{p}}_{x_A}$ and $\hat{\mathbf{p}}_{x_B}$. Without access to the realization of Y, we want to incentivize both Alice and Bob play *truth-telling* strategies, i.e., honestly reporting their forecast $\mathbf{p}_{x_A}, \mathbf{p}_{x_B}$ for Y.

We define the *strategy* of Alice as a mapping s_A from x_A (private signal) to a probability distribution over the space of all possible forecast for random variable Y. Analogously, we define Bob's strategy s_B. Note that essentially each (possibly mixed) strategy s_A can be seen as a (possibly random) predictor P_A where $P_A(x_A)$ is a random forecast drawn from distribution $s_A(x_A)$. In particular, the truthful strategy corresponds to the Bayesian posterior predictor.

We say agents play a *permutation strategy profile* if there exists permutation $\pi : \Sigma \mapsto \Sigma$ such that each agent always reports $\pi\mathbf{p}$ given her truthful report is $\mathbf{p}$.

Note that without any side information about Y, we cannot distinguish the scenario where agents are honest and the scenario where agents play a permutation strategy profile. Thus, it is too much to ask truth-telling to be strictly better than any other strategy profile. The focal property defined in the following paragraph is the optimal property we can obtain.

Mechanism Design Goals.

(Strictly) Truthful Mechanism $\mathcal{M}$ is (strictly) truthful if truth-telling is a (strict) equilibrium.

Focal Mechanism $\mathcal{M}$ is focal if it is strictly truthful and each agent's expected payment is maximized if agents tell the truth; moreover, when agents play a non-permutation strategy profile, each agent's expected payment is *strictly* less.

We consider two settings:

Multi-task Each agent is assigned several independent a priori similar forecasting tasks in a random order and is asked to report her forecast for each task.

Single-task All agents are asked to report their forecast for the same single task.

In the single-task setting, it's impossible to design focal mechanisms since agents can collaborate to pick an arbitrary $y^* \in \Sigma$ and pretend that they know $Y = y^*$. However, we will show we can design strictly truthful mechanism in the single-task setting. In the multi-task setting, since agents may be assigned different tasks and the tasks show in random order, they cannot collaborate to pick an arbitrary $y^* \in \Sigma$ for each task. In fact, we will show if the number of tasks is greater or equal to 2, we can design a family of focal mechanisms.

Achieving the focal goal in the multi-task setting is very similar to what we did in finding the common ground truth. Note that in the forecast elicitation problem, incentivizing a truthful strategy is equivalent to incentivizing the Bayesian posterior predictor. Thus, we can directly use the f-mutual information gain as the reward in the multi-task setting. Achieving the strictly truthful goal in the single-task setting is more tricky and we will return to it later.

5.1 Multi-task: focal forecast elicitation without verification

We assume Alice is assigned tasks set $\mathcal{L}_A$ and Bob is assigned tasks set $\mathcal{L}_B$. For each task ℓ, Alice's private information is x_A^ℓ and Bob's private information is x_B^ℓ. The ground truth of this task is y^ℓ.

Multi-task common ground mechanism MCG(f). Given the prior distribution over Y, a convex and *differentiable* function f whose convex conjugate is $f^\star$,

Report for each task $\ell \in \mathcal{L}_A$, Alice is asked to report $\mathbf{p}_{x_A{}^\ell} := (\Pr[Y = y|x_A^\ell])_y$; for each task $\ell \in \mathcal{L}_B$, Bob is asked to report $\mathbf{p}_{x_B{}^\ell} := (\Pr[Y = y|x_B^\ell])_y$. We denote their actual reports by $\hat{\mathbf{p}}^\ell_{x_A{}^\ell}$ and $\hat{\mathbf{p}}^\ell_{x_B{}^\ell}$.

Payment For each $\ell \in \mathcal{L}_A \cap \mathcal{L}_B$, reward both Alice and Bob "the amount of agreement" between their forecast in task ℓ, i.e.,

$$R(\hat{\mathbf{p}}^\ell_{x_A{}^\ell}, \hat{\mathbf{p}}^\ell_{x_B{}^\ell});$$

for each pair of distinct tasks (ℓ_A, ℓ_B), $\ell_A \in \mathcal{L}_A, \ell_B \in \mathcal{L}_B, \ell_A \neq \ell_B$, punish both Alice and Bob "the amount of agreement" between their forecast in distinct tasks (ℓ_A, ℓ_B), i.e.,

$$f^\star(R(\hat{\mathbf{p}}^{\ell_A}_{x_A{}^{\ell_A}}, \hat{\mathbf{p}}^{\ell_B}_{x_B{}^{\ell_B}})).$$

In total, both Alice and Bob are paid

$$\frac{1}{|\mathcal{L}_A \cap \mathcal{L}_B|} \sum_{\ell \in \mathcal{L}_A \cap \mathcal{L}_B} R(\hat{\mathbf{p}}^\ell_{x_A{}^\ell}, \hat{\mathbf{p}}^\ell_{x_B{}^\ell})$$

$$-\frac{1}{|\mathcal{L}_A||\mathcal{L}_B|-|\mathcal{L}_A\cap\mathcal{L}_B|^2}\sum_{\ell_A\in\mathcal{L}_A,\ell_B\in\mathcal{L}_B,\ell_A\neq\ell_B} f^{\star}(R(\hat{\mathbf{p}}^{\ell_A}_{x_A{}^{\ell_A}},\hat{\mathbf{p}}^{\ell_B}_{x_B{}^{\ell_B}})$$

where

$$R(\mathbf{p}_1,\mathbf{p}_2):=f'(\sum_y\frac{\mathbf{p}_1(y)\mathbf{p}_2(y)}{\Pr[Y=y]}).$$

We do not want agents to collaborate with each other based on the index of the task or other information in addition to the private information. Thus, we make the following assumption to guarantee the index of the task is meaningless for all agents.

Assumption 5.1 (A priori similar and random order). *For each task ℓ, fresh i.i.d. realizations of $(X_A,X_B,Y)=(x_A^{\ell},x_B^{\ell},y^{\ell})$ are generated. All tasks appear in a random order, independently drawn for each agent.*

Theorem 5.2. *With the conditional independence assumption, and a priori similar and random order assumption, when the prior Q is stable and well-defined, given the prior distribution over the Y, given a differential convex function f whose derivative f' is invertible, if $\max\{|\mathcal{L}_A|,|\mathcal{L}_B|\}\geq 2$, then $MCG(f)$ is focal. When both Alice and Bob are honest, each of them's expected payment in $MCG(f)$ is $MI^f(X_A;X_B)$.*

The non-negativity of MI^f implies that agents are willing to participate in the mechanism. Like Theorem 4.3, in order to show Theorem 5.2, we need to first introduce a lemma which is very similar to Lemma 4.1.

Lemma 5.3. *With the conditional independence assumption, the expected total payment is maximized over Alice and Bob's strategies if and only if $\forall \ell_1\in\mathcal{L}_A,\ell_2\in\mathcal{L}_B$, for any $(x_A^{\ell_1},x_B^{\ell_2})\in\Sigma_A\times\Sigma_B$,*

$$R(\hat{\mathbf{p}}^{\ell_1}_{x_A{}^{\ell_1}},\hat{\mathbf{p}}^{\ell_2}_{x_B{}^{\ell_2}})=f'(K(x_A^{\ell_1},x_B^{\ell_2})).$$

The maximum is $MI^f(X_A;X_B)$.

The proofs of Lemma 5.3 and Theorem 5.2 are very similar with Lemma 4.1 and Theorem 4.3. We defer the formal proofs to the full version.

5.2 Single-task: strictly truthful forecast elicitation without verification

This section introduces the strictly truthful mechanism in the single-task setting. If we know the realization y of Y, we can simply apply a proper scoring rule and pay Alice and Bob $PS(y,\hat{\mathbf{p}}_{x_A})$ and $PS(y,\hat{\mathbf{p}}_{x_B})$ respectively. Then according to the property of the proper scoring rule, Alice and Bob will honestly report their truthful forecast to maximize their expected payment. However, we do not know the realization of Y. In the information elicitation without verification setting where Alice and Bob are required to report their information, Miller et al. [23] propose the "peer prediction" idea, that is, pays Alice the accuracy of the forecast that predicts Bob's information conditioning Alice's information, i.e., $PS\big(\hat{x}_B,(\Pr[X_B=x_B|\hat{x}_A])_y\big)$ where $\hat{x}_A$ and $\hat{x}_B$ are Alice and Bob's reported information. We note the peer prediction mechanism in Miller et al. [23] is truthful. With a similar "peer prediction" idea, we propose a strictly truthful mechanism in forecast elicitation.

Common ground mechanism. Given the prior distribution over Y,

Report Alice and Bob are required to report $\mathbf{p}_{x_A},\mathbf{p}_{x_B}$. We denote their actual reports by $\hat{\mathbf{p}}_{x_A}$ and $\hat{\mathbf{p}}_{x_B}$.

Payment Both Alice and Bob are paid

$$\log\sum_y\frac{\hat{\mathbf{p}}_{x_A}(y)\hat{\mathbf{p}}_{x_B}(y)}{\Pr[Y=y]}.$$

Theorem 5.4. *With the conditional independence assumption (and when the prior is stable), given the prior distribution over the Y, the common ground mechanism is (strictly) truthful; moreover, when both Alice and Bob are honest, each of them's expected payment in the common ground mechanism is the Shannon mutual information between their private information $I(X_A; X_B) = MI^{KL}(X_A; X_B)$.*

The (strictly) truthful property of the common ground mechanism is proved by the fact that log scoring rule *LSR* is strictly proper. We defer the proof to the full version.

6 CONCLUSION AND DISCUSSION

We build a natural connection between mechanism design and machine learning by addressing two related problems: (1) co-training: learning to forecast ground truth using two conditionally independent sources, without access to labeled data; (2) forecast elicitation: eliciting high quality forecasts from the crowds without verification, by the same information theoretic approach.

For the co-training problem, as usual in the related literature, we reduce the problem to an optimization problem and do not investigate the computation complexity or the data requirements. To implement our f-mutual information gain framework in practice, we implicitly assume that for high dimensional X_A, X_B, there exists a trainable set of hypotheses (e.g. neural networks) that is sufficiently rich to contain the Bayesian posterior predictor but not everything to cause over-fitting. The most apparent empirical direction will be running experiments on real data by training two neural networks to test our algorithms. Interesting theoretic directions include the analysis of the Bayesian risk and the influence of the choice of the convex function f on the convergence rate.

For forecast elicitation, the most apparent direction will be performing real-world experiments. To apply our mechanisms, we do not need that every two agents' information is conditionally independent. In fact, for each agent, we only need to find a single reference agent for her such that the reference agent's information is conditionally independent of hers. Then we can run our mechanisms on the agent and her reference agent. In practice, we can pair the agents with some side information and make sure each pair of agents' information is conditionally independent.

Another interesting direction is to ensure fairness, in particular, that agents are not incentivized to coordinate on stereotypes. One solution, is suppressing information from some of the agents and using our framework. However, when this is not possible, the prior peer prediction work on cheap signals [12, 18] may be helpful in addressing this issue.

ACKNOWLEDGEMENT

We thank Clayton Scott for useful conversations.

REFERENCES

[1] Arpit Agarwal and Shivani Agarwal. 2015. On consistent surrogate risk minimization and property elicitation. In *Conference on Learning Theory*. 4–22.

[2] Syed Mumtaz Ali and Samuel D Silvey. 1966. A general class of coefficients of divergence of one distribution from another. *Journal of the Royal Statistical Society. Series B (Methodological)* (1966), 131–142.

[3] Suzanna Becker. 1996. Mutual information maximization: models of cortical self-organization. *Network: Computation in neural systems* 7, 1 (1996), 7–31.

[4] Anthony J Bell and Terrence J Sejnowski. 1995. An information-maximization approach to blind separation and blind deconvolution. *Neural computation* 7, 6 (1995), 1129–1159.

[5] Avrim Blum and Tom Mitchell. 1998. Combining labeled and unlabeled data with co-training. In *Proceedings of the eleventh annual conference on Computational learning theory*. ACM, 92–100.

[6] J-F Cardoso. 1997. Infomax and maximum likelihood for blind source separation. *IEEE Signal processing letters* 4, 4 (1997), 112–114.

[7] Michael Collins and Yoram Singer. 1999. Unsupervised models for named entity classification. In *1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*.

[8] Thomas M Cover and Joy A Thomas. 2006. Elements of information theory 2nd edition. (2006).

[9] Imre Csiszár, Paul C Shields, et al. 2004. Information theory and statistics: A tutorial. *Foundations and Trends® in Communications and Information Theory* 1, 4 (2004), 417–528.

[10] Anirban Dasgupta and Arpita Ghosh. 2013. Crowdsourced judgement elicitation with endogenous proficiency. In *Proceedings of the 22nd international conference on World Wide Web*. 319–330.

[11] Sanjoy Dasgupta, Michael L Littman, and David A McAllester. 2002. PAC generalization bounds for co-training. In *Advances in neural information processing systems*. 375–382.

[12] A. Gao, J. R. Wright, and K. Leyton-Brown. 2016. Incentivizing Evaluation via Limited Access to Ground Truth: Peer-Prediction Makes Things Worse. *ArXiv e-prints* (June 2016). arXiv:cs.GT/1606.07042

[13] Tilmann Gneiting and Adrian E Raftery. 2007. Strictly proper scoring rules, prediction, and estimation. *J. Amer. Statist. Assoc.* 102, 477 (2007), 359–378.

[14] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in neural information processing systems*. 2672–2680.

[15] Sham M Kakade and Dean P Foster. 2007. Multi-view regression via canonical correlation analysis. In *International Conference on Computational Learning Theory*. Springer, 82–96.

[16] Roni Khardon and Gabriel Wachman. 2007. Noise tolerant variants of the perceptron algorithm. *Journal of Machine Learning Research* 8, Feb (2007), 227–248.

[17] Y. Kong and G. Schoenebeck. 2016. An Information Theoretic Framework For Designing Information Elicitation Mechanisms That Reward Truth-telling. *ArXiv e-prints* (May 2016). arXiv:cs.GT/1605.01021

[18] Y. Kong and G. Schoenebeck. 2018. Eliciting Expertise without Verification. *ArXiv e-prints* (Feb. 2018). arXiv:cs.GT/1802.08312

[19] Yingming Li, Ming Yang, and Zhongfei Zhang. 2016. Multi-view representation learning: A survey from shallow methods to deep methods. *arXiv preprint arXiv:1610.01206* (2016).

[20] Yang Liu and Yiling Chen. 2017. Machine-Learning Aided Peer Prediction. In *Proceedings of the 2017 ACM Conference on Economics and Computation (EC '17)*. ACM, New York, NY, USA, 63–80. https://doi.org/10.1145/3033274.3085126

[21] Yang Liu and Yiling Chen. 2018. Surrogate Scoring Rules and a Dominant Truth Serum for Information Elicitation. *CoRR* abs/1802.09158 (2018). arXiv:1802.09158 http://arxiv.org/abs/1802.09158

[22] D. McAllester. 2018. Information Theoretic Co-Training. *ArXiv e-prints* (Feb. 2018). arXiv:cs.LG/1802.07572

[23] N. Miller, P. Resnick, and R. Zeckhauser. 2005. Eliciting informative feedback: The peer-prediction method. *Management Science* (2005), 1359–1373.

[24] Nagarajan Natarajan, Inderjit S Dhillon, Pradeep K Ravikumar, and Ambuj Tewari. 2013. Learning with noisy labels. In *Advances in neural information processing systems*. 1196–1204.

[25] XuanLong Nguyen, Martin J Wainwright, and Michael I Jordan. 2009. On surrogate loss functions and f-divergences. *The Annals of Statistics* (2009), 876–904.

[26] XuanLong Nguyen, Martin J Wainwright, and Michael I Jordan. 2010. Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Transactions on Information Theory* 56, 11 (2010), 5847–5861.

[27] Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. 2016. f-gan: Training generative neural samplers using variational divergence minimization. In *Advances in Neural Information Processing Systems*. 271–279.

[28] D. Prelec. 2004. A Bayesian Truth Serum for subjective data. *Science* 306, 5695 (2004), 462–466.

[29] Alexander J Ratner, Christopher M De Sa, Sen Wu, Daniel Selsam, and Christopher Ré. 2016. Data programming: Creating large training sets, quickly. In *Advances in Neural Information Processing Systems*. 3567–3575.

[30] Vikas C Raykar, Shipeng Yu, Linda H Zhao, Gerardo Hermosillo Valadez, Charles Florin, Luca Bogoni, and Linda Moy. 2010. Learning from crowds. *Journal of Machine Learning Research* 11, Apr (2010), 1297–1322.

[31] R Tyrrell Rockafellar et al. 1966. Extension of Fenchel'duality theorem for convex functions. *Duke mathematical journal* 33, 1 (1966), 81–89.

[32] Clayton Scott, Gilles Blanchard, and Gregory Handy. 2013. Classification with asymmetric label noise: Consistency and maximal denoising. In *Conference On Learning Theory*. 489–511.

[33] Victor Shnayder, Arpit Agarwal, Rafael Frongillo, and David C Parkes. 2016. Informed truthfulness in multi-task peer prediction. In *Proceedings of the 2016 ACM Conference on Economics and Computation*. ACM, 179–196.

[34] Sainbayar Sukhbaatar and Rob Fergus. 2014. Learning from noisy labels with deep neural networks. *arXiv preprint arXiv:1406.2080* 2, 3 (2014), 4.

[35] Robert L Winkler. 1969. Scoring rules and the evaluation of probability assessors. *J. Amer. Statist. Assoc.* 64, 327 (1969), 1073–1078.

[36] Jens Witkowski, Pavel Atanasov, Lyle H Ungar, and Andreas Krause. 2017. Proper Proxy Scoring Rules.. In *AAAI*. 743–749.

[37] Chang Xu, Dacheng Tao, and Chao Xu. 2013. A survey on multi-view learning. *arXiv preprint arXiv:1304.5634* (2013).

[38] Yuchen Zhang, Xi Chen, Denny Zhou, and Michael I Jordan. 2014. Spectral methods meet EM: A provably optimal algorithm for crowdsourcing. In *Advances in neural information processing systems*. 1260–1268.