Proactive Skill Posting in Referral Networks

Ashiqur R. KhudaBukhsh[1], Jaime G. Carbonell[1], and Peter J. Jansen[1]

Carnegie Mellon University
{akhudabu, jgc, pjj}@cs.cmu.edu

Abstract Distributed learning in expert referral networks is an emerging challenge in the intersection of Active Learning and Multi-Agent Reinforcement Learning, where experts—humans or automated agents—can either solve problems themselves or refer said problems to others with more appropriate expertise. Recent work demonstrated methods that can substantially improve the overall performance of a network and proposed a distributed referral-learning algorithm, DIEL (Distributed Interval Estimation Learning), for learning appropriate referral choices. This paper augments the learning setting with a proactive skill posting step where experts can report some of their top skills to their colleagues. We found that in this new learning setting with meaningful priors, a modified algorithm, proactive-DIEL, performed initially much better and reached its maximum performance sooner than DIEL on the same data set used previously. Empirical evaluations show that the learning algorithm is robust to random noise in an expert's estimation of her own expertise, and there is little advantage in misreporting skills when the rest of the experts report truthfully, i.e., the algorithm is near Bayesian-Nash incentive-compatible.

Keywords: Active learning; referral network; proactive skill posting

1 Introduction

Consider a network of experts with differing expertise, where any expert may receive a problem (aka a task or a query) and must decide whether to work on it or to refer the problem, and if so to which other expert. For instance, in a clinical network, a physician may diagnose and treat a patient or refer the patient to another physician whom she believes may have more appropriate knowledge, given the presenting symptoms. The referring physician may charge a referral fee and the receiving physician may charge a larger fee for diagnosing and treating the patient. Referral networks are common across other professions as well, such as members of large consultancy firms. If the experts are software agents, then the need for referral may be greater, given the likely narrower "expertise" typical of intelligent agents (including old-style expert systems). We can also envision a hybrid referral network comprising automated agents and possibly crowd-sourced human experts.

How does a network or how do individual experts in the network learn to refer effectively? Human referral networks are neither hardwired nor static. Potentially much larger networks of automated experts or hybrid networks with dynamic membership must likewise learn to refer with membership drift. One option is to maintain a global index and/or a "boss agent" telling all the others when to try and solve a problem or when to refer and to whom. In practice, however – in medicine, in academia, or in consulting companies – referrals occur to those one knows and trusts to do a good job. Hence a distributed learning setting, although it poses greater challenges to learning referrals, is a more realistic alternative.

To this end, a simple yet effective learning-to-refer method, dubbed DIEL (Distributed Interval Estimation Learning), has been proposed in (7). The referral model assumes an initial sparse topology of a static referral graph where each expert knows a handful of colleagues so that $E \sim O(V)$ (E and V denote the number of edges and vertexes in the network, respectively). Learning consists of each expert improving its estimates of the ability of colleagues to solve different classes of problems. On a wide range of simulations with different network structures and parameters chosen to represent possible real-life scenarios, DIEL consistently outperformed greedy (Distributed Mean-Tracking, DMT) and random baselines, and Q-learning variants. However, all such experiments assumed an uninformative prior on the expertise of colleagues which may not correspond to a real-world setting. Moreover, in real life, we often see that experts clearly mention which type of tasks they are particularly good at and also often forge links to their colleagues via social networks. In turn, their colleagues may re-estimate their beliefs of expertise levels based on actual performance.

In this work, we augment the original DIEL expertise-learning setting with a local-network advertisement of expertise-by-topic by each expert in the network. Our primary contribution is this augmented learning setting and a distributed referral-learning algorithm, proactive-DIEL, that takes advertised priors from other colleagues into account. On the same data set used in (7), with an accurate estimate of true skill and truthful reporting proactive-DIEL substantially outperformed DIEL in the initial phase of learning. Additionally, we found that proactive-DIEL is robust to limited Gaussian noise in an expert's estimation of her own skills. Also, our experimental evaluations reveal that proactive-DIEL displays empirical evidence of being near Bayesian-Nash incentive-compatible, i.e., when all the other experts are truthful, lying about one's own level of expertise has little or no advantage.

The rest of the paper is organized as follows. We first illustrate the basic referral mechanism through a small example. After a discussion of related work, we present the list of assumptions on the referral network and expertise. Next, we describe our distributed learning algorithms and experimental setup, after which we present and discuss the results from our experiments. Finally, we end with some general conclusions and an outlook on future work.

2 Referral Mechanism

We illustrate the referral mechanism and its effectiveness with the simple graph of Figure 1 (taken from (7)), which represents a five-expert network.

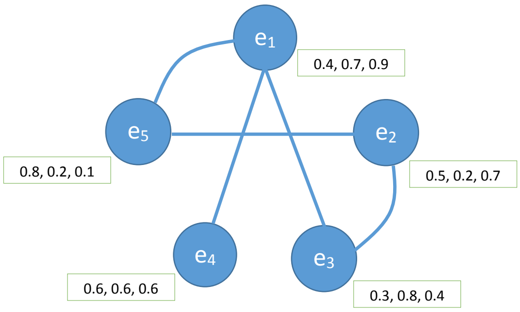


Figure 1. A referral network with five experts.

The nodes of the graph are the experts, and the edges indicate that the experts 'know' each other, that is, they can send or receive referrals and communicate results. In the domain, three different topics (subdomains) can be distinguished – call them t_1, t_2, and t_3 – and the figures in brackets indicate an expert's expertise in each of these.

In this referral network, with a query belonging to t_2, if there was no referral, the client may consult first e_2 and then possibly e_5, leading to a probability of getting the correct answer of $0.2 + (1 - 0.2) \times 0.2 = 0.36$. With referrals, an expert handles a problem she knows how to answer, and otherwise if she had knowledge of all the other experts' expertise she could ask e_2 who would refer to e_3 for the best skill in t_2, leading to a solution probability of $0.2 + (1 - 0.2) \times 0.8 = 0.84$.

The referral mechanism consists of the following steps.

1. A user issues an *initial query* to an *initial expert*, chosen from a prior distribution, e.g. uniform: equiprobably among all experts, or based on proactive-social-net advertisement priors.
2. The initial expert examines the instance. If she is able to solve it, she returns the solution or label to the user.
3. If she is not able to solve the problem, she issues a *referral query* to a *referred expert*, i.e. colleague who may be better able to solve the problem. *Learning-to-refer* means improving the estimate of who is most likely to solve the problem and/or, which expert should best be sampled in order to learn their level of expertise on the topic of the problem.
4. If the referred expert succeeds, she communicates the solution to the initial expert, who in turn, communicates it to the user.

3 Related Work

In terms of the learning setting and referral learning algorithm, our primary basis for this work is (7) which first proposed a novel learning setting in the context of Active learning where experts are connected through a network and can refer instances to one another. We augment this learning setting by allowing advertisement to colleagues, and modify the proposed algorithm DIEL both for improved performance and to encourage incentive compatibility. The referral learning algorithm proposed in (7) builds upon a chain of research on interval estimation learning, a reinforcement learning technique which strives to strike a balance between exploration and exploitation first proposed in (4; 5), which has been successfully used in the context of estimating accuracy of multiple labelers in (3). The referral framework draws inspiration from earlier work in referral chaining, first proposed in (6) and subsequently extended in(11; 12; 14; 13).

The primary focus of our work is presenting an extended learning setting by augmenting the existing referral framework with proactive skill posting and designing an appropriate distributed referral learning algorithm. However, in a real-world application, success will depend on preventing experts from misreporting their true skills as they advertise (e.g. to acquire more business). Among a large body of literature in truthful mechanism design (1; 2; 10; 9) we highlight a few key differences with the *budgeted multi-armed bandit mechanism* motivated by crowdsourcing platforms presented in (2). First, while *learning-to-refer* can be interpreted as a multi-armed bandit problem where each arm is a referral choice, our work deals with several such parallel multi-armed bandit problems in a distributed network setting. Also, in our setting each individual expert needs to learn appropriate referral choices based on topics and expertise estimates of colleagues on said topics. This is in contrast with homogeneous tasks, for example as considered in (2). Finally, proactive-DIEL deals with partially available priors since experts are not bidding for all the topics because of a restricted advertisement budget (a factor (2) did not need to consider because of homogeneous tasks).

For similar reasons, our work differs from past work studying the communication between experts and non-experts in crowdsourcing. In (8), domain experts break down a complex task into simpler micro-tasks and actively supervise the non-expert crowd. In our approach, there is no such simplifying assumption: it is highly unlikely that one expert Pareto dominates other experts in a professional network across all topics. Instead, referral is bi-directional, and the main focus is on learning appropriate referral choices in a distributed manner.

4 Referral Networks

We follow the same notation, general setting for referral mechanism, and initial set of assumptions as in (7), which we describe here in further detail. Section 4.4 presents the additional assumptions and mechanisms for proactive skill posting first introduced in this paper.

4.1 Notation

– A *referral network* can be represented by a graph (V, E) of size k in which each vertex v_i corresponds to an expert e_i and each bidirectional edge $\langle v_i, v_j \rangle$ indicates a *referral link*.

- We call the set of experts linked to an expert e_i via a referral link, the *subnetwork* of expert e_i.
- A *scenario* is a set of m queries $(q_1, \ldots, q_m)$ belonging to n topics $(t_1, \ldots, t_n)$, to be addressed by the k experts $(e_1, \ldots, e_k)$.
- As *expertise* for an expert-instance pair we simply take the probability that she can successfully solve the problem, i.e., $\mathit{Expertise}(e_i, q_j) = P(\mathit{solve}(e_i, q_j))$.

4.2 Initial Expert and Expertise assumptions

Topic-wise distributional assumption: We take the expertise distribution for a given topic t to be a mixture of two truncated Gaussians (with parameters $\lambda = \{w_i^t, \mu_i^t, \sigma_i^t\}\ i = 1, 2.$). One of them ($\mathcal{N}(\mu_2^t, \sigma_2^t)$) has a greater mean ($\mu_2^t > \mu_1^t$), smaller variance ($\sigma_2^t < \sigma_1^t$) and lower mixture weight ($w_2^t << w_1^t$). Intuitively, this represents the expertise of experts with specific training for the given topic, contrasted with the lower-level expertise of the layman population.

Instance-wise distributional assumption: We model the expertise of a given expert on instances under a topic by a truncated Gaussian distribution with small variance. i.e.,
$\mathit{Expertise}(e_i, q_j) \sim \mathcal{N}(\mu_{topic_p, e_i}, \sigma_{topic_p, e_i})$,
$\forall q_j \in topic_p, \forall p, i : \sigma_{topic_p, e_i} \leq 0.2$.

We further assume that an expert can accurately identify the topic of a query, that expertise does not change over time (but see future work in the conclusion), and that experts have no capacity constraints.

4.3 Network assumption

The probability that a referral link exists between expert e_i and e_j is a function of how similar the two experts are, which we modeled as
$P(\mathit{ReferralLink}(v_i, v_j)) = \tau + c\ \mathit{Sim}(e_i, e_j)$. As a similarity metric we used *cosine similarity of topic-means*. The parameter τ captures any extraneous reason two experts can be connected, e.g., same geolocation, common acquaintances, etc.

4.4 Proactive skill posting

An advertising unit is a tuple $\langle e_i, e_j, t_k, \mu_{t_k} \rangle$, where e_i is the *target expert*, e_j is the *advertising expert*, t_k is the topic and μ_{t_k} is e_j's (advertised) topical expertise.

We initially assume that experts can estimate their expertise (topic-means) accurately, and report truthfully. We assume each expert is allocated a budget of B advertising units, where B is twice the size of that expert's subnetwork. The advertising budget addresses the limited time that experts have to socialize with different colleagues and get to know each other's experience.

The advertising expert e_j reports to each target expert e_i in her subnetwork the two tuples $\langle e_i, e_j, t_{best}, \mu_{t_{best}} \rangle$ and $\langle e_i, e_j, t_{secondBest}, \mu_{t_{secondBest}} \rangle$, i.e., the top two topics in terms of the advertising expert's topic means. This is a one-time advertisement and happens right at the beginning of the simulation.

5 Distributed Referral Learning

In this section, we first present interval estimation learning (3) which is the primary building block of DIEL. Next, we outline the steps for DIEL and present the key ways in which proactive-DIEL differs.

5.1 Interval Estimation Learning

```
Input: A set of k experts e_1, e_2, ..., e_k. A set of n topics topic_1, topic_2, ...,
       topic_n. A k × k referral network.
Initialize rewards.
for iter ← 1 to maxIter do
    Assign instance q to an initial expert e randomly
    if e fails to solve q then
        topic ← getTopic(q)
        expectedReward ← 0
        bestExpert ← 0
        for each expert e' in the subnetwork of e do
            if expR_h(e', topic) ≥ expectedReward then
                bestExpert ← e'
                expectedReward ← expR_h(e', topic)
            end
        end
    end
    referredExpert ← bestExpert
    if referredExpert solves q then
        update(reward(e,topic,referredExpert),1)
    else
        update(reward(e,topic,referredExpert),0)
    end
end
```

Algorithm 1: Distributed Referral Learning, Q = 2

Action selection using Interval Estimation Learning (IEL) first estimates for each action a the upper confidence bound for the mean reward by

$$UI(a) = m(a) + t_{\frac{\alpha}{2}}^{(n(a)-1)} \frac{s(a)}{\sqrt{n(a)}} \tag{1}$$

where m(a) is the mean observed reward for a, $s(a)$ is the sample standard deviation of the reward, $n(a)$ is the number of observed samples from a, and $t_{\frac{\alpha}{2}}^{(n(a)-1)}$ is is the critical value for the Student's t-distribution ($n(a)-1$ degrees of freedom, $\frac{\alpha}{2}$ confidence level) (in our case, the action is the selection of a referred expert among possible choices in the subnetwork). Next, IEL selects the action with the highest upper confidence bound.

The intuition is that high mean selects for best performance, and high variance selects for unexplored expert capability on topic, thus optimizing for amortized performance, as variance decreases over time, and best mean is selected reliably among the top candidates. The parameter α weights exploration more when small and exploitation more when large. A partial parameter sweep confirmed that a value of $\alpha = 0.05$, settled on in (3), also worked well in (7).

5.2 Distributed Referral Learning

Algorithm 1 outlines the steps for expertise estimation in a distributed setting with single referral (a per-task query budget $Q = 2$). The function $expR_h(e', topic)$ estimates e''s topical expertise using heuristic h. DIEL (Distributed Interval Estimation Learning), and DMT (Distributed Mean-Tracking) differ in this heuristic, DIEL estimating reward by equation (1) and DMT by using the sample-mean.

One major challenge in the distributed setting is that there is no global visibility of rewards, i.e., $reward(e_i, topic_p, e_j)$ is only visible to expert e_i. When the task is solved successfully, *update* assigns an additional reward of 1 to the referred expert (experts keep track of success and failure of experts they refer to by means of a sequence of 0s and 1s; here, we just mean that a “1” is appended to this sequence in the case of success).

Proactive-DIEL differs from DIEL in the following three key ways.

1. Emphasis on exploitation: The Student’s t-distribution parameter has a large value for smaller n’s and drops down towards 1 as n increases, thereby boosting exploration in early on in the learning. This is less important in our current setting where we start with a partially informative prior, hence we dropped this parameter in equation (1), leading to

$$UI(a) = m(a) + \frac{s(a)}{\sqrt{n(a)}} \tag{2}$$

In fact, since we found that even the original DIEL performed better on the original data in (7) with this revised formula, we used it for subsequent performance comparisons in this paper.

2. Initialization: Rather than initializing DIEL sets $reward(e_i, t_k, e_j)$ for each i, j and k with a pair $(0, 1)$ in order to initialize mean and variance, as in DIEL, proactive-DIEL initializes $reward(e_i, t_k, e_j)$ for each advertisement unit $\langle e_i, e_j, t_k, \mu_{t_k} \rangle$ with two rewards of μ_{t_k}.

To initialize topics for which no advertisement units are available (recall that the budget was assumed to suffice for advertising an expert’s top two skills only) we assumed it was safe and informative to initialize the rewards as if the expert’s skill was the same as on her second best topic, that is, with two rewards of $\mu_{t_{secondBest}}$, effectively being an upper bound on the actual value.

3. Reward update function:

When a referred expert e_j succeeds in solving a task on topic t_k, *update* in proactive-DIEL’s *update* function, like DIEL’s, assigns an additional reward of 1 to $reward(e_i, t_k, e_j)$.

When e_j fails, however, instead of always appending a 0 to $reward(e_i, t_k, e_j)$, proactive-DIEL, in the presence of an advertisement unit $\langle e_i, e_j, t_k, \mu_{t_k}\rangle$, appends a (negative) penalty P with probability μ_{t_k}. This way, over-reporting of skill leads to more frequent incurrence of the penalty.

In the absence of an advertisement unit, a penalty P is still appended, but with a probability equal to the sample mean of e_j observed by e_i on topic t_k. In our experiments, we set P to -0.35.

6 Experimental Setup

Parameter	Description	Distribution
τ	$P(ReferralLink(v_i, v_j\ ))$	Uniform(0.01, 0.1)
c	$= \tau + c\ Sim(e_i, e_j)$.	Uniform(0.1,0.2)
μ_1	Truncated mixture of two	Uniform(0,b)
μ_2	Gaussians for topics	Uniform(b,1)
		$b \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$
σ_1		Uniform(0.2,0.4)
σ_2		Uniform(0.05,0.15)
w_2		$\mathcal{N}(0.03, 0.01)$, $w_2 \geq 0$

Table 1. Parameters for synthetic data set.

Since the performance of a referral network is sensitive to the topology of the network and expertise of experts on individual topics, we evaluated the performance of proactive-DIEL on the same wide range of scenarios considered in (7) by varying the parameters in the network. Table 1 lists the distributions from which the parameters are sampled. The data set consisted of 1000 scenarios, each with 100 experts, 10 topics and a referral network.

Following (7), we also report upper-bound performance of a network where every expert has access to an oracle that knows the true topic-mean (i.e., $mean(Expertise(e_i, q) : q \in topic_p)\ \forall i, p)$ of every expert-topic pair. Our measure of performance is the overall task accuracy of our multi-expert system.

7 Results

In this section we present our three main results: (1) Access to (noisy) priors on their colleagues' expertise improves an expert's performance in both the DIEL and DMT algorithms, (2) in the augmented setting, even partial and possibly inaccurate information on the priors helps, and (3) misreporting skills confers little or no advantage when the other experts report truthfully.

7.1 DIEL and DMT with informative prior

We first show that informative priors on the means can be easily incorporated into DIEL and DMT and incorporating the priors is beneficial for both DIEL

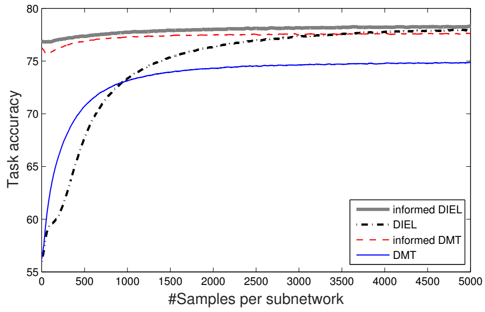


Figure 2. DIEL and DMT with informative prior.

and DMT. Suppose every expert has access to an oracle than can estimate the true topic-mean of every other expert-topic pair within an error bound, i.e. $|\mu_{e_i,t_k} - \hat{\mu}_{e_i,t_k}| \leq \delta$. Unlike (7), instead of a 0 and a 1, all rewards $reward(e_i, t_k, e_j)$ are initialized with two rewards of $\hat{\mu}_{e_i,t_k}$. In figure 2, we see that even when δ is as high as 0.2, the performance of both DIEL and DMT substantially improved with uninformed DIEL still outperforming informed DMT at the later stage given enough samples.

7.2 Performance of proactive-DIEL

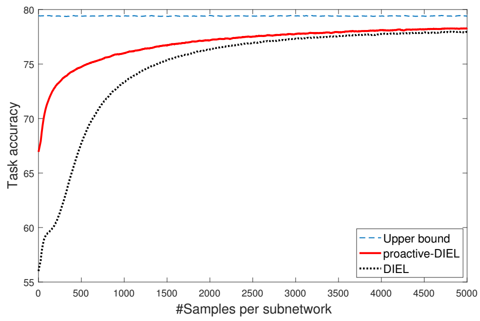
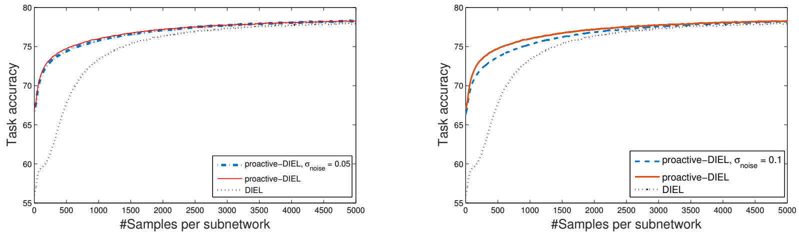


Figure 3. Performance comparison of DIEL and proactive-DIEL.

Next, we analyze the performance of DIEL with proactive skill posting. Figure 3 shows that even with a limited budget of $2 \times |subnetwork|$ (i.e., two advertisements per expert in the subnetwork), proactive-DIEL requires very few samples to reach a reasonably high overall network performance.

Even when experts post their skills truthfully, their self-estimates may be off. Here, we relax the accurate skill estimation assumption. Specifically, we considered $\hat{\mu} = \mu + \mathcal{N}(0, \sigma_{noise})$, where $\hat{\mu}$ is an expert's own estimate of her true topic-mean μ, and σ_{noise} is a small constant (we tried two different values for σ_{noise}, 0.05 and 0.1). Figure 4 compares the performance of proactive-DIEL with



(a) $\sigma_{noise} = 0.05$ (b) $\sigma_{noise} = 0.10$

Figure 4. proactive-DIEL with noisy skill estimation.

noisy estimates with the initial noise-free version and DIEL. Figure 4(a) shows that in the early stage, with a small amount of noise in estimation, eventually the same performance as the noise-free version is achieved while remaining strictly superior to DIEL throughout the entire course of simulation. With a larger value of noise, Figure 4(b) shows that proactive-DIEL is slightly worse than the noise-free version, but nonetheless, outperforms DIEL.

7.3 Bayesian-Nash Incentive-Compatibility

Our results show that proactive-DIEL is immune to a small amount of Gaussian noise in estimating topical expertise. Another important but somewhat orthogonal goal is to prevent deliberate misreporting, e.g. experts trying to get more business by overstating their skills. We treat the number of referrals received as a proxy for payment. Intuitively, proactive-DIEL's design discourages misreporting in the following two ways: For an advertised skill, if over-reported, a failed task would receive a higher skill-estimation penalty that it would otherwise. When a skill is under-reported, the expert will be selected less often, and hence it is naturally self-balancing.

$\mu_{t_{best}}$	$\mu_{t_{secondBest}}$	Factor
Truthful	Overbid	0.99
Overbid	Truthful	**1.00**
Overbid	Overbid	0.97
Truthful	Underbid	**1.04**
Underbid	Truthful	**1.09**
Underbid	Underbid	**1.22**
Underbid	Overbid	**1.11**
Overbid	Underbid	**1.04**

Table 2. Empirical analysis on Bayesian-Nash incentive-compatibility. Strategies where being truthful is no worse than being dishonest are highlighted in bold.

Our empirical evaluation shows that in the steady-state, proactive-DIEL is fairly resilient to strategic lying, and is almost Bayesian-Nash incentive-compatible, i.e., there is little or no advantage for an individual expert in

misreporting topical expertise when all other experts in the network report truthfully. We analyze the effect of misreporting the following way. First, we considered different combinations an expert can use while reporting their best and second-best skill (listed in Table 2). For a given strategy and scenario $scenario_i$, we first fix one expert, say e_l^i. Let $truthfulReferrals(e_l^i)$ denote the number of referrals received by e_l^i beyond a steady-state threshold (i.e., a referral gets counted if the initial expert has referred 1000 or more instances to her subnetwork) when e_l^i and all other experts report truthfully. Similarly, let $strategicReferrals(e_l^i)$ denote the number of referrals received by e_l^i beyond a steady-state threshold when e_l^i misreports while everyone else advertises truthfully. We compute the following factor:
$\frac{\sum_{i=1}^{1000} truthfulReferrals(e_l^i)}{\sum_{i=1}^{1000} strategicReferrals(e_l^i)}$ i.e., the ratio of the total number of $truthfulReferrals(e_l^i)$ to the total number of $strategicReferrals(e_l^i)$ across the entire data set. A value greater than 1 implies truthful reporting fetched more referrals than strategic lying. Table 2 shows that beyond the steady-state threshold, strategic misreporting is hardly beneficial and in most of the cases honest reporting of skills fetched more referrals.

8 Conclusion

In this work, we extended the referral-learning as proposed in (7) with a skill posting or advertising step, and revised the DIEL algorithm to (1) take advantage of informative priors and (2) include a mechanism to discourage cheating, such as skill over-reporting to get additional "business" by individual experts in a referral network

Our new algorithm, proactive-DIEL, outperformed its predecessor convincingly, while nearly reaching empirical Bayesian Nash incentive compatibility.

We intend to extend these results in the following ways in the future.

- **Strategyproofness**: While misreporting was shown to be of little or no benefit when all other experts report truthfully, a stronger degree of incentive compatibility, strategyproofness, would require this to be the case no matter what other experts do. One future goal is to investigate what modifications to proactive-DIEL, or which conditions would ensure this.
- **Dynamic networks**: Our results showed an improved performance through informed priors in a static setting. Future research could compare the resilience of proactive-DIEL and DIEL in a dynamic setting, for instance when new experts join the network, or existing ones drop off.
- **Expertise drift**: In this work, we assumed expertise does not change with time. But it is conceivable that experts, for instance, improve with practice. Modifying proactive-DIEL to deal with time-varying expertise will be particularly challenging and may require designing an adaptive version of the reward mechanism.

Bibliography

[1] Babaioff, M., Sharma, Y., Slivkins, A.: Characterizing truthful multi-armed bandit mechanisms. In: Proceedings of the 10th ACM conference on Electronic commerce. pp. 79–88. ACM (2009)

[2] Biswas, A., Jain, S., Mandal, D., Narahari, Y.: A truthful budget feasible multi-armed bandit mechanism for crowdsourcing time critical tasks. In: Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems. pp. 1101–1109. International Foundation for Autonomous Agents and Multiagent Systems (2015)

[3] Donmez, P., Carbonell, J.G., Schneider, J.: Efficiently learning the accuracy of labeling sources for selective sampling. Proc. of KDD 2009 p. 259 (2009)

[4] Kaelbling, L.P.: Learning in embedded systems. MIT press (1993)

[5] Kaelbling, L.P., Littman, M.L., Moore, A.P.: Reinforcement Learning: A Survey. Journal of Artificial Intelligence Research 4, 237–285 (1996)

[6] Kautz, H., Selman, B., Milewski, A.: Agent amplified communication pp. 3–9 (1996)

[7] KhudaBukhsh, A.R., Jansen, P.J., Carbonell, J.G.: Distributed learning in expert referral networks. In: European Conference on Artificial Intelligence (ECAI), 2016. pp. 1620–1621 (2016)

[8] Nallapati, R., Peerreddy, S., Singhal, P.: Skierarchy: Extending the power of crowdsourcing using a hierarchy of domain experts, crowd and machine learning. Tech. rep., DTIC Document (2012)

[9] Tran-Thanh, L., Chapman, A., Rogers, A., Jennings, N.R.: Knapsack based optimal policies for budget-limited multi-armed bandits. arXiv preprint arXiv:1204.1909 (2012)

[10] Tran-Thanh, L., Stein, S., Rogers, A., Jennings, N.R.: Efficient crowdsourcing of unknown experts using multi-armed bandits. In: European Conference on Artificial Intelligence. pp. 768–773 (2012)

[11] Yolum, P., Singh, M.P.: Dynamic communities in referral networks. Web Intelligence and Agent Systems 1(2), 105–116 (2003)

[12] Yu, B.: Emergence and evolution of agent-based referral networks. Ph.D. thesis, North Carolina State University (2002)

[13] Yu, B., Singh, M.P.: Searching social networks. Proc. of AAMAS '03 (2003)

[14] Yu, B., Venkatraman, M., Singh, M.P.: An adaptive social network for information access: Theoretical and experimental results. Applied Artificial Intelligence 17, 21–38 (2003)