Toward Reciprocity-aware Distributed Learning in Referral Networks

Ashiqur R. KhudaBukhsh[1] ✉ and Jaime G. Carbonell[1]

Carnegie Mellon University, Pittsburgh, PA 15213, USA
{akhudabu, jgc}@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—have varying skills across different topics and can redirect difficult problem instances to connected colleagues with more appropriate expertise. The *learning-to-refer* challenge involves estimating colleagues' topic-conditioned skills for appropriate referrals. Prior research has investigated different reinforcement learning algorithms both with uninformative priors and partially available (potentially noisy) priors. However, most human experts expect mutually-rewarding referrals, with return referrals on their expertise areas so that both (or all) parties benefit from networking, rather than one-sided referral flow. This paper analyzes the extent of referral reciprocity imbalance present in high-performance referral-learning algorithms, specifically multi-armed bandit (MAB) methods belonging to two broad categories – frequentist and Bayesian – and demonstrate that both algorithms suffer considerably from reciprocity imbalance. The paper proposes modifications to enable distributed learning methods to better balance referral reciprocity and thus make referral networks win-win for all parties. Extensive empirical evaluations demonstrate substantial improvement in mitigating reciprocity imbalance, while maintaining reasonably high overall solution performance.

Keywords: Referral networks · Reciprocity awareness · Active Learning.

1 Introduction

A referral network consists of multiple agents, human or autonomous, who learn to estimate the expertise of other known agents in order to optimize referral decisions when they are unable to solve a problem instance. *Learning-to-refer* in multi-agent referral networks has witnessed recent progress on several fronts, including distributed reinforcement learning algorithms [16], coping with some experts quitting the network and others joining [15], addressing expertise drift [12], e.g., as some experts hone their primary skills over time or others atrophy when disused. Other practical issues addressed include capacity constraints on how many problems an agent can address per unit of time [15]. These lines of work, however, implicitly assume altruistic agents, intent on solving problems collectively, rather than maximizing individual gain, where gain is proportional to business volume, i.e., incoming clients and referrals.

An extension beyond implicit altruism is the advent of resource-bounded proactive skill advertisement [11, 13, 14] among agents, where each agent attempts to maximize gain by attracting the largest number of referrals, assuming incoming referrals for problems an agent can solve result in economic gain. This extension required creating incentive-compatible mechanisms to induce agents to accurately report their skill levels (vs strategic lying), so that local economic gain would align with overall network problem-solving accuracy.

Another extension beyond implicit altruism requires addressing referral reciprocity among agents. Consider a network of physicians who know each other where A and B are dermatologists with different skill levels and C and D are neurologists, also with different skill levels. C might refer all patients with dermatological conditions to A if she believes A is the more skilled dermatologist. If B refers patients with neurological issues to C and D, and after a while notices that C never returns any referrals, a natural reaction would be "Why should I refer anyone to C if she never returns the business?" and henceforth B sends her neurological referrals only to D, even if she may believe D is not the best or most appropriate neurologist. If the reader would prefer to think that physicians act only in the patients' best interests, substitute dermatologists and neurologists with liability and tort lawyers, or with used car salesmen specializing in different auto brands, or with automated agents programmed to optimize their economic benefit. The key issue is reciprocity of referrals, or rather reciprocity imbalance, where an agent, repeatedly slighted by a peer via highly imbalanced referrals, changes her behavior in a manner that may not lead to optimal network referral behavior.

Reciprocity as a means to improve overall multi-agent cooperation has been studied in biological settings [26], economic settings [6], and AI-based multi-agent settings [9, 18], but not in the context of referral networks. This paper focuses on reciprocity imbalance (*RI*) which we define as the divergence from absolute reciprocity in the [0,1] interval. Absolute reciprocity means the referral flow between two agents is identical in both directions, in which case *RI* $\rightarrow 0$. Total lack of reciprocity means that either the first agent receives many referrals from the second agent but never reciprocates or vice versa (RI $\rightarrow 1$). The reasons for the arrows is that we measure empirical reciprocity imbalance, and with few data points we regularize it to be close to 0, and adjust upwards if warranted by new observations. Hence, the actual challenge we address in this paper is *learning-to-refer* in a multi-expert distributed setting taking reciprocity of referrals into account. In a broader context, our work is an example of a distributed AI application where a global goal (in our case, network-level task accuracy) is met via self-interested agents locally maximizing their self-interest.

Our work is different from extensive literature on *trust* and *reputation* [22, 23, 25] on the following key aspects. First, reputation is trust in aggregate, in contrast for reported work all rewards and referrals are fine-grained, how one agent models another agents behavior; there is no explicit communication among agents on reciprocity, nor any other requirements for global visibility. Second, unlike trust, reciprocity depends on mutual interaction (concerning referrals, estimated expertise levels and skill complementarities in each agents subnetwork etc.); an expert can be highly reciprocating to some while being completely non-reciprocating to others.

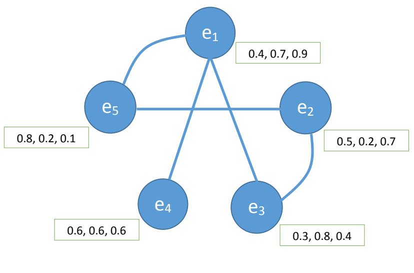
Key contributions: First, our extensive analysis on two high-performance referral learning algorithms that include the current state-of-the-art (a frequentist MAB algorithm `DIEL` [15]), and `Thompson Sampling` [20] (a well-known Bayesian MAB algorithm) reveals that both algorithms suffer from serious reciprocity imbalance and hence may not be well-suited for practical settings. Second, we propose a simple technique melding dual objectives that allows continual estimation of expertise of the colleagues taking both historical performance and reciprocity into account achieving considerable reduction in reciprocity imbalance at a small cost of overall performance. As baseline, we compared against an algorithmic setting where after observing a certain number of mutual referrals, an expert severs a connection with a colleague if the expert is unhappy with the reciprocal behavior of the colleague and forges a new link with another expert in the network. Our results indicate that such abrupt change in behavior is sub-optimal and achieves less reciprocity and a worse referral-learning performance as compared to our proposed solution. Third, we show that when all experts are reciprocity-aware, strategic deviation to altruism or greed fetches lesser referrals in expectation. Finally, we show that even when we start with the constrained referral-learning algorithm but at a later stage, if we switch to the unconstrained version, the algorithms are able to recover its performance and match the corresponding unconstrained version.

2 Background

2.1 Motivation

We illustrate the effectiveness of appropriate referrals with a small simplified example of a referral network with five experts shown in Figure 1 (this example is taken from [16]). The nodes of the graph represent the experts, and the edges indicate a potential referral link, i.e., 5 the experts 'know' each other and can send or receive referrals and communicate results. Consider three different topics – call them t_1, t_2, and t_3 – and the figures in brackets indicate an expert's topical expertise (probability of solving a given task) in each of these.

Fig. 1. A referral network with five experts.



In the example, with a query belonging to t_2, without any referral, the client may consult first v_2 and then possibly e_5, leading to a solution probability 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 connected colleagues' expertise, v_2 could 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 true topic-conditioned skills of the experts in the network are initially unknown and the *learning-to-refer* challenge is to estimate topical skills of the colleagues in a distributed setting with each expert independently estimating colleagues' topical expertise.

2.2 Preliminaries and Notation

Referral network: Represented by a graph (V, E) of size k in which each vertex v_i $(1 \leq i \leq k)$ corresponds to an expert and each bidirectional edge $\langle v_i, v_j \rangle$ indicates a *referral link* which implies v_i and v_j can co-refer problem instances.
Subnetwork: of an expert v_i: The set of experts linked to an expert v_i by a referral link.
Referral scenario: Set of m instances $(q_1, \ldots, q_m)$ belonging to n topics $(t_1, \ldots, t_n)$ addressed by the k experts $(v_1, \ldots, v_k)$ connected through a referral network (V, E).
Expertise: Expertise of an expert/instance pair $\langle v_i, q_l \rangle$ is the probability with which v_i can solve q_l.
Referral mechanism: For a query budget $Q = 2$, and a given instance, q_l, this consists of the following steps.

1. A user issues an *initial query* to a randomly chosen *initial expert* v_i.
2. The initial expert v_i examines q_l and solves it if possible. This depends on the *expertise* of v_i wrt. q_l.
3. If not, a *referral query* is issued by v_i to a *referred expert* v_j within her subnetwork, with a remaining query budget of $Q - 2$. *Learning-to-refer* involves improving the estimate of who is most likely to solve the problem.
4. If the referred expert succeeds, she sends the solution to the initial expert, who sends it to the user.

The first two steps are identical to Active Learning [19]; step 3 and 4 are the extension to the Active Learning setting. Understandably, with a higher per-instance query budget, the referred expert can re-refer instances to other experts as long as the budget permits. Following [15], in addition to single-hop referral ($Q = 2$), we also considered bounded multi-hop referrals with $Q = 3$ (two-hop) and $Q = 4$ (three-hop). Further details regarding expertise, network parameters, and simulation details can be found in [13, 15, 16].

2.3 Assumptions

We follow the same set of *assumptions* made in [15, 16]. Some of the important assumptions are: the network connectivity depends on (cosine) similarity between the topical expertise, and the distribution of topical-expertise across experts can be characterized by a mixture of Gaussian distributions; for any given instance, we assume that any of the k experts is equally likely to be the initial expert receiving the problem (query) externally. The network connectivity assumption is guided by the observation that experts with similar expertise are more likely to know each other. For topical-expertise distribution, a mixture of two Gaussians is considered. Gaussian distributions are widely used to model real-valued random variables (e.g., height, weight, expertise) in natural and social sciences. A mixture of two Gaussians was used to represent the expertise of experts with specific training for the given topic (higher mean, lower variance), contrasted with the lower-level expertise (lower mean, higher variance) of the layman population.

From the point of view of a single expert, for a given topic, learning referral policy maps to the classic MAB setting where each arm corresponds to a referral choice, and similar to the unknown reward distributions of the arms, the expertise of the colleagues is not initially known. In order to learn an effective referral strategy, depending on the outcome of a referred task, the initial expert assigns a reward to the referred colleague. All our rewards are

- **bounded**: In all our experiments, we considered binary rewards, with a failed and successful tasks receiving a reward of 0 and 1, respectively.
- **i.i.d**: The reward for a given expert on a specific instance belonging to a topic is independent of any reward observed from any other experts and any reward or sequence of rewards belonging to that topic or any other topic by the same expert.
- **locally assigned and locally visible**: $\textit{reward}(v_i, t, v_j)$, a function of initial expert v_i, referred expert v_j and topic t, is assigned by v_i and visible to v_i only.

2.4 Distributed Referral Learning

In a distributed setting, each expert maintains an action selection thread for each topic in parallel. In order to describe an action selection thread, we first name the topic T and expert v. Let $q_1, \ldots, q_N$ be the first N referred queries belonging to topic T issued by expert v to any of her K colleagues denoted by $v_1, \ldots, v_K$. For each colleague v_i, v maintains a reward vector $\mathbf{r}_{i,n_i}$ where $\mathbf{r}_{i,n_i} = (r_{i,1}, \ldots, r_{i,n_i})$, i.e., the sequence of rewards observed from expert v_i on issued n_i referred queries. Understandably, $N = \sum_{i=1}^{K} n_i$. Let $m(v_i)$ and $s(v_i)$ denote the sample mean and sample standard deviation of these reward vectors. Additional to the reward vectors, for each expert v_i, v maintains S_{v_i} and F_{v_i} where S_{v_i} denotes the number of observed successes (reward = 1) and F_{v_i} denotes the number of observed failures (reward = 0).

```
Algorithm 1: DIEL($v, T$)
  Initialization: $\forall i, n_i \leftarrow 2$, $\mathbf{r}_{i,n_i} \leftarrow (0, 1)$
  Loop: Select expert $v_i$ who maximizes
      $\textit{score}(v_i) = m(v_i) + \frac{s(v_i)}{\sqrt{n_i}}$
      Observe reward $\textit{reward}$
      Update $\mathbf{r}_{i,n_i}$ with $\textit{reward}$, $n_i \leftarrow n_i + 1$
```

We next focus on two well-established referral-learning algorithms that have extensive use in other reinforcement learning and MAB contexts. These algorithms are *reciprocity-agnostic*, i.e., they do not consider reciprocity while making any referral decision.

DIEL: Distributed Interval Estimation Learning (DIEL) is the known state-of-the-art referral learning algorithm [15]. First proposed in [10], Interval Estimation Learning (IEL) has been extensively used in stochastic optimization [7] and action selection problems [4, 24]. As described in Algorithm 1, at each step, DIEL selects the expert v_i with highest $m(v_i) + \frac{s(v_i)}{\sqrt{n_i}}$ (recall that, $m(v_i)$ and $s(v_i)$ denote the sample mean and sample standard deviation of the reward vector of expert v_i, respectively) . Every expert is initialized with two rewards of 0 and 1, allowing us to initialize the mean and variance.

DIEL addresses the classic *exploration-exploitation* trade-off [3] present in MAB algorithm design in the following way. A large variance implies greater uncertainty, indicating that the expert has not been sampled with sufficient frequency to obtain reliable skill estimates. Selecting such an expert is an *exploration step* which will increase the confidence of v in her estimate. Also, such steps have the potential of identifying a highly skilled expert, whose earlier skill estimate may have been too low. Selecting an expert with a high $m(v_i)$ amounts to *exploitation*. Initially, choices made by v tend to be explorative since the intervals are large due to the uncertainty of the reward estimates. With an increased number of samples, the intervals shrink and the referrals become more exploitative.

```
Algorithm 2: TS($v, T$)
  Initialization: $\forall i, S_{v_i} \leftarrow 0, F_{v_i} \leftarrow 0$
  Loop: Select expert $v_i$ who maximizes
      $score(v_i) = \theta_i$
      Observe reward $reward$
      $S_{v_i} \leftarrow S_{v_i} + reward$
      $F_{v_i} \leftarrow F_{v_i} + 1 - reward$
```

Thompson Sampling (TS): First proposed in the 1930's [20], finite-time regret bound of Thompson Sampling (TS) remained unsolved for decades [1] until recent results on its competitiveness with algorithms that exhibit provable regret bounds renewed interest [5, 8]. As described in Algorithm 2, at each step, for each expert v_i, TS first samples θ_i from $Beta(S_{v_i} + 1, F_{v_i} + 1)$ (recall that, S_{v_i} denotes the number of observed successes and F_{v_i} denotes the number of observed failures of expert v_i, respectively). Next, TS selects the action with highest θ_i. When the number of observations is 0, θ_i is sampled from $Beta(1, 1)$, which is $U(0, 1)$ which makes all colleagues equally likely to receive referral. As the number of observations increases, the distribution for a given expert becomes more and more centered around the empirical mean favoring experts with better historical performance.

3 Incorporating Reciprocity-awareness

We first introduce a quantitative measure of reciprocity imbalance in referrals starting with the definitions required to formalize the measure.

Interaction: At any given point, *interaction* of a *referral link* $\langle v_i, v_j \rangle$, denoted as *interaction*$(\langle v_i, v_j \rangle)$, is measured as *interaction*$(\langle v_i, v_j \rangle) = R(v_j \to v_i) + R(v_i \to v_j)$, where $R(v_i \to v_j)$ denotes the total number of referrals (across all topics) v_j has so far received from v_i. Since *interaction*$(\langle v_i, v_j \rangle)$ is used as a denominator in *referral share* (defined next), in order to avoid any divide-by-zero boundary condition, $\forall i, j$, $R(v_i \to v_j)$ is initialized to 1, effectively initializing *interaction*$(\langle v_i, v_j \rangle)$ $\forall i, j$ to 2.

Referral share: At any given point, the *referral share* of an expert v_i in a *referral link*, $\langle v_i, v_j \rangle$, denoted as *refShare*$(v_i, \langle v_i, v_j \rangle)$, is measured as *refShare*$(v_i, \langle v_i, v_j \rangle) = \frac{R(v_j \to v_i)}{interaction(\langle v_i, v_j \rangle)}$. In a reciprocal setting, for every expert in a *referral link*, we would like the *referral share* to be close to $\frac{1}{2}$.

Reciprocity imbalance of a referral link: For a given *referral link*, $\langle v_i, v_j \rangle$, the reciprocity imbalance, denoted as $RI(\langle v_i, v_j \rangle)$, is measured as $RI(\langle v_i, v_j \rangle) = |\frac{1}{2} - refShare(v_i, \langle v_i, v_j \rangle)| + |\frac{1}{2} - refShare(v_j, \langle v_i, v_j \rangle)|$.

For every *referral link*, $RI(\langle v_i, v_j\rangle)$ is initialized to zero. For any *referral link*, reciprocity imbalance is bounded within the range [0, 1] with 0 being a case of perfect reciprocity and 1 being the extreme case where one expert in a *referral link* does not receive any referrals from her colleague. Say, v_1 and v_2 have referred 100 instances between each other of which v_2 received 80 instances, the reciprocity of the link will be $|\frac{1}{2} - \frac{80}{100}| + |\frac{1}{2} - \frac{20}{100}|$ = $|0.5 - 0.8| + |0.5 - 0.2|$ = 0.6.

Reciprocity imbalance of a referral scenario: The reciprocity imbalance of a *referral scenario* is the average reciprocity imbalance present in its *referral links*.

Reciprocity-aware algorithm: We are now ready to define our reciprocity-aware algorithms. For any action selection algorithm $\mathcal{A}$, the corresponding reciprocity-aware (denoted as $\mathcal{A}_{RA}$) variant will only differ in the following way: The reciprocity-aware score of an expert colleague v_i of v for a given topic T, denoted as *RAscore*$^{v,T}_{\mathcal{A}}(v_i)$, is a function of its actual algorithmic score and referral share.

RAscore$^{v,T}_{\mathcal{A}}(v_i)$ = *score*$^{v,T}_{\mathcal{A}}(v_i)$ + *refScore*$^{v}_{\mathcal{A}}(v_i)$,

where *refScore*$^{v}_{\mathcal{A}}(v_i)$ = *refShare*$(v_i, \langle v, v_i\rangle)\ \zeta($*interaction*$(\langle v, v_i\rangle))$, $\zeta(n) = \frac{n}{n+C}$, a factor ramping up to 1 in the steady state and C is a configurable parameter. In all our experiments, we set the value of C to 10[1]. Our proposed technique to incorporate reciprocity into existing algorithms melding dual objectives is fairly general; the overall expression of *RAscore*$^{v,T}_{\mathcal{A}}(v_i)$ depends on the algorithmic score, *score*$^{v,T}_{\mathcal{A}}(v_i)$. For DIEL, *RAscore*$^{v,T}_{\texttt{DIEL}}(v_i)$ = $m(v_i) + \frac{s(v_i)}{\sqrt{n_i}}$ + *refScore*$^{v}_{\texttt{DIEL}}(v_i)$, while for TS, *RAscore*$^{v,T}_{\texttt{TS}}(v_i)$ = θ_i + *refScore*$^{v}_{\texttt{TS}}(v_i)$, where $m(v_i)$, $\frac{s(v_i)}{\sqrt{n_i}}$ and θ_i all are computed with respect to topic T. Since both the algorithmic score for DIEL and TS and *refScore* have identical range [0, 1], *RAscore*$^{v,T}_{\mathcal{A}}(v_i)$ has range [0, 2].

For any algorithm $\mathcal{A}$, if two colleagues v_i and v_j have identical scores *score*$^{v,T}_{\mathcal{A}}(v_i)$ and *score*$^{v,T}_{\mathcal{A}}(v_j)$ and *interactions*, the reciprocity-aware variant will select the colleague with greater *referral share* thus favoring colleagues who return the favor more often. As the value of *interaction*(v, v_i) increases, v becomes more sure of its estimate of the referral share thus putting more weight to referral share in its combined reciprocity-aware score computation. Note that, for a given expert, both *RAscore*$^{v,T}_{\mathcal{A}}(v_i)$ and *score*$^{v,T}_{\mathcal{A}}(v_i)$ are computed for a specific topic, however, *refShare* and *interaction* are computed across all topics. While the underlying approach to combine reciprocity with performance is simple, in a distributed multi-agent setting, several such threads of continual estimation and updates of referral shares and expertise of their colleagues are happening in parallel, thus creating a complicated mesh of interaction guided by self-interest, i.e., maximizing incoming referrals.

4 Experimental Setup

Performance measure: We considered two different performance measures. Following previous literature [12, 15], our first performance measure is the overall task accuracy of our multi-expert system. If a network receives n tasks of which m tasks are solved

[1] Additionally, we present experimental results in Table 3 indicating that the performance is not sensitive to the choice of C over a reasonable set of values.

(either by the *initial expert* or a *referred expert*), the overall task accuracy is $\frac{m}{n}$. Q, the per-instance query budget, is set to 2, 3 and 4. Each algorithm is run on a data set of 200 *referral scenarios* and the average over such 200 scenarios is reported in our results section. Similarly, we report the average reciprocity imbalance over 200 *referral scenarios*. We summarize each algorithm's performance with a pair $\langle a, b \rangle$ where 'a' denotes the overall task accuracy at the horizon (5000 samples per *subnetwork*) and 'b' denotes the reciprocity imbalance.

Algorithm class, upper bound and baseline: We define an algorithm class, $\mathcal{A} \in \{\texttt{DIEL}^Q, \texttt{TS}^Q\}$, with per-instance query budget $Q \in \{2, 3, 4\}$. For a proposed reciprocity-aware algorithm, the upper bound is its underlying reciprocity-agnostic action selection algorithm class, $\mathcal{A}$. We chose this upper bound to answer the following research questions: a) how much reciprocity imbalance is present in existing algorithms? b) and to what extent of improvement our modification brings in (in terms of reciprocity imbalance) at what cost of performance (in terms of overall task accuracy)?

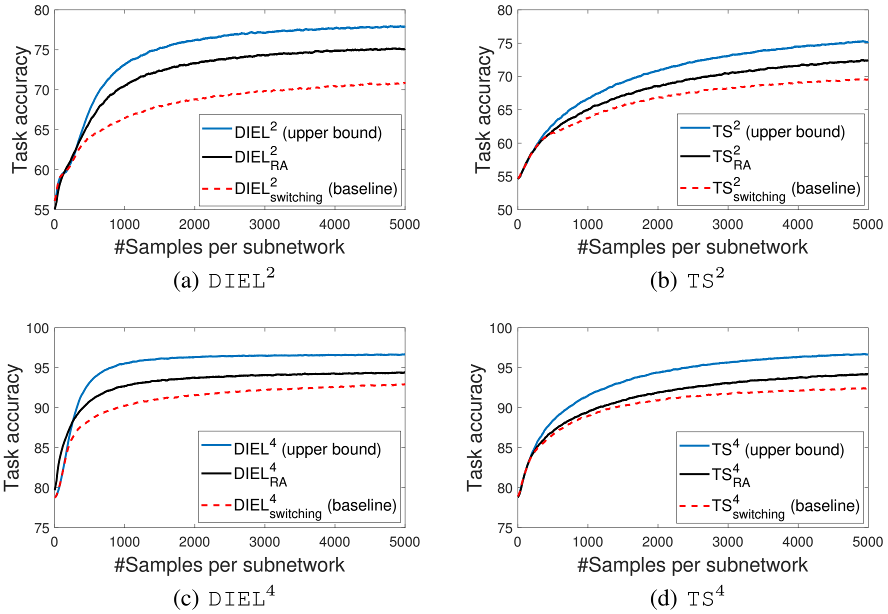
For an algorithm class $\mathcal{A}$, we propose the following switching variant, $\mathcal{A}_{switching}$, as baseline. Similar to $\mathcal{A}$, $\mathcal{A}_{switching}$ uses $\textit{score}_{\mathcal{A}}^{v,T}(v_i)$ (as opposed to $\textit{RAscore}_{\mathcal{A}}^{v,T}(v_i)$ used by $\mathcal{A}_{RA}$). However, for any referral link, after the *interaction* crosses a certain threshold (expressed through a parameter $\textit{interaction}_{\textit{thresh}}$), if the *referral share* of any of the participating two experts in the link falls below a threshold (expressed through a parameter $\textit{refShare}_{\textit{thresh}}$), the expert with smaller *referral share* disconnects and forms a new connection with another expert in the network. Essentially, this means after certain number of interactions between an expert pair, if an expert is unhappy with the reciprocity imbalance, she decides to form a new connection with another expert. Since every time a referral link is deleted a new referral link is formed, at any given point, the total number of referral links in network remains unchanged. For our experiments, $\textit{refShare}_{\textit{thresh}}$ is set to 0.3 and $\textit{interaction}_{\textit{thresh}}$ is set to 50.

For any algorithm class $\mathcal{A}$, we compare the performance of three algorithms: our proposed reciprocity-aware variant $\mathcal{A}_{RA}$, a baseline $\mathcal{A}_{switching}$, and an unconstrained upper bound. For instance, for $\texttt{DIEL}^2$ class, we compare $\texttt{DIEL}^2$ (the upper bound), $\texttt{DIEL}^2_{\textit{RA}}$ (proposed reciprocity-aware algorithm) and the switching variant $\texttt{DIEL}^2_{\textit{switching}}$.

Data set: We used the same data set used in [12]. The data set comprises of 200 *referral scenarios*. Each *referral scenario* consists of 100 experts connected through a referral network with a connection density of 16 ± 4.96 and 10 topics (for further details, see, e.g., [13, 15]).

5 Results

Substantial improvement in reciprocity imbalance at small performance cost: As the first step to establish the significance of this work, we need to analyze to what extent reciprocity is lacking in existing algorithms. For each algorithm class, Table 1 summarizes the reciprocity imbalance present in the upper bound, proposed corresponding reciprocity-aware versions and baseline switching variants. We first note that DIEL, the state-of-the-art referral learning algorithm and the upper bound for DIEL algorithm class, exhibits a substantially high reciprocity imbalance. In fact, both algorithms with-



(a) `DIEL`^{2}

(b) `TS`^{2}

(c) `DIEL`^{4}

(d) `TS`^{4}

Fig. 2. Performance comparison between reciprocity-aware referral-learning algorithms, corresponding unconstrained counterpart (upper bound) and switching variant (baseline). Qualitatively similar results for per-instance query budget $Q = 3$ are omitted due to space constraint.

out a mechanism to account for reciprocity suffer from high reciprocity imbalance and this phenomenon is independent of the per-instance query budget Q.

For both `DIEL` and `TS`, the improvement in reciprocity imbalance is highly noticeable in its corresponding reciprocity-aware version; each reciprocity-aware version brought about a 2-fold or better improvement in reducing the reciprocity imbalance (a 5x improvement in `DIEL`^{4}). While the switching variants are in general useful in reducing the reciprocity imbalance, our results indicate that our proposed reciprocity-aware solutions substantially outperform the switching variants. With an increase in query-budget, the imbalance in $\mathcal{A}$ or $\mathcal{A}_{switching}$ show no visible improvement, while $\mathcal{A}_{RA}$'s performance slightly improves. Since referrals are indivisible, with more budget, allocation gets smoother, and this accounts for $\mathcal{A}_{RA}$'s slight performance boost. For $\mathcal{A}_{switching}$, each switch requires estimating the new connections (and implicitly its subnetworks) expertise from scratch, which affects the reciprocity.

Understandably, the unconstrained upper bound achieved better task accuracy than our proposed solution as indicated in Figure 2 and Table 1. However, the performance gap is small and as we have already seen in Table 1, the resulting reduction in reciprocity imbalance is substantial. When compared with the switching baseline, mimicking a realistic algorithmic setting in which slighted peers disconnect from non-reciprocating colleagues and forge new links in the network, we found that our reciprocity-aware versions always performed better than the switching baseline on both performance measures. In terms of task accuracy, a paired t-test reveals that for all $\mathcal{A}$, beyond 1000

Algorithm class	$\mathcal{A}$	$\mathcal{A}_{RA}$	$\mathcal{A}_{switching}$
$\texttt{DIEL}^2$	⟨ **77.92**, 0.51 ⟩	⟨ 75.13, **0.19** ⟩	⟨ 70.84, 0.33 ⟩
$\texttt{DIEL}^3$	⟨ **92.45**, 0.55 ⟩	⟨ 89.71, **0.15** ⟩	⟨ 86.21, 0.34 ⟩
$\texttt{DIEL}^4$	⟨ **96.67**, 0.55 ⟩	⟨ 94.44, **0.11** ⟩	⟨ 92.95, 0.33 ⟩
$\texttt{TS}^2$	⟨ **75.24**, 0.36 ⟩	⟨ 72.44, **0.16** ⟩	⟨ 69.58, 0.28 ⟩
$\texttt{TS}^3$	⟨ **92.00**, 0.36 ⟩	⟨ 88.30, **0.13** ⟩	⟨ 85.39, 0.29 ⟩
$\texttt{TS}^4$	⟨ **96.64**, 0.36 ⟩	⟨ 94.17, **0.09** ⟩	⟨ 92.47, 0.29 ⟩

Table 1. Performance comparison of referral-learning algorithms. For any given algorithm class, the best task accuracy and reciprocity imbalance are highlighted in bold.

P_1	P_2	$\texttt{DIEL}^2_{switching}$	$\texttt{DIEL}^2_{RA}$	$\texttt{DIEL}^2$
0.35	20	⟨68.31, 0.33⟩	⟨ 75.13, **0.19** ⟩	⟨ **77.92**, 0.51⟩
0.35	50	⟨ 68.89, 0.28 ⟩		
0.35	100	⟨ 69.53, 0.33 ⟩		
0.30	20	⟨ 70.83, 0.37 ⟩		
0.30	50	⟨ 70.84, 0.33 ⟩		
0.30	100	⟨ 71.10, 0.36 ⟩		
0.25	20	⟨ 72.15, 0.42 ⟩		
0.25	50	⟨ 72.23, 0.38 ⟩		
0.25	100	⟨ 72.13, 0.39 ⟩		

Table 2. Performance analysis of different parameter configurations of $\texttt{DIEL}^2_{switching}$. Each row represents a parameter configuration with the left-most two columns indicating the parameter values. P_1 denotes *refShare*$_{thresh}$ and P_2 denotes *interaction*$_{thresh}$. The performance of $\texttt{DIEL}^2_{RA}$ and $\texttt{DIEL}^2$ is presented for reference. The best task accuracy and reciprocity imbalance are highlighted in bold.

samples or more per subnetwork, $\mathcal{A}_{RA}$ outperforms its switching counterpart, $\mathcal{A}_{switching}$, with p-value less than 0.0001. For every referral-learning algorithm class, our proposed solution achieved both better task accuracy and improved reciprocity than the corresponding switching baseline.

Robustness to parameter configurations: $\mathcal{A}_{switching}$ has two parameters: *refShare*$_{thresh}$ (set to 0.3), a threshold for the referral share, and *interaction*$_{thresh}$ (set to 50), a threshold on the *interaction* of a given *referral link* before the disgruntled expert decides to sever connection. $\mathcal{A}_{RA}$ has only one parameter, C set to 10. Table 2 demonstrates that for $\texttt{DIEL}^2$, when evaluated across a wide range of configurations, the reciprocity-aware $\texttt{DIEL}^2_{RA}$ consistently outperforms the corresponding switching variant; Table 3 demonstrates that the performance of $\texttt{DIEL}^2_{RA}$ and $\texttt{TS}^2_{RA}$ is not sensitive to choice of C over a reasonable set of values.

C	$\texttt{DIEL}^2_{RA}$	$\texttt{TS}^2_{RA}$
5	⟨ 75.15, 0.1893 ⟩	⟨ 72.25, 0.1604 ⟩
10	⟨ 75.13, 0.1891 ⟩	⟨ 72.44, 0.1616 ⟩
15	⟨ 74.92, 0.1875 ⟩	⟨ 72.32, 0.1625 ⟩
20	⟨ 74.98, 0.1879 ⟩	⟨ 72.45, 0.1633 ⟩

Table 3. Robustness to parameter C

Algorithm class	F_{greed}	$F_{altruism}$
DIEL^2	1.10	1.16
DIEL^3	1.60	1.12
DIEL^4	1.85	1.08
TS^2	1.08	1.07
TS^3	1.50	1.05
TS^4	1.64	1.04

Table 4. Strategic referral behavior

Reciprocity-awareness fetches more referrals in expectation: At network level, substantial improvement in reciprocity imbalance can be obtained at a small cost of task accuracy. However, the more important questions are

- What is the individual incentive to meld dual objectives while making referral decisions?
- Can an expert benefit from not following the protocol either by showing extreme altruism or unfettered greed?

Recall that, the reciprocity-aware score melds dual objective by combining algorithmic score and reciprocity:
$RAscore^{v,T}_{\mathcal{A}}(v_i)$ = $score^{v,T}_{\mathcal{A}}(v_i)$ + $refScore^{v}_{\mathcal{A}}(v_i)$. We now consider two extreme conditions: an expert showing absolute altruism and only using $score^{v,T}_{\mathcal{A}}(v_i)$; an expert showing unfettered greed and only using $refScore^{v}_{\mathcal{A}}(v_i)$. Accordingly, we define the following strategy set of an expert $S = \{altruism, greed, reciprocity\text{-}awareness\}$. For a given scenario $scenario_i$, we first fix one expert, say v^i_l. Apart from v^i_l, all other experts always adopt the same *reciprocity-awareness* strategy. Let $R^s(v^i_l)$ denote the number of referrals received by v^i_l in $scenario_i$ when she adopts strategy $s \in S$. We now calculate the following two factors:

$$F_{greed} = \frac{\sum\limits_{i=1}^{200} R^{s=reciprocity\text{-}awareness}(v^i_l)}{\sum\limits_{i=1}^{200} R^{s=greed}(v^i_l)}, \text{ and } F_{altruism} = \frac{\sum\limits_{i=1}^{200} R^{s=reciprocity\text{-}awareness}(v^i_l)}{\sum\limits_{i=1}^{200} R^{s=altruism}(v^i_l)}.$$

Table 4 shows that $\forall \mathcal{A}, F_{greed} > 1$ and $F_{altruism} > 1$. This implies that when all other experts follow the *reciprocity-awareness* strategy, deviating to *altruism* or *greed* fetches lesser number of referrals in expectation. It is straight-forward why *greed* would fetch less referrals as Q increases. Intuitively, if v^i_l adopts *altruism*, and the rest of the field requires reciprocity, connected colleagues will balance reciprocity with those colleagues at v^i_l's expense.

Teasing apart different factors in learning: In order to separate the effects of learning behavior from reciprocity considerations, we first consider a hypothetical situation where perfect knowledge about expertise is available. Acknowledging both that this situation is unlikely in a real setting and in our work we focus on a much harder problem of joint learning of reciprocity imbalance and expertise, this particular experiment allows us to take a cleaner look at the effect of reciprocity considerations. Specifically, we consider an algorithm, ORACLE^2, where each expert has an access to an oracle that accurately estimates the topical expertise of all expert/topic pairs. The $\langle$Accuracy, Reciprocity Imbalance$\rangle$ of ORACLE^2 and corresponding reciprocity aware version ORACLE$^2_{RA}$ are respectively: $\langle$79.43, 0.54$\rangle$ and $\langle$76.27, 0.24$\rangle$, i.e., we obtained

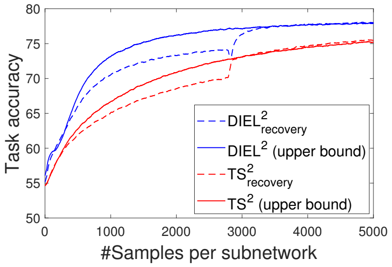


Fig. 3. Recovery performance

greater than 200% improvement in reciprocity imbalance at 3.98% loss of task accuracy.

Recovery performance: In terms of task accuracy, the performance gap between $\mathcal{A}$ and $\mathcal{A}_{RA}$ indicates reciprocity comes at a modest performance cost. However, it is important to analyze how much does *learning-to-refer* get affected because of this additional reciprocity constraint. In particular, we were interested in observing the performance of a learning algorithm that starts as constrained $\mathcal{A}_{RA}$, and somewhere in the process abruptly shifts to the unconstrained version $\mathcal{A}$, for instance in a crisis where task performance trumps all other considerations. We evaluate the following research question: can such learning algorithm identify the 'true' expert colleagues as well as a learning algorithm unconstrained from the start? The practical significance of this research question is there could be certain mission critical instances for which the network must find the best expert possible, regardless of reciprocity. If so, how fast can the algorithm match its performance with the unconstrained version? In Figure 3, after a randomly chosen point in the operation of the algorithm, $\mathcal{A}_{RA}$ switches to $\mathcal{A}$ (we denote this algorithm as $\mathcal{A}_{recovery}$). Our results indicate that neither $\texttt{DIEL}^2_{recovery}$ nor $\texttt{TS}^2_{recovery}$ had any difficulty in quickly re-establishing the performance of unconstrained $\texttt{DIEL}^2$ or $\texttt{TS}^2$ from the beginning. Note that, we opted for the same fixed point in time for all experts for clearer visualization; we obtained qualitatively similar performance even when the shifts are distributed across time-steps.

6 Conclusions and Future Work

In this paper, we argue that in the real-world, reciprocity in referral is a crucial practical factor. First we performed an extensive empirical evaluation focusing on two high-performance referral-learning algorithms and found that both of them suffer from substantial reciprocity imbalance. Second, we proposed algorithmic modifications to address reciprocity imbalance and we determined its efficacy empirically. Finally, we have shown our technique is extensible, and without any modification can be effectively applied to other algorithms or settings. Future lines of work include (1) expanding our investigation into other referral settings (e.g., [13]) and other active learning settings involving multiple teachers (e.g., [2, 17, 21, 27]), (2) addressing malicious agents and (3) considering fine-grained referrals.

Bibliography

[1] Agrawal, S., Goyal, N.: Analysis of thompson sampling for the multi-armed bandit problem. In: COLT. pp. 39–1 (2012)

[2] Ambati, V., Vogel, S., Carbonell, J.G.: Active learning and crowd-sourcing for machine translation (2010)

[3] Audibert, J.Y., Munos, R., Szepesvári, C.: Exploration–exploitation tradeoff using variance estimates in multi-armed bandits. Theoretical Computer Science **410**(19), 1876–1902 (2009)

[4] Berry, D.A., Fristedt, B.: Bandit problems: sequential allocation of experiments (Monographs on statistics and applied probability), vol. 12. Springer (1985)

[5] Chapelle, O., Li, L.: An empirical evaluation of thompson sampling. In: Advances in neural information processing systems (NIPS). pp. 2249–2257 (2011)

[6] De Marco, G., Immordino, G.: Reciprocity in the principal–multiple agent model. The BE Journal of Theoretical Economics **14**(1), 445–482 (2014)

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

[8] Graepel, T., Candela, J.Q., Borchert, T., Herbrich, R.: Web-scale bayesian click-through rate prediction for sponsored search advertising in microsoft's bing search engine. In: Proceedings of the 27th international conference on machine learning (ICML-10). pp. 13–20 (2010)

[9] Hütter, C., Böhm, K.: Cooperation through reciprocity in multiagent systems: an evolutionary analysis. In: The 10th International Conference on Autonomous Agents and Multiagent Systems-Volume 1 (AAMAS). pp. 241–248. International Foundation for Autonomous Agents and Multiagent Systems (2011)

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

[11] KhudaBukhsh, A.R., Carbonell, J.G.: Endorsement in referral networks. In: European Conference on Multi-Agent Systems. pp. 172–187. Springer (2018)

[12] KhudaBukhsh, A.R., Carbonell, J.G.: Expertise drift in referral networks. In: Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS). pp. 425–433. International Foundation for Autonomous Agents and Multiagent Systems (2018)

[13] KhudaBukhsh, A.R., Carbonell, J.G., Jansen, P.J.: Proactive Skill Posting in Referral Networks. In: Australasian Joint Conference on Artificial Intelligence. pp. 585–596. Springer (2016)

[14] KhudaBukhsh, A.R., Carbonell, J.G., Jansen, P.J.: Incentive compatible proactive skill posting in referral networks. In: European Conference on Multi-Agent Systems. Springer. pp. 29–43 (2017)

[15] KhudaBukhsh, A.R., Carbonell, J.G., Jansen, P.J.: Robust learning in expert networks: a comparative analysis. Journal of Intelligent Information Systems **51**(2), 207–234 (2018)

[16] 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)

[17] Murugesan, K., Carbonell, J.: Active learning from peers. In: Advances in Neural Information Processing Systems (NIPS). pp. 7011–7020 (2017)

[18] Sen, S., Sekaran, M.: Using reciprocity to adapt to others. In: International Joint Conference on Artificial Intelligence (IJCAI). pp. 206–217. Springer (1995)

[19] Settles, B.: Active learning. Synthesis Lectures on Artificial Intelligence and Machine Learning **6**(1), 1–114 (2012)
[20] Thompson, W.R.: On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. Biometrika **25**(3/4), 285–294 (1933)
[21] Urner, R., David, S.B., Shamir, O.: Learning from weak teachers. In: Artificial Intelligence and Statistics (AISTATS). pp. 1252–1260 (2012)
[22] Vogiatzis, G., MacGillivray, I., Chli, M.: A probabilistic model for trust and reputation. In: Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: volume 1-Volume 1. pp. 225–232. International Foundation for Autonomous Agents and Multiagent Systems (2010)
[23] Wang, Y., Singh, M.P.: Formal trust model for multiagent systems. In: IJCAI. vol. 7, pp. 1551–1556 (2007)
[24] Wiering, M., Schmidhuber, J.: Efficient model-based exploration. In: Proceedings of the Fifth International Conference on Simulation of Adaptive Behavior (SAB98). pp. 223–228 (1998)
[25] Yu, H., Shen, Z., Leung, C., Miao, C., Lesser, V.R.: A survey of multi-agent trust management systems. IEEE Access **1**, 35–50 (2013)
[26] Zamora, J., Millán, J.R., Murciano, A.: Learning and stabilization of altruistic behaviors in multi-agent systems by reciprocity. Biological cybernetics **78**(3), 197–205 (1998)
[27] Zhang, C., Chaudhuri, K.: Active learning from weak and strong labelers. In: Advances in Neural Information Processing Systems (NIPS). pp. 703–711 (2015)