Autonomous Agents and Multi-Agent Systems (2019) 33:645–671
https://doi.org/10.1007/s10458-019-09419-9

Expertise drift in referral networks

Ashiqur R. KhudaBukhsh[1] · **Jaime G. Carbonell**[1]

Published online: 6 August 2019
© Springer Science+Business Media, LLC, part of Springer Nature 2019

Abstract
Learning-to-refer is a challenge in expert referral networks, wherein Active Learning helps experts (agents) estimate the topic-conditioned skills of other connected experts for problems that the initial expert cannot solve and therefore must seek referral to experts with more appropriate expertise. Recent research has investigated different reinforcement action selection algorithms to assess viability of the learning setting both with uninformative priors and with partially available noisy priors, where experts are allowed to advertise a subset of their skills to their colleagues. Prior to this work, time-varying expertise drift (e.g., experts learning with experience) had not been considered, though it is an aspect that may often arise in practice. This paper addresses the challenge of referral learning with time-varying expertise, proposing `Hybrid`, a novel combination of `Thompson Sampling` and Distributed Interval Estimation Learning (`DIEL`) with variance reset, first proposed in this paper. In our extensive empirical evaluation, considering both biased and unbiased drift, the proposed algorithm outperforms the previous state-of-the-art (`DIEL`) and other competitive algorithms e.g., `Thompson Sampling` and `Optimistic Thompson Sampling`. We further show that our method is robust to topic-dependent drifts and expertise level-dependent drifts, and the newly-proposed $\texttt{DIEL}_{reset}$ can be effectively combined with other Bayesian approaches e.g., `Optimistic Thompson Sampling` and `Dynamic Thompson Sampling` and `Discounted Thompson Sampling` for improved performance.

Keywords Active Learning · Referral networks · Expertise drift

A preliminary version of this work appeared in [39]. The previous version contained an experimental bug due to an inadvertent error in our random sequence generation which we fixed and re-designed `Hybrid` accordingly. Our new design of Hybrid is more elegant and capable of producing qualitatively similar results to our previously published results. Additionally, this version contains a thorough robustness analysis considering topic-dependent drifts, expertise-level-dependent drifts, and combined topic-and-expertise drift. Extending our results to effectively combining other `Thompson Sampling` variants such as `Dynamic Thompson Sampling` [28], `Discounted Thompson Sampling` [54] and `Optimistic Thompson Sampling` [52], is also new. We also provide an extensive design-component analysis of `Hybrid` showing empirical evidence that any simpler design of `Hybrid` cannot match our current design's performance.

✉ Ashiqur R. KhudaBukhsh
akhudabu@cs.cmu.edu

Jaime G. Carbonell
jgc@cs.cmu.edu

[1] Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213, USA

1 Introduction

Learning-to-refer in agent or human referral networks is a recently proposed Active Learning challenge where experts (humans or autonomous agents) can redirect difficult instances (or problems) to more appropriate colleague experts based on estimates of the colleagues' topic-conditioned skills. Referral networks are common in human professional networks such as in clinical contexts and consultancy firms. Recent work [44,45] has compared a wide variety of referral learning algorithms in the stationary-expertise setting, i.e., where distributional parameters of expertise do not change over time. In that setting, Distributed Interval Estimation Learning (`DIEL`), a simple yet effective algorithm, was found to outperform other reinforcement learning methods, including: `UCB` variants [6,7], `Q-Learning` [32,61], `Thompson Sampling` variants [52,59], and ϵ-`Greedy` algorithms on both real and synthetic data [44]. A different direction along the lines of adversarial Machine Learning research [9,34,40–42] has proposed algorithms to work with partially available noisy priors and mechanisms to truthfully elicit such priors. Previous literature also considered robustness criterion such as capacity constraints and evolving networks where old experts leave the network and new experts join in at regular interval [40,44]. However, none of the past work addressed time-varying expertise that often arises in real world settings; expertise may change via refinement of existing skills with practice, acquisition of newer skills, decay of unpracticed skills, and could possibly depend on practical factors like fatigue, workload etc. Learning to track drifting expertise of colleagues in a referral network and thereby improve referrals is the primary focus of this paper.

The *partial information* [16] or the *information obstacle* [9] approches present in *multi-armed bandit* (MAB) settings (a gambler trying to maximize the total reward she receives by pulling one of the k arms at a time, where each arm has an unknown reward distribution) is a key perspective in referral networks too. When an expert refers a task to a colleague, there is no way to know how other colleagues would have performed on the same task. Moreover, local visibility of rewards, and the distributed nature of learning, i.e., each expert is independently estimating topical expertise of her colleagues, contributes to the challenges in *learning-to-refer*. For practical viability, early-learning-phase performance gain is crucial even over a large network, as we cannot afford an unbounded number of samples to estimate topical expertise. Understandably, *learning-to-refer* becomes even more challenging with non-stationary expertise since initially-weak experts who were discarded for future consideration on any given topic, could gain expertise over time, becoming real contenders who should not be ignored at a later time in optimizing referral decisions.

Our contributions are the following: First, we introduce time-varying expertise in referral networks, a practical consideration not previously addressed in the literature to the best of our knowledge. Second, in addition to bidirectional drift, the typical drift model in the literature, we also consider drift with a positive bias, where agents mostly improve with practice. Third, we propose `Hybrid`, a novel combination of `Thompson Sampling` (`TS`) and `DIEL` with variance reset. There is little established theoretical basis for the dynamic MAB setting (for example, `Dynamic Thompson Sampling` [30] has no known finite-horizon regret bound and `DIEL` (which outperforms `UCB` variants) is based on earlier algorithms with no known finite-horizon regret bound even in the static case). However, this paper is geared towards the design of a learning algorithm robust to expertise drift in referral networks, a challenging problem not previously studied, rather than a theoretical-bound analysis. Our empirical evaluation indicates that our proposed hybrid algorithm is more robust to expertise drift and tracks drift better than any existing method, including `DIEL` or `Thompson`

Sampling at the network level, improving overall referral accuracy. Additionally, we perform a design analysis establishing that both our proposed novelties: variance reset in DIEL and randomizing between DIEL and TS, are necessary for superior drift-tracking performance. Moreover, we show that other TS variants can be used to design similar hybrid algorithms with improved performance, a case which needs deeper exploration. Although our primary focus is on referral networks where aggregate task performance of the network is the performance measure, Hybrid's strong network-level performance encourages us to believe our work is applicable to the broader and more general context of multi-armed bandit setting with non-stationary reward distributions.

The rest of the paper is organized as follows. Previous work is summarized in Sect. 2. Section 3 lists the key research questions we focus on this paper. Section 4 presents our preliminaries on referral networks, key assumptions, and modeling choices for expertise drift. Sections 5, 6 and 7 describe the distributed learning algorithms we used for comparison, our experimental setup, and the results. We conclude by revisiting the research questions, presenting our main takeaways and outlining possible lines of future research in Sect. 8.

2 Related work

The referral framework draws inspiration from *referral chaining*, first proposed in [38] and subsequently extended in [22,26,67–70]. Referral learning in the context of an Active Learning framework was first proposed in [45], and subsequently extended [44] with performance comparisons over of a wide range of competing algorithms, multi-hop referrals, and consideration of practical factors such as capacity constraints and evolving networks. Most prior work considers non-informative priors [44,45]. In an augmented setting [40,41], similar to the line of research in multi-armed bandits with history where algorithms do not start from scratch [14,48,55], experts are allowed a one time local network advertisement of a subset of their skills which essentially extends the setting to partially available noisy priors where eliciting truthful advertisements and effectively initializing with the available priors are the primary challenges. In this paper, we work with the uninformative prior setting and consider time-varying expertise which none of the previous works in referral networks has addressed. Our results expose DIEL's, the state-of-the-art referral learning algorithm's susceptibility to expertise drift as we propose new algorithms that demonstrate superior tracking of drifting experts.

Our work on expertise drift fits in the broader context of multi-agent learning in non-stationary settings [1,15,36,53,56]. In the context of Proactive Learning, prior work on Interval Estimation Learning (the basic building block of DIEL) to track time-varying accuracy [19] used a particle filtering approach. Whereas this approach is elegant, it is infeasible in our case because it requires a large number of samples even for a single central learner, and the distributed nature of learning by each member of the referral network only exacerbates the problem.

From each expert's point of view, the core problem of learning appropriate referrals for a given topic can be viewed as a mutli-armed bandit (MAB) problem where referral choices are the arms. In the MAB literature, [25,65] are the first few seminal papers where time-varying reward distributions were introduced, which led to several contributions further addressing various nuances of the challenge (see, e.g., [13,29,31,50,63,71]). A wide range of subtleties like time instants when distributions change (at every step or at a smaller subset of steps), the fraction of arms whose distributions change (either every arm or the arm being

pulled), the nature of distributional change (monotonically decreasing [49] or arbitrary [57], adversarial [5,8,17] or non-adversarial) etc. have given rise to several existing settings (e.g., *restless bandits* [65], *rested bandits* [58], *rotting bandits* [49] etc.). Similar to [24], we consider the expertise distributions remain constant over epochs with unknown, arbitrary epoch lengths. However, our work is different from previous non-stationary bandits literature by introducing richer algorithm and operating in scale, with multiple agents learning several threads of referral policies for each topic. Moreover, standard Brownian perturbation, an often-adopted design choice [30,57] for modeling random drift is insufficient for capturing human expertise change, as it often improves with time and hence requires considering positively biased drifts. We also present a less common approach in tackling drift including concept drift [23,60] where the most popular approaches are window-based [24,30]. A related problem is that of fault detection-isolation [46]. However, the goals are different; as opposed to detecting the change and classifying the post-change distribution within a finite set of possibilities, we are primarily concerned with addressing the drift by incurring minimum possible regret. Our setting is also more complex with several possible change points; a similar problem is addressed in [4].

For expertise drift, we propose a hybrid algorithm that combines `Thompson Sampling`, and $\texttt{DIEL}_{reset}$, also proposed first in this paper. In order to switch between the same algorithm with different parameterizations, meta-bandits were proposed in [31]. We are unaware of any previous work in the MAB context where a Bayesian and a frquentist approach are randomized to obtain improved empirical performance. However, several lines of work in the past have studied adaptive strategies in algorithm design in other contexts. For instance, in the Active Learning literature, [20] obtained performance improvement over static strategies by adaptively updating strategy selection parameters. In a similar vein, [11] cast the problem of algorithm selection as an MAB problem and proposed a maximum entropy semi-supervised criterion to select between two high-performance Active Learning algorithms. In a completely different domain, `Hybrid` has its namesake in the form of a stochastic local search (SLS) solver of propositional satisfiability (SAT) instances. Similar to addressing our current challenge of balancing the trade-off between exploration and exploitation, in [64], a hybrid stochastic local search SAT solver that switches between two well-known SLS solvers is proposed to strike a balance between search diversification and intensification.

Previous research on referral networks considered a wide range of algorithms on stationary setting that includes `IEL`-based algorithm (Interval Estimation Learning) [35], `UCB` (Upper Confidence Bound) class of algorithms [2,6,7,47], ϵ-`Greedy` and its variants [7,43], `Q-Learning` algorithms [32,61] and `Thompson Sampling` variants [52,59]. In this work, we focus on two high-performance referral-learning algorithms: `DIEL`, the state-of-the-art in stationary setting and `Thompson Sampling`, an algorithm with a long history [59] that has received a recent surge of interest with proofs on finite-horizon bounds [3,37], empirical evidence of strong performance [18,44] in practical applications, and several recently-proposed algorithmic modifications to `Thompson Sampling` address the non-stationary setting [28,51,54]. Apart from randomizing between two MAB algorithms, our other novel contribution in this paper is $\texttt{DIEL}_{reset}$, an algorithm that resets its variance at regular intervals for superior drift-tracking. Our proposed variance reset is inspired by proactive-`DIEL`'s success in dealing with noisy partially available prior [41]. Variance to detect distributional change has been used in recent `Thompson Sampling` literature [51].

3 Research questions and challenges

In this paper, we focus on the following research questions:

1. **Are existing high-performance learning-to-refer algorithms vulnerable to time-varying expertise? If yes, how to track expertise drift in referral networks?**
   In [44], as the number of hops increased in a multi-hop referral setting, the performance gap between DIEL and Optimistic Thompson Sampling decreased with eventually Optimistic Thompson Sampling marginally outperforming DIEL, which indicates DIEL's vulnerability to non-stationary expertise. DIEL's. In this paper, we aim to conduct extensive experiments considering a wide range of drift scenarios to assess the extent of DIEL's vulnerability to expertise drift.
   All high-performance referral algorithms presented in the stationary expertise setting, such as DIEL or Thompson Sampling, will be able to detect an initially good expert whose performance deteriorates. However, if an expert initially exhibits low performance and then improves, designing referral algorithms that quickly detect such expertise shifts could be a challenging task, given that previously dismissed low performers would have a low probability of being sampled by the current algorithms.
2. **Previous literature provides experimental evidence of DIEL's early learning advantage and Thompson Sampling's (TS) competitive performance due to Bayesian exploration. Can DIEL or a variant of DIEL be effectively combined with TS variants for superior drift-tracking?**
   Algorithms exhibit a varying range of approaches to strike a balance between exploration and exploitation. Combining multiple algorithms could prove beneficial in tackling expertise drift.
3. **Is the newly-proposed Hybrid algorithm robust to both topic-dependent drifts and expertise-level-dependent drifts?**
   In real world, some topics may be prone to rapid skills change, whereas others are more stable. Also, certain experts may have natural abilities to learn some topics faster than others. Also, a strong expert is unlikely to lose or improve her skill rapidly, whereas a weak expert may be more likely to substantially improve in a short span of time, e.g., a student rapidly learning to become a true expert. Hence robustness to topic-dependent drifts and expertise-level-dependent drifts is crucial for practical performance.
4. **How crucial is the variance reset for improved performance?**
   The novelty in our proposed Hybrid algorithm is twofold: randomizing between a frequentist ($\texttt{DIEL}_{reset}$) and a Bayesian approach (TS), and performing a variance reset to facilitate exploitation. We seek to analyze the role of each design novelty in isolation and evaluate if Hybrid's design can be made any simpler without losing performance.
5. **Can DIEL with variance reset be combined with other Bayesian approaches for improved performance?**
   Finally, we are interested in exploring the generalizability of our approach through combining $\texttt{DIEL}_{reset}$ with other Thompson Sampling variants.

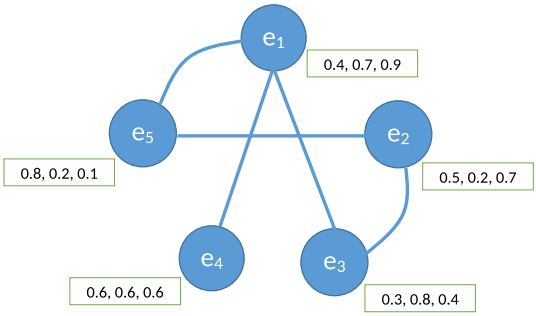


Fig. 1 A referral network with five experts

4 Referral networks

4.1 Motivation

We first illustrate how appropriately-targeted referrals can improve the overall performance of a network with a small simplified example of a referral network with five experts shown in Fig. 1 (this example is taken from [44]). Each node in the graph represents an expert, and the figures in brackets indicate an expert's topical expertise (probability of solving a given task) in three topics – t_1, t_2, and t_3. The bi-directional edges indicate a potential referral link, i.e., the experts 'know' each other and can send or receive referrals and communicate results.

Consider a query belonging to t_2, and without any referral, the client consults first e_2 and then e_5. In that case, the solution probability is $0.2 + (1 - 0.2) \times 0.2 = 0.36$. 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, e_2 could refer to e_3 for the best skill in t_2, leading to a solution probability of $0.2 + (1 - 0.2) \times 0.8 = 0.84$.

In an uninformed prior setting, 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 colleagues in a distributed setting with each expert independently estimating colleagues' topical expertise. Time-varying expertise poses an additional challenge to such estimation tasks as experts who may have been dismissed as weak can improve their skill over time, or other experts degrade their skills through disuse, requiring the referral learning algorithms to have a balanced re-sampling and re-estimation approach.

4.2 Preliminaries

Essentially, a *referral network* is a graph (V, E) of size k; each vertex v_i corresponds to an expert e_i $(1 \leq i \leq k)$ and each bidirectional edge $\langle v_i, v_j \rangle$ represents a *referral link* indicating e_i and e_j can refer problem instances to each other. A *subnetwork* of expert e_i is the set of her colleagues, i.e., the set of experts linked to an expert e_i by a referral link. A *referral scenario* consists of a set of m instances $(q_1, \ldots, q_m)$ belonging to n topics $(t_1, \ldots, t_n)$ are to be addressed by the k experts $(e_1, \ldots, e_k)$.

For a per-instance query budget of $Q = 2$, the referral mechanism for a task (we use task and instance interchangeably) q_j consists of the following steps.

1. A user (learner) issues an *initial query* to an expert e_i (*initial expert*) chosen uniformly at random from the network.

2. Expert e_i examines q_j and solves it if able and communicates the solution to the learner. This depends on the *expertise* (defined as the probability that e_i can solve q_j correctly) of e_i wrt. q_j.
3. If not, she issues a *referral query* to a *referred expert* within her subnetwork. The *Learning-to-refer* challenge is improving the estimate of who is most likely to solve the problem.
4. If the referred expert succeeds, she communicates the solution to the initial expert, who in turn, communicates it to the user.

The first two steps in our referral network are identical to Active Learning. Steps 3 and 4 are the extension to the Active Learning setting. Understandably, with a higher per-instance query budget $Q > 2$, step 4 can loop back to step 2 and the referred expert can re-refer instances to other experts as long as budget permits. Following [39,41,45], we set $Q = 2$ in all our experiments (we relax this assumption in Sect. 7.3.1).

We follow the same set of *assumptions* made in [45] a detailed description of which can be found in [41], but we remove the stationarity assumption on individual expert skills per topic. The more 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. We made the modeling choice regarding network connectivity because of the general observation that people sharing common expertise areas are more likely to know each other. Gaussian distributions are widely used to model real-valued random variables (e.g., height, weight, expertise) in natural and social sciences. For topical-expertise distribution, we considered a mixture of two 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.

4.3 Expertise drift

In previous work, [43,45], the expertise of an expert e_i on $topic_p$ was modeled as a truncated Gaussian distribution with small variance:

$$expertise(e_i, q_j \in topic_p) \sim \mathcal{N}(\mu_{topic_p,e_i}, \sigma_{topic_p,e_i}),$$
$$\forall p, i : \sigma_{topic_p,e_i} \leq 0.2, 0 \leq \mu_{topic_p,e_i} \leq 1.$$

We use a truncated Gaussian since *expertise* is a probability, it must remain within [0, 1]. Small variance implies an expert's within-topic expertise does not vary by a large amount. In a time-varying expertise setting, expertise of an expert e_i on $topic_p$ is expressed as

$$expertise(e_i, q_j \in topic_p) \sim \mathcal{N}(\mu_{topic_p,e_i,epoch_k}, \sigma_{topic_p,e_i}),$$
$$\mu_{topic_p,e_i,epoch_{k+1}} = \mu_{topic_p,e_i,epoch_k} + \mathcal{N}(\mu_{drift}, \sigma_{drift})$$

For convenience, we assume discrete changes at epoch boundaries, and within a given epoch, we assume the distributional parameters on expertise do not change appreciably. The epochs can be small, approximating continuous change. When μ_{drift} is 0, the unbiased drift is similar to the Brownian perturbation previously considered in [30]. The epochs can have arbitrary length and an expert has no knowledge of the epoch-lengths of their colleagues. After every discrete change, we ensure that $\mu_{topic_p,e_i,epoch_{k+1}}$ always remains within [0, 1] by setting it to 0 (or 1) if it is less than 0 (or greater than 1). Once $\mu_{topic_p,e_i,epoch_{k+1}}$ reaches

the boundary (0 or 1), we assume that it remains there until drift in the opposite direction moves it away from the boundary.

The expertise of people often improve over time by acquiring a new skill, explicit learning on how to improve a skill, or just practice through solving more problems. We consider this case in our positive-bias drift (with $\mu_{drift} > 0$), where the overall expertise of the experts in the network improves on certain topics over time.

4.4 Reward assumptions

From the point of view of a single expert, for a given topic, learning a referral policy maps to the classic *multi-armed bandit 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 known. In order to learn an effective referral strategy, whenever an expert refers a task to her colleague, she assigns a reward to the referred colleague depending on whether the task was successfully completed. Computational aspects (e.g., what type of information regarding the sequence of rewards is necessary?, how to score an expert depending on her past performance?) of the referral decision are described in the following section; here we outline the main assumptions related to rewards.

- *bounded* All our rewards are bounded within the the range [0, 1]. In all our experiments, we considered binary rewards, with a failed and successful task 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* Rewards are both locally assigned and locally visible. For example, $reward(e_i, t, e_j)$, a function of initial expert e_i, referred expert e_j and topic t, is assigned by e_i and visible to e_i only.

5 Distributed referral learning

As previously mentioned, considering a single expert and a given topic, *learning-to-refer* is an action selection problem (the problem of selecting an appropriate referral maps to selecting an effective arm in the *multi-armed bandit* setting). 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 fix topic to T and expert to e.

Let $q_1, \ldots, q_N$ be the first N referred queries belonging to topic T issued by expert e to any of her K colleagues denoted by $e_1, \ldots, e_K$. For each colleague e_j, e maintains a reward vector $\mathbf{r}_{j,n_{e_j}}$ where $\mathbf{r}_{j,n_{e_j}} = (r_{j,1}, \ldots, r_{j,n_{e_j}})$, i.e., the sequence of rewards observed from expert e_j on issued n_{e_j} referred queries. Hence, $N = \sum_{j=1}^{K} n_{e_j}$. Let $m(e_j)$ and $s(e_j)$ denote the sample mean and sample standard deviation of these reward vectors. Some of the algorithms we consider require initializing these reward vectors; we will explicitly mention any such initialization. In addition to the reward vectors, for each colleague e_j, e maintains S_{e_j} and F_{e_j} where S_{e_j} denotes the number of observed successes (reward = 1) and F_{e_j} denotes the number of observed failures (reward = 0). Clearly, without any initialization of the reward vectors, $\forall (S_{e_j} + F_{e_j}) > 0, m(e_j) = \frac{S_{e_j}}{S_{e_j}+F_{e_j}}$ (i.e., empirical mean is the ratio of total number of observed successes and total number of observations).

Like any other action selection problem, *learning-to-refer* also poses the classic exploration-exploitation trade-off: on one hand, we would like to refer to an expert who has performed well in the past on this topic (exploitation), and on the other hand we want to perform enough exploration to make sure we are not missing out on stronger experts for the topic. In the next subsection, we provide a short description of different action selection algorithms. We first describe three existing algorithms from the literature: `DIEL`, the state-of-the-art in the stationary, uninformed prior setting; `Thompson Sampling`, an important component of our proposed algorithm—`Hybrid`; `Optimistic Thompson Sampling`, a well-known variant with theoretical guarantees in the stationary setting [52]. Next, we present $\texttt{DIEL}_{reset}$, a `DIEL` variant first proposed in this paper which is a key building block of `Hybrid`. Finally, we present `Hybrid`, our main new contribution. At a high level, each of the algorithms computes a score for every expert e_j (denoted by $score(e_j)$) and selects the expert with the highest score breaking any remaining ties randomly. In our algorithm description, we chose a simplified notation and implicitly included topic T as an input, noting T explicitly only to indicate that all referral queries are on instances belonging to topic T and we are describing the learning thread for expert–topic pair $\langle e, T \rangle$.

5.1 Action selection algorithms

DIEL: Distributed Interval Estimation Learning (`DIEL`) is the known state-of-the-art referral learning algorithm [44,45] in the stationary, uninformed prior setting. `DIEL` is based on Interval Estimation Learning which was first proposed in [35] and has been extensively used in stochastic optimization [21] and action selection problems [12,66]. As described in Algorithm 1, at each step, `DIEL` [41] selects the expert e_j with highest $m(e_j) + \frac{s(e_j)}{\sqrt{n_{e_j}}}$. Every action is initialized with two rewards of 0 and 1, allowing us to initialize the mean and variance.

The intuition behind selecting an expert with a high expected reward ($m(e_j)$) and/or a large amount of uncertainty in the reward ($s(e_j)$) is the following. A large variance implies greater uncertainty, indicating that the expert has not been sampled with sufficient frequency to obtain reliable estimates. Selecting such an expert is an *exploration step* which will increase the confidence of e in her estimate. Such exploration steps have the potential of identifying a highly skilled expert. Selecting an expert with a high $m(e_j)$ amounts to exploitation. Initially, the choices made by e 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.

Thompson Sampling (TS) `Thompson Sampling` was first proposed in the 1930's [59] and the finite-time regret bound remained unsolved for decades [3] until recent results on its competitiveness with other algorithms with provable regret bounds renewed interest [18,27].

As described in Algorithm 2, at each step, for each expert e_j, `TS` first samples θ_j from $Beta(S_{e_j} + 1, F_{e_j} + 1)$. We briefly recall that *Beta* distributions form a family of continuous probability distributions on the interval (0, 1) parameterized by two positive shape parameters, α and β. The probability density function of $Beta(\alpha, \beta)$, is given by $f(x; \alpha, \beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1}(1-x)^{\beta-1}$, where for any positive integer z, $\Gamma(z) = (z-1)!$. Next, `TS` selects the action with highest θ_j. When the number of observations is 0, θ_j is sampled from $Beta(1, 1)$, which is uniform distribution on (0, 1), i.e., $U(0, 1)$; this makes all colleagues equally likely to receive referrals. 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.

Algorithm 1: DIEL(e, T)

Initialization:
$\forall e_j \in subnetwork(e), n_{e_j} \leftarrow 2, \mathbf{r}_{j,n_{e_j}} \leftarrow (0, 1)$
Main loop:
foreach *referral query* **do**
  foreach $e_j \in subnetwork(e)$ **do**
    $score(e_j) \leftarrow m(e_j) + \frac{s_{e_j}}{\sqrt{n_{e_j}}}$
  end
  $best = \arg\max_j score(e_j)$
  Observe reward r after referring to e_{best}
  Update:
  Update $\mathbf{r}_{best,n_{e_{best}}}$ with r, $n_{e_{best}} \leftarrow n_{e_{best}} + 1$
end

Algorithm 2: TS(e, T)

Initialization:
$\forall e_j \in subnetwork(e), S_{e_j} \leftarrow 0, F_{e_j} \leftarrow 0$
Main loop:
foreach *referral query* **do**
  foreach $e_j \in subnetwork(e)$ **do**
    $\theta_j \sim Beta(1 + S_{e_j}, 1 + F_{e_j})$
  end
  $best = \arg\max_j \theta_j$
  Observe reward r after referring to e_{best}
  Update:
  $S_{e_{best}} \leftarrow S_{e_{best}} + r$
  $F_{e_{best}} \leftarrow F_{e_{best}} + 1 - r$
end

Optimistic Thompson Sampling (Optimistic TS) Optimistic TS [52] is very similar to TS with an additional restriction: θ_j is never allowed to be less than the mean observed reward $m(e_j)$; θ_j is set to $m(e_j)$ whenever it is less than $m(e_j)$ (in the boundary condition when number of observed samples is zero, $m(e_j)$ is considered to be zero). The reason this sampling technique is called optimistic is because this technique always assumes that the true mean is at least as high as the sampled mean. Note that, each time we refer to e_j where $\theta_j > m(e_j)$, we are essentially performing an *exploration step*.

DIEL$_{reset}$, described here, is a component of Hybrid. As the number of observed samples increases, DIEL becomes more exploitative since the relative importance of DIEL's variance term diminishes and the mean observed reward, $m(e_j)$, exerts more influence in the overall score of an expert. In a static setting, favoring experts with historically strong performance has an intuitive appeal, and DIEL's success in this setting can be attributed to its early exploration (through the variance term) followed by aggressive exploitation. However, in order to succeed in a time-varying expertise setting, an algorithm needs to continually explore. In DIEL$_{reset}$, we propose a partial reset to facilitate exploration. We break down the referrals into referral-windows of w referrals where w is a configurable parameter. After each referral-window w_l, for each expert e_j, we reset n_{e_j} to 2 and $\mathbf{r}_{j,n_{e_j}} = (m(e_j), m(e_j))$, i.e., we summarize our past observations into two observed rewards equal to the mean-observed reward and while resetting the variance to 0.

Algorithm 3: `Optimistic TS`(e, T)

Initialization:
$\forall e_j \in subnetwork(e), S_{e_j} \leftarrow 0, F_{e_j} \leftarrow 0$
Main loop:
foreach *referral query* **do**
  foreach $e_j \in subnetwork(e)$ **do**
    $\theta_j \sim Beta(1 + S_{e_j}, 1 + F_{e_j})$
    if $(S_{e_j} + F_{e_j}) == 0$ **then**
      $score(e_j) \leftarrow \theta_j$
    else
      $score(e_j) \leftarrow max(\theta_j, \frac{S_{e_j}}{S_{e_j}+F_{e_j}})$
    end
  end
  $best = \arg\max_j score(e_j)$
  Observe reward r after referring to e_{best}
  Update:
  $S_{e_{best}} \leftarrow S_{e_{best}} + r$
  $F_{e_{best}} \leftarrow F_{e_{best}} + 1 - r$
end

Algorithm 4: `DIEL`$_{reset}(e, T)$

Initialization:
$\forall e_j \in subnetwork(e), n_{e_j} \leftarrow 2, \mathbf{r}_{j,n_{e_j}} \leftarrow (0, 1)$
Main loop:
foreach *referral window* **do**
  foreach *referral query* **do**
    foreach $e_j \in subnetwork(e)$ **do**
      $score(e_j) \leftarrow m(e_j) + \frac{s_{e_j}}{\sqrt{n_{e_j}}}$
    end
    $best = \arg\max_j score(e_j)$
    Observe reward r after referring to e_{best}
    Update:
    Update $\mathbf{r}_{best,n_{e_{best}}}$ with r, $n_{e_{best}} \leftarrow n_{e_{best}} + 1$
  end
  Reset:
  foreach $e_j \in subnetwork(e)$ **do**
    $n_{e_j} \leftarrow 2$
    $\mathbf{r}_{j,n_{e_j}} \leftarrow (m(e_j), m(e_j))$
  end
end

The partial reset is inspired by proactive-`DIEL`'s [41] recent success in dealing with partially available (potentially noisy) priors. We first present a short introduction of the augmented learning setting and proactive-`DIEL`'s initialization to provide better insight to this design decision. Proactive skill posting is an augmented referral-learning setting [41,42], where agents are allowed a one-time local network advertisement of a subset of their skills. After such advertisement in a truthful setting where agents can accurately estimate their own skills, for each colleague expert e_j, for any given topic T, e has an initial estimate of expertise,

Algorithm 5: `Hybrid`(e, T)

Initialization:
$\forall e_j \in subnetwork(e), n_{e_j} \leftarrow 2, \mathbf{r}_{j,n_{e_j}} \leftarrow (0, 1)$
$S_{e_j} \leftarrow 0, F_{e_j} \leftarrow 0$
Main loop:
foreach *referral window* **do**
  foreach *referral query* **do**
    Assign algorithm $\mathcal{A}$ to `DIEL` or `TS` uniformly at random
    foreach $e_j \in subnetwork(e)$ **do**
      if $\mathcal{A}$ = *DIEL* **then**
        $score(e_j) \leftarrow m(e_j) + \frac{s_{e_j}}{\sqrt{n_{e_j}}}$
      else
        $score(e_j) \sim Beta(1 + S_{e_j}, 1 + F_{e_j})$
      end
    end
    $best = \arg\max_j score(e_j)$
    Observe reward r after referring to e_{best}
    Update:
    Update $\mathbf{r}_{best,n_{e_{best}}}$ with $r, n_{e_{best}} \leftarrow n_{e_{best}} + 1$
    $S_{e_{best}} \leftarrow S_{e_{best}} + r, F_{e_{best}} \leftarrow F_{e_{best}} + 1 - r$
  end
  Reset:
  foreach $e_j \in subnetwork(e)$ **do**
    $n_{e_j} \leftarrow 2$
    $\mathbf{r}_{j,n_{e_j}} \leftarrow (m(e_j), m(e_j))$
  end
end

$\hat{\mu}_{j,T}$ such that $\hat{\mu}_{j,T} \geq \mu_{j,T}$, where $\mu_{j,T}$ is true topical expertise of colleague e_j on topic T. Instead of initializing `DIEL` with a pair of rewards $\langle 0, 1 \rangle$, proactive-`DIEL` initializes each expert with a pair of reward $\langle \hat{\mu}_{j,T}, \hat{\mu}_{j,T} \rangle$, effectively initializing variance to 0, number of observed samples to 2 and mean-observed reward to $\hat{\mu}_{j,T}$. Empirical evaluation has shown that proactive-`DIEL` is robust to noise in self-skill estimates and strategic misreporting of skills. In our case, incentive compatibility is not a concern since we actually observe the rewards we are using to reset the mean and the variance. However, in a time-varying setting, historical mean can be viewed as a noisy estimate of the current true mean. Proactive-`DIEL`'s robustness to noisy self-skill estimates encouraged us to explore this variance reset idea to facilitate exploration.

We are now ready to describe `Hybrid`, our primary contribution in this paper.

Hybrid As presented in Algorithm 5, at each step `Hybrid` randomizes between selecting $\texttt{DIEL}_{reset}$ and `Thompson Sampling` (`TS`). After each referral window, the variance and mean of $\texttt{DIEL}_{reset}$ are reset and after each referral, irrespective of the chosen algorithm, $m(e_j)$, $s(e_j)$, n_{e_j}, S_{e_j} and F_{e_j}, all are updated accordingly. Effectively, in `Hybrid`, two different MAB algorithms benefit from each others exploration and exploitation. $\texttt{DIEL}_{reset}$'s episodic exploration through the variance reset at regular interval is bolstered by `TS`'s continual exploration. Although we chose the most widely known, simplest and oldest `TS` algorithm [59], `Hybrid`'s design is flexible and can be extended to other `TS` variants. In fact, in Sect. 7.4, we present experimental results after extending the `Hybrid` design to

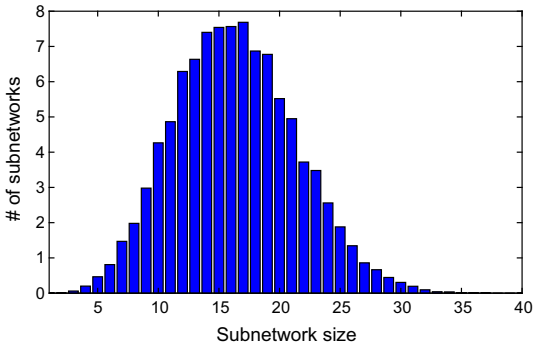


Fig. 2 Distribution of *subnetwork* size

`Optimistic Thompson Sampling` [52], `Dynamic Thompson Sampling` [28], and `Discounted Thompson Sampling` [54], with good results.

6 Experimental setup

Baselines and upper bounds We use `DIEL` as our primary baseline; it is the previously-known state-of-the-art referral learning algorithm for the stationary setting. As additional baselines we include two `Thompson Sampling` variants, and we also report two topical upper bounds for performance comparison. `Thompson Sampling` variants and the `DIEL` version used [41,43] are parameter free. We refer all three algorithms, `DIEL`, `TS`, and `Optimistic TS`, as challengers. We considered two upper bounds: `Drift-blind` and `Drift-aware`. The `Drift-aware` upper bound is the performance of a network where every expert has access to an oracle that knows the true topic-mean (i.e., $\mu_{topic_p,e_j,epoch_k}$ where k is the current epoch) of every expert–topic pair $\langle e_j, topic_p \rangle$. At all points in the simulation, every expert refers to the colleague with highest distribution mean on that topic at that point of time. Effectively, the `Drift-aware` upper bound mimics the performance of a network where every expert starts with a perfect estimation of her colleagues' expertise and precisely tracks subsequent drifts without requiring learning. The `Drift-blind` "upper bound" is the performance of a network where every expert has access to an oracle that only knows the true topic-mean of every expert–topic pair at the beginning of the simulation but ignores any subsequent drift. This means for every instance belonging to $topic_p$, an expert e will always refer to e_{best} such that $e_j \in subnetwork(e)$ and $best = \arg\max_j(\mu_{topic_p,e_j,epoch_1})$.
Data set Our test set for performance evaluation is the same data set used in [40],[1] which is a random subset of 200 *referral scenarios* also used in [41,43,45]. Each *referral scenario* consists of a network of 100 experts and 10 topics. The distribution of *subnetwork* size is presented in Fig. 2; the average connection density is 16.36 ± 5.03. In our simulation, we start with the same parameter values describing topical expertise of each expert. As the simulation progresses, the expertise drifts according to the drift parameter values are described in Table 1. For modeling expertise drift, we believe a gradual change in expertise is more realistic than abrupt changes. Hence, we considered the distribution for expertise as piece-wise stationary and selected small values for μ_{drift} and σ_{drift}. Recall that in a time-varying expertise setting, expertise of an expert e_i on $topic_p$ is modeled as

[1] The data set can be downloaded from https://www.cs.cmu.edu/~akhudabu/referral-networks.html.

Table 1 Drift parameters

Drift distribution	Drift nature	μ_{drift}	σ_{drift}
$\mathcal{D}_1$	Weak, unbiased	0	0.03
$\mathcal{D}_2$	Strong, unbiased	0	0.06
$\mathcal{D}_3$	Weak, small positive bias	0.005	0.03
$\mathcal{D}_4$	Strong, small positive bias	0.005	0.06
$\mathcal{D}_5$	Strong, large positive bias	0.01	0.06

$\mu_{topic_p,e_i,epoch_{k+1}} = \mu_{topic_p,e_i,epoch_k} + \mathcal{N}(\mu_{drift}, \sigma_{drift})$. We use #samples as a proxy for time as is typical in Machine Learning for evolving or streaming scenarios. For each expert, the epoch boundaries are chosen uniformly at random. The total number of epochs for a given topic is set to 40 (with 10 topics, this essentially means, the total number of time the expertise of an expert changes is 400).

In addition to drift distributions $\mathcal{D}_1, \ldots, \mathcal{D}_5$, we also used topic-dependent drift distribution, $\mathcal{D}_{\mathcal{T}}$, and combined expert–topic dependent drift distribution, $\mathcal{D}_{e,\mathcal{T}}$. We describe these two distributions in Sect. 7.2.

Performance Measure We use the same performance measure, overall task accuracy of our multi-expert system, as in previous work in referral networks. So if a network receives n tasks of which m tasks are solved (either by the *initial expert* or the *referred expert*), the overall task accuracy is $\frac{m}{n}$. Q, the per-instance query budget, is set to 2. Each algorithm is run on the data set of 200 referral networks and the average over such 200 simulations is reported in our results section. In order to facilitate comparability, for a given simulation across all algorithms, we chose the same sequence of initial expert and topic pairs; for each expert in a network, the epoch length and expertise shift for each given topic are identical across different referral algorithm runs.

Algorithm Configuration For all our experiments, `Hybrid`'s parameter w is set to 100. In Sect. 7.4, we demonstrate that over a reasonable set of choices for w, `Hybrid`'s performance is not highly sensitive. Additionally, we found that instead of a one-size-fits-all approach, for each expert, w can be set to the square of the expert's *subnetwork* size without any loss of performance.

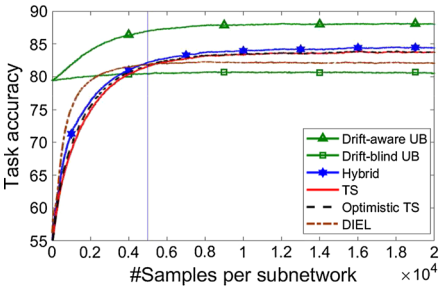
Computational Environment Experiments were carried out on Matlab R2016 running Windows 10.

7 Results

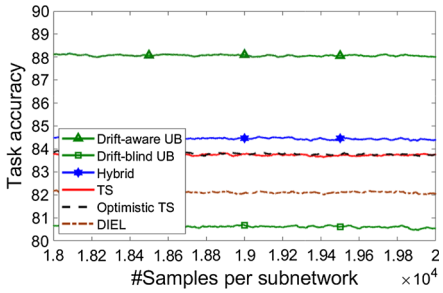
7.1 Performance comparison with drift distributions uniform across networks

Figure 3 summarizes the performance of referral-learning algorithms with drift distributions $\mathcal{D}_1$, $\mathcal{D}_4$ and $\mathcal{D}_5$ listed in Table 1 (we omit qualitatively similar results on $\mathcal{D}_2$ and $\mathcal{D}_4$). Our results demonstrate the following points:

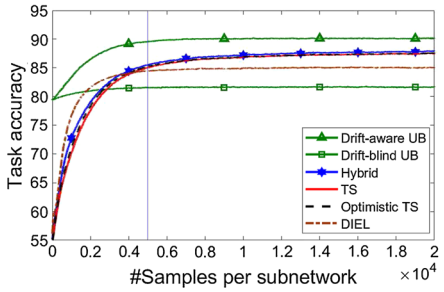
First, the `Drift-aware` upper bound outperforms the `Drift-blind` by a considerable margin, as expected. In fact, all algorithms eventually outperformed the `Drift-Blind` upper bound. This underscores the importance of tracking drift in expertise estimation and continual learning, since starting with a perfect information on the topical mean of every



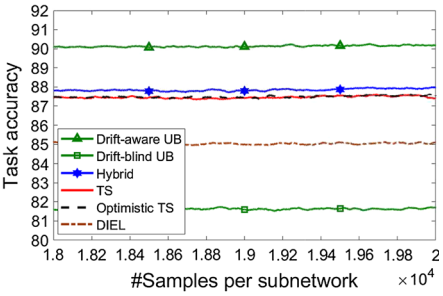
(a) Small positive bias, strong drift



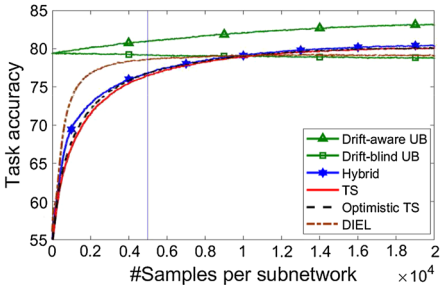
(b) Steady-state performance closeup: zooming in to 3(a) upper right portion



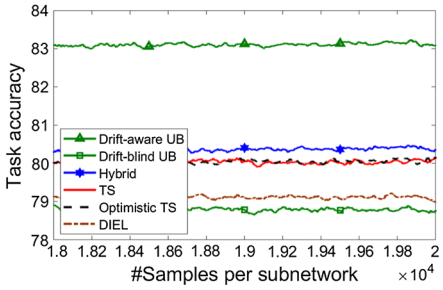
(c) Large positive bias, strong drift



(d) Steady-state performance closeup: zooming in to 3(c) upper right portion



(e) Unbiased, weak drift



(f) Steady-state performance closeup: zooming in to 3(e) upper right portion

Fig. 3 Performance comparison of referral learning algorithms

expert–topic pair was not enough to overcome expertise-drift, even if the expertise tracking methods start with imperfect estimates.

Next, we evaluate the relative expertise-tracking performance of algorithms in the literature. The vertical line at 5000 samples per subnetwork marks the horizon considered in previously reported results. Earlier results demonstrated `DIEL` outperformed several algorithms including `UCB` variants, `Q-Learning` variants [43–45] in the stationary expertise setting. In our new results, we find that in the presence of unbiased drift, `DIEL` still outperforms the `TS` variants when the number of observed samples is small, once again highlighting the early performance gain that made `DIEL` suitable for multi-hop referral learning and

proactive skill posting. However, with a larger number of samples under the expertise-drift condition, we find that both TS algorithms eventually outperform DIEL, thus presenting better long-term steady-state performance due to superior tracking of drifting experts. In the presence of positive bias drift, a particular case we are interested in evaluating, we found the TS algorithms outperform DIEL even when the number of observed samples is small. Much of DIEL's success stems from early identification of strong candidates and then aggressively pursuing them. However, with positive biased drift, weaker experts may improve and if an algorithm fails to detect those late-bloomers, it will perform sub-optimally.

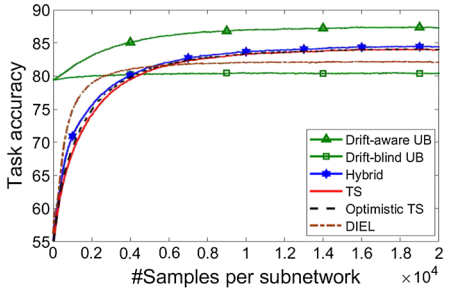
Finally, we focus on Hybrid, our primary proposed algorithm. As shown in Fig. 3, Hybrid outperformed all three challenger algorithms by effectively combining exploration through TS and exploitation through our new DIEL variant with variance reset. With stronger drift, the performance gap between Hybrid and the challenger widens indicating Hybrid's superior drift tracking performance. We next present a thorough robustness analysis considering less restrictive drift settings and a component analysis of Hybrid pin-pointing particular design choices that are crucial for Hybrid's success.

7.2 Robustness analysis

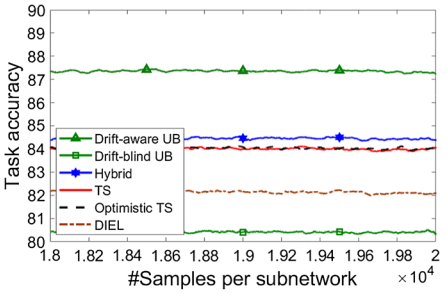
So far, we assumed the distribution parameters for drift do not vary across topics. However, in reality, some topics may be prone to rapid skill evolution, whereas others are more stable. Also, certain experts may have natural abilities to learn some topics faster than others. In our current set of experiments, we consider two additional drift distributions: $\mathcal{D}_{\mathcal{T}}$, and $\mathcal{D}_{e,\mathcal{T}}$. In $\mathcal{D}_{\mathcal{T}}$, for any given topic t, the underlying drift distribution is chosen uniformly at random from $\mathcal{D}_1, \ldots, \mathcal{D}_5$ and the distribution remains the same across all experts. In $\mathcal{D}_{e,\mathcal{T}}$, for a given expert topic pair, $\langle e, t \rangle$, the underlying drift distribution is chosen uniformly at random from $\mathcal{D}_1, \ldots, \mathcal{D}_5$. This implies that $\mathcal{D}_{e,\mathcal{T}}$ is by far the least restrictive distribution allowing experts a flexible choice from $\mathcal{D}_1, \ldots, \mathcal{D}_5$. For our remaining experiments, we will focus on $\mathcal{D}_{\mathcal{T}}$, and $\mathcal{D}_{e,\mathcal{T}}$ as they represent more realistic drift conditions.

Figure 4 summarizes the referral-learning performance on these two new distributions. Our findings are consistent with our previous results; Hybrid outperforms all challengers and the Drift-blind upper bound on both distributions. Since for each expert, Hybrid maintains a separate learning thread for a given topic, we expected Hybrid to perform better than the challengers on $\mathcal{D}_{\mathcal{T}}$. However, $\mathcal{D}_{e,\mathcal{T}}$ is a much harder case where even within the same topic different experts may exhibit different drift behavior. Hybrid's superior drift-tracking on $\mathcal{D}_{e,\mathcal{T}}$ indicates robustness to drift behavior which we further stress-test on our next series of experiments involving expertise-level dependent drifts.

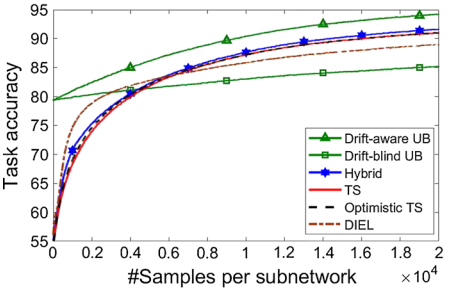
Before we describe our next set of results involving expertise-level dependent drifts, we make a quick digression to highlight a subtle point. In both $\mathcal{D}_{\mathcal{T}}$ and $\mathcal{D}_{e,\mathcal{T}}$, the choice of drift distribution is uniform at random; the only distinction is that in one case ($\mathcal{D}_{\mathcal{T}}$), the choice is fixed for a given topic across all experts and in another ($\mathcal{D}_{e,\mathcal{T}}$), it varies across different expert–topic pairs. Nonetheless, the total number of experts exhibiting a specific drift behavior ($\in \{\mathcal{D}_1, \ldots, \mathcal{D}_5\}$) has the same expected value. However, we notice that the performance of both the Drift-aware upper bound and Hybrid on $\mathcal{D}_{e,\mathcal{T}}$ is better than their respective performances on $\mathcal{D}_{\mathcal{T}}$. This actually highlights that while the number of experts exhibiting certain drift behavior remains roughly the same, by allowing different experts exhibiting different drift behaviors across the same topic, we allow the network to be more versatile; following the primary essence of the referral network setting: "different agents have different



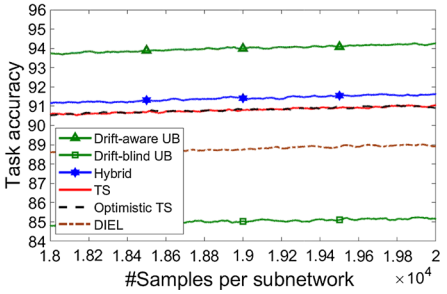
(a) Drift distribution $\mathcal{D}_{\mathcal{T}}$



(b) Steady-state performance closeup: zooming in to 4(a) upper right portion



(c) Drift distribution $\mathcal{D}_{e,\mathcal{T}}$

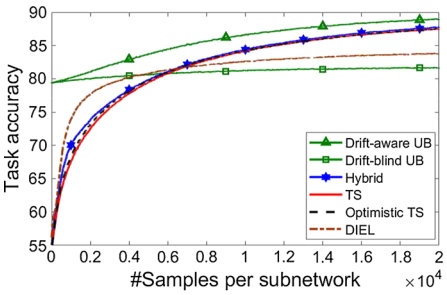


(d) Steady-state performance closeup: zooming in to 4(c) upper right portion

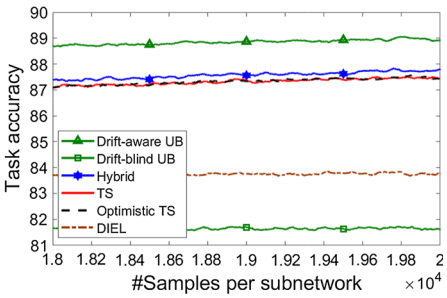
Fig. 4 Robustness analysis

strengths, referring to the most appropriate agent would improve overall performance", we notice just such a performance boost.

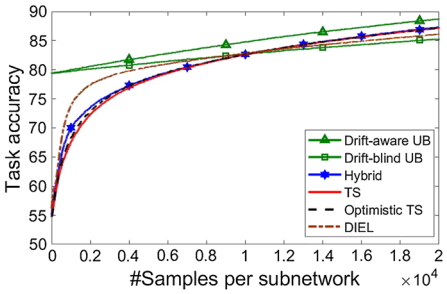
In all our experiments so far, we assumed that the nature of drift is independent of the expertise level of each expert. However, a strong expert is unlikely to lose expertise and also has less headroom for rapid additional improvement, whereas a weak expert may be more likely to substantially improve in a short span of time if they focus their efforts, e.g., a student rapidly learning to become a true expert. We incorporate this in our next set of experiments where drift is also a function of current expertise. Recall that, in a time-varying expertise setting, expertise of an expert e_i on $topic_p$ is modeled as $\mu_{topic_p,e_i,epoch_{k+1}} = \mu_{topic_p,e_i,epoch_k} + \mu_\Delta$ where $\mu_\Delta \sim \mathcal{N}(\mu_{drift}, \sigma_{drift})$. In an expertise-level dependent drift setting, $\mu_{topic_p,e_i,epoch_{k+1}} = \mu_{topic_p,e_i,epoch_k} + \frac{1+\epsilon-\mu_{topic_p,e_i,epoch_k}}{1+\epsilon}\mu_\Delta$, where ϵ is a configurable parameter. The multiplicative factor, $\frac{1+\epsilon-\mu_{topic_p,e_i,epoch_k}}{1+\epsilon}$, has a range of $[\frac{\epsilon}{\epsilon+1}, 1]$ and approaches 1 when expertise approaches zero and approaches $\frac{\epsilon}{\epsilon+1}$ when expertise approaches one. I.e., in this setting, stronger experts gain (or lose) expertise at a slower rate than weaker experts. In all our experiments, we set ϵ to 0.05. Our experimental results on expertise-level dependent drift are summarized in Fig. 5. We consider two cases where μ_Δ is drawn from $\mathcal{D}_{\mathcal{T}}$ and $\mathcal{D}_{e,\mathcal{T}}$. Our results indicate that `Hybrid` still outperformed the challengers, however, the performance gap was narrower. This is because as experts improve, the rate of change slows down allowing weaker algorithms to track drift somewhat better than in the previous setting.



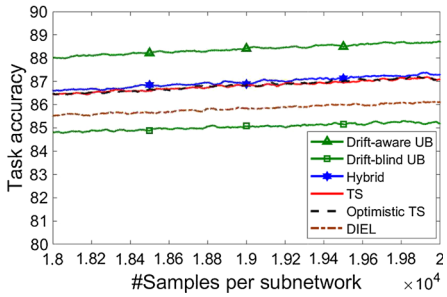
(a) Drift distribution $\mathcal{D}_{\mathcal{T}}$



(b) Steady-state performance closeup: zooming in to 5(a) upper right portion

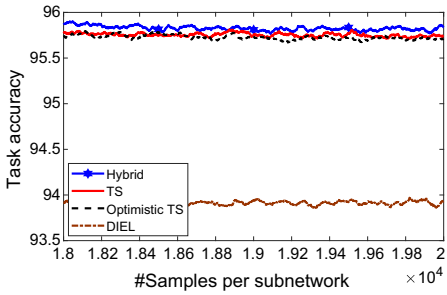


(c) Drift distribution $\mathcal{D}_{e,\mathcal{T}}$

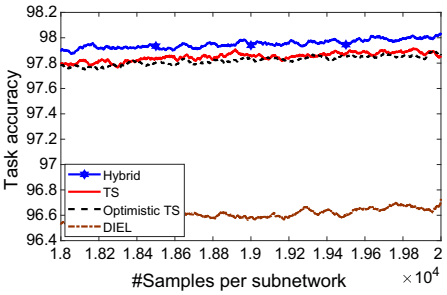


(d) Steady-state performance closeup: zooming in to 5(c) upper right portion

Fig. 5 Expertise-level dependent drift



(a) Drift distribution $\mathcal{D}_{\mathcal{T}}$



(b) Drift distribution $\mathcal{D}_{e,\mathcal{T}}$

Fig. 6 Multi-hop referral with $Q = 3$

7.3 Higher query budget and different network distributions

7.3.1 Higher query budget

So far, we conducted all our experiments with the per-instance query budget Q set to 2. For our next set of experiments, we relax this assumption and consider bounded multi-hop referrals, setting $Q = 3$. In case of multi-hop referrals, suppose an instance is first received by expert A who redirects it to B, B redirects it to C who eventually solves it. C will inform

B of the solution who in turn will inform A. So A will learn B solved the instance and will assign a reward 1 to B while B will learn C (the actual solver) solved the instance and will assign a reward 1 to C. Multi-hop referrals introduce additional non-stationarity in expertise in a sense that a weak expert can find a strong colleague in a later part of the simulation which effectively changes her ability to solve problems, i.e. her observable expertise (even if she solves few if any problems).

We focus on two drift distributions $\mathcal{D}_{\mathcal{T}}$ and $\mathcal{D}_{e,\mathcal{T}}$. Figure 6 summarizes the performance of `Hybrid` and the challengers. In order to obtain a better visualization of the results, we exclude the upper bounds and focus only on the steady-state performance. As shown in Fig. 6, `Hybrid` marginally outperformed the challengers. Understandably, with a higher per-instance query budget, the overall performance of all algorithms improved. However, due to additional non-stationarity in expertise, the performance gap between `Hybrid` and `TS` variants became narrower.

7.3.2 Different network topologies

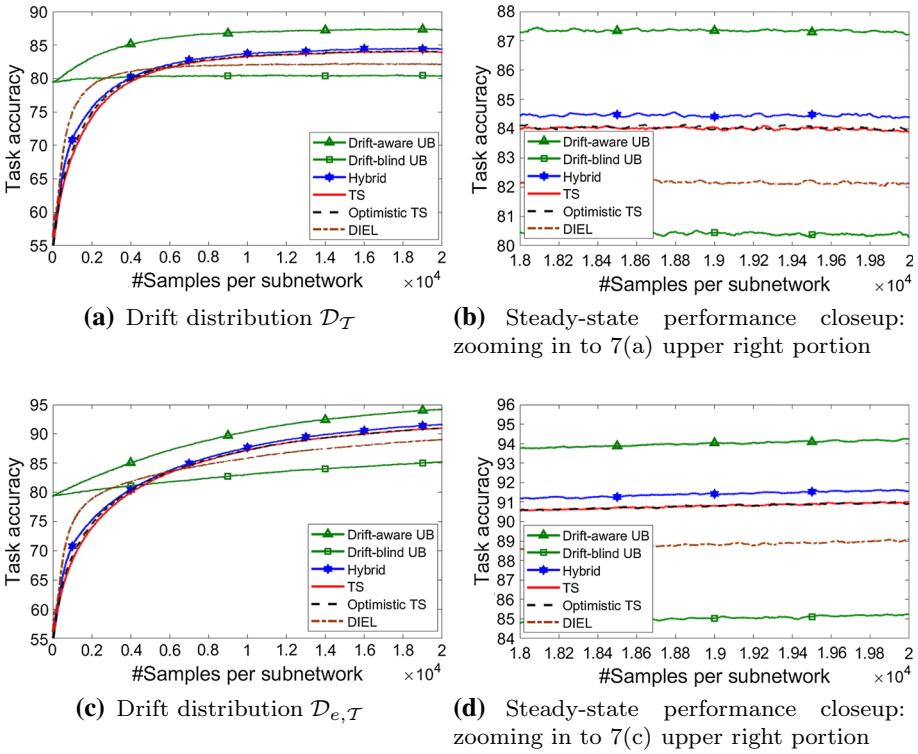
Intuitively, the distributed nature of referral-learning with each expert only focusing on estimating the expertise of other experts in her *subnetwork* lends implicit robustness across network topologies. In fact, previous work [44] has reported that the relative ordering of referral-learning algorithms' performance in static expertise setting is robust to a wide range of network topologies generated using well-known random graph generators [10,33,62]. In Fig. 7, we report the results on small-world graphs known to model collaborative networks. In this set of experiments, we assume the drift distributions (which only concern expert–topic pairs) remain the same as previous, only the underlying network topology has changed. Figure 7 shows that across different network topologies, `Hybrid` remains the best-performing algorithm. In the interest of saving space, we omit qualitatively similar results for other two network distributions.

7.3.3 Asymmetric referrals

In an expert referral network, the standard assumption is referral links are bi-directional [41, 44,45]. However the most renowned experts may be referred to by many more than those to whom they refer. Hence the referral links may be uni-directional with the renowned experts having more incoming links than outgoing links.

In this section, we construct a new data set that obeys the above condition. Let a uni-directional referral link be denoted by $\langle e_i \rightarrow e_j \rangle$ indicating that only e_i can refer to e_j but not the other way round. We next describe the steps with which we change some of the existing bi-directional referral links to uni-direcitonal links (*sparsify* step) and introduce additional new uni-directional referral links (*densify* step).

For a given scenario, we construct a set of renowned experts (denoted as $\mathcal{RE}$) consisting of the top 3 experts on each topic. In our *sparsify* step, for a renowned expert on topic T, $e^T_{renowned}$, we remove outgoing connections to the three weakest experts on topic T (denoted as e^T_{weak}) in her subnetwork such that the weakest experts do not belong to $\mathcal{RE}$, i.e., we modify $\langle e^T_{weak}, e^T_{renowned} \rangle$ to $\langle e^T_{weak} \rightarrow e^T_{renowned} \rangle$ such that $e^T_{weak} \notin \mathcal{RE}$ and $e^T_{weak} \in subnetwork(e^T_{renowned})$. We stop at the boundary condition where modifying a link would eliminate all outgoing links from $e^T_{renowned}$ (e.g., if a renowned expert has only three or less bi-directional links to start with). The intuition for granting experts belonging to $\mathcal{RE}$



(a) Drift distribution $\mathcal{D}_{\mathcal{T}}$

(b) Steady-state performance closeup: zooming in to 7(a) upper right portion

(c) Drift distribution $\mathcal{D}_{e,\mathcal{T}}$

(d) Steady-state performance closeup: zooming in to 7(c) upper right portion

Fig. 7 Performance on referral networks constructed using small world graph generator [62]

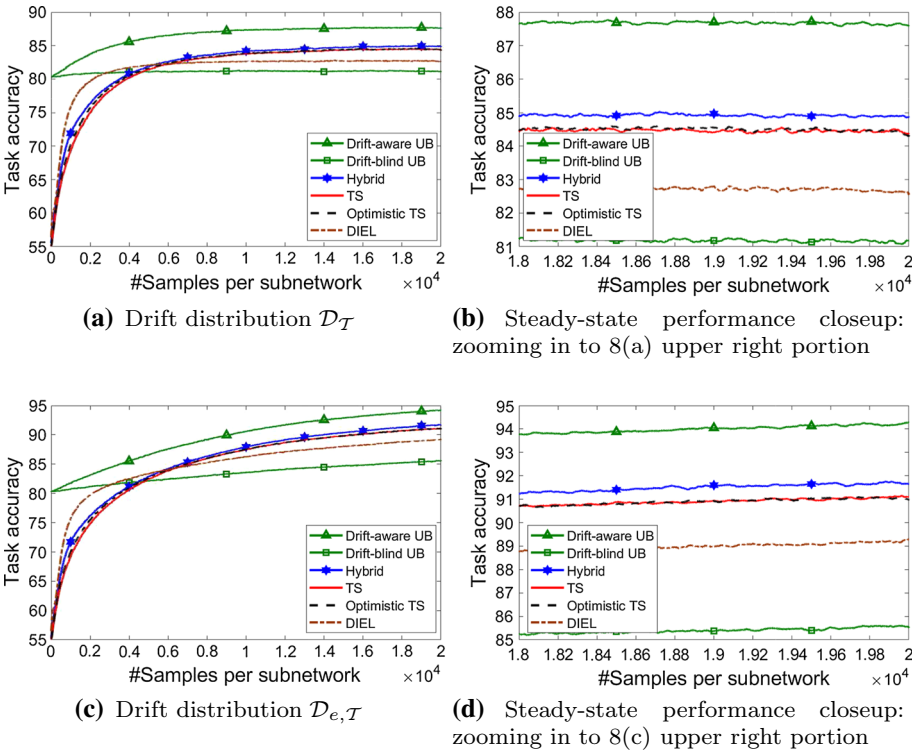
sparsification immunity was that the expert may simply be known because of other expertise (albeit on a different topic).

Our *densify* step is rather simple and guided by the intuition that a renowned expert may be known by other experts whom she may not know. In this step, for a renowned expert $e^T_{renowned}$, we randomly select three experts without any existing incoming referral links to $e^T_{renowned}$ and create $\langle e_{random} \rightarrow e^T_{renowned} \rangle$, an incoming link from the randomly chosen expert e_{random} to the renowned expert. Note that, for every bi-directional link modified to a uni-directional link, we create an additional uni-directional link that did not exist before. Hence, the overall number of links in the referral network remains roughly the same, though we may not modify some of the bi-direcitonal links during the *sparsify* step due to the boundary condition. As shown in Fig. 8, on our new data set, `Hybrid` still leads the pack outperforming `DIEL` and `TS` variants.

7.4 Component analysis

In this subsection, we perform a thorough component analysis of `Hybrid`. The key research questions we ask are the following:

- How critical is the variance reset?
- How sensitive is `Hybrid` to different choices of values for w?
- Can `DIEL` with variance reset alone prove sufficient to track drift?



(a) Drift distribution $\mathcal{D}_{\mathcal{T}}$

(b) Steady-state performance closeup: zooming in to 8(a) upper right portion

(c) Drift distribution $\mathcal{D}_{e,\mathcal{T}}$

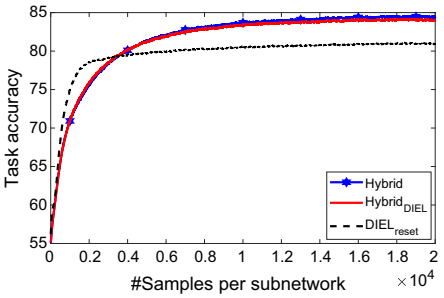
(d) Steady-state performance closeup: zooming in to 8(c) upper right portion

Fig. 8 Performance on networks with asymmetric referral links

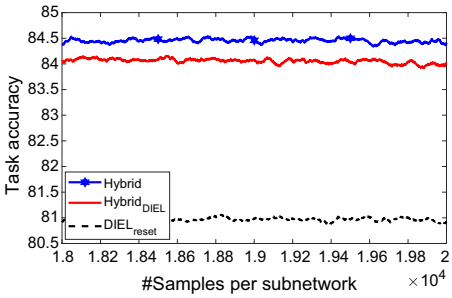
– Can Hybrid's design be simplified without compromising on performance?
– and finally, the most important, can $\texttt{DIEL}_{reset}$ be combined with other Bayesian approaches?

In the design of Hybrid, the novelties are twofold: first, we use a new variant of DIEL with variance reset, second, we randomize between a frequentist ($\texttt{DIEL}_{reset}$) and a Bayesian TS approach. We focus on the criticality of both aspects on the success of Hybrid. In Fig. 9, we compare three algorithms: $\texttt{Hybrid}_{\texttt{DIEL}_{reset}}$, $\texttt{Hybrid}_{\texttt{DIEL}}$ and $\texttt{DIEL}_{reset}$. $\texttt{Hybrid}_{\texttt{DIEL}_{reset}}$ is basically Hybrid, our proposed new algorithm. $\texttt{Hybrid}_{\texttt{DIEL}}$ randomizes between TS and DIEL, but does not use the variance reset mechanism first proposed in this paper. As shown in Fig. 9, Hybrid outperforms both $\texttt{DIEL}_{reset}$ and $\texttt{Hybrid}_{\texttt{DIEL}}$ indicating that neither the variance reset alone nor the randomization between between TS and DIEL without variance reset is sufficient to obtain drift-tracking performance similar to Hybrid. Hence, our current design of Hybrid does not have any component that can be discarded for the sake of further simplicity without significantly losing performance.

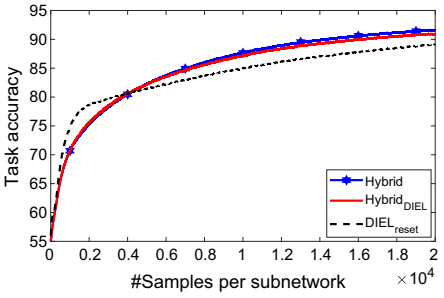
Hybrid has a configurable parameter w that specifies the size of the referral window. Apart from 100, the value w was set to in all our experiments, we considered three additional choices: 50, 100, and square of the *subnetwork* size of a given expert (i,e., $|subnetwork(e)|^2$, where the size of the *subnetwork* of expert e is denoted by $|subnetwork(e)|$). In Fig. 10, we report the steady-state performance of Hybrid with different parameter configurations on $\mathcal{D}_{\mathcal{T}}$ and $\mathcal{D}_{e,\mathcal{T}}$. Our results indicate that over a reasonable set of values for w, different configurations of Hybrid's performance is practically indistinguishable. Additionally, set-



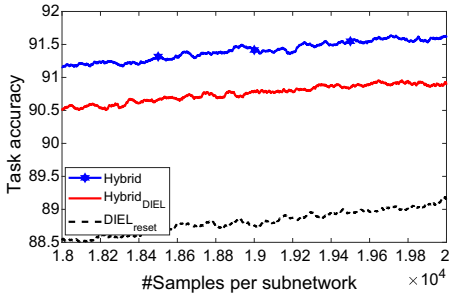
(a) Drift distribution $\mathcal{D}_{\mathcal{T}}$



(b) Steady-state performance closeup: zooming in to 10(a) upper right portion

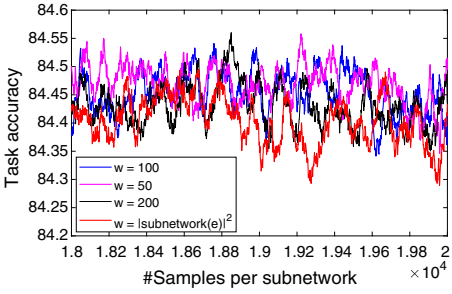


(c) Drift distribution $\mathcal{D}_{e,\mathcal{T}}$

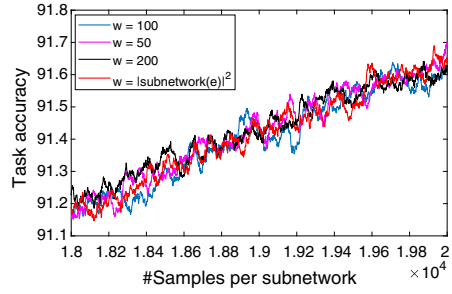


(d) Steady-state performance closeup: zooming in to 10(c) upper right portion

Fig. 9 Components of `Hybrid`



(a) Drift distribution $\mathcal{D}_{\mathcal{T}}$

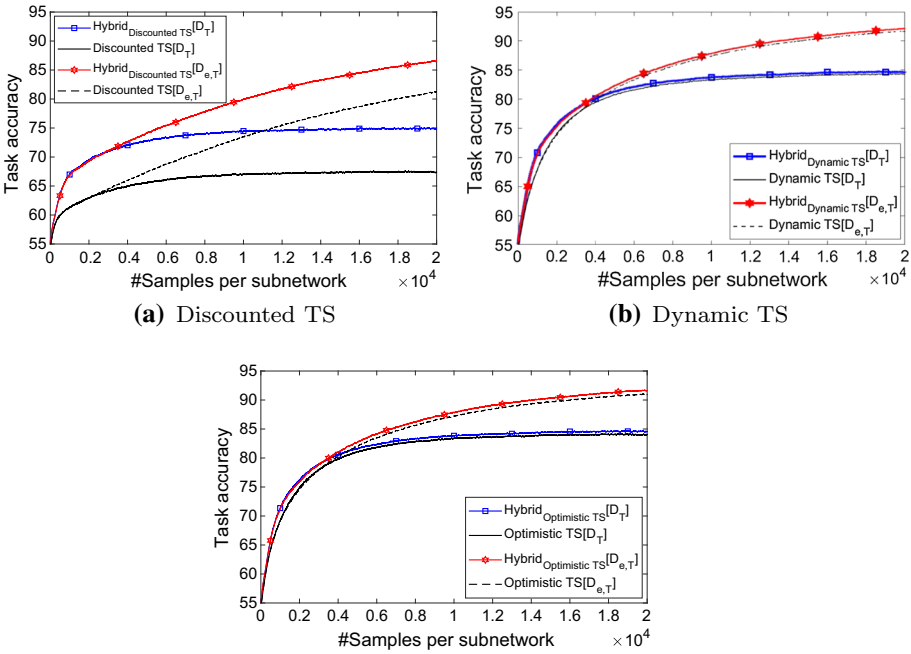


(b) Drift distribution $\mathcal{D}_{e,\mathcal{T}}$

Fig. 10 Sensitivity of the w parameter. The steady-state performance of `Hybrid` with different parameter configurations is practically indistinguishable

ting w w.r.t. the *subnetwork* size did not affect its performance either. Hence, `Hybrid`'s performance is not highly sensitive to the choice of w and it is possible to empirically set the window size for different sizes of data sets.

Finally, we pursue a deeper design question: can we combine $\texttt{DIEL}_{reset}$ with other Bayesian approaches and obtain similar performance boost? We considered three `Thompson Sampling` variants for this design analysis: `Optimistic TS` [52], an algorithm already included as one of our challengers, `Discounted TS` [54] a recently-proposed `Thompson Sampling` variant for non-stationary bandits, and `Dynamic TS` [28], another `TS` variant



(a) Discounted TS **(b)** Dynamic TS

(c) Optimistic TS

Fig. 11 Combining `Hybrid` with different `Thompson Sampling` variants. The drift distribution is indicated within square brackets

for non-stationary distributions. As we already described in Sect. 5.1, `Optimistic TS` is parameter-less. However, both `Discounted TS` (γ) and `Dynamic TS` (C) has one configurable parameter. We set C to 100, the same as the size of the referral-window in `Hybrid`. We set γ to 0.95.[2]

As shown in Fig. 11, we found that `Hybrid`'s design is fairly general, in a sense that other `TS` variant can be plugged into and the resulting algorithm would track drift better than its individual `TS` components. In fact, with $\texttt{Hybrid}_{\texttt{Dynamic TS}}$, we obtained slightly better performance than `Hybrid`. Our results lead us to the following observation: different `MAB` algorithms have different exploitation-exploration mechanisms, combining these strategies in an interleaved manner with one algorithm benefiting from another algorithm's exploration (or exploitation), may lead to improved performance, and that improvement can be significant and robust as shown for `Hybrid`.

8 Conclusions and future work

Learning to refer is a recent Active Learning setting where experts can redirect difficult tasks they cannot solve to other connected experts. In this work, we introduced the notion of

[2] [54] reports 0.8 as the optimal value of γ for slowly moving distributions. However, in our experiments, we obtained better performance for both `Discounted TS` and $\texttt{Hybrid}_{\texttt{Discounted TS}}$ when γ was set to 0.95. We have not performed extensive parameter tuning for the new `TS` variants and chose values that seemed reasonable. We admit that with parameter tuning, it may be possible to squeeze further performance boost out of $\texttt{Hybrid}_{\texttt{Dynamic TS}}$, but our primary goal was to test `Hybrid`'s design compatibility with other `TS` variants.

time-varying expertise in referral networks, an important practical factor not considered in the literature. Our results indicate that DIEL, the state-of-the-art referral learning algorithm on referral networks without time-varying expertise, is vulnerable to expertise drift. Hence, we proposed a novel combination of Thompson Sampling and DIEL$_{reset}$ which outperformed DIEL on different types of drift conditions that include topic-dependent drifts and expertise-level-dependent drifts. Moreover, we have shown that DIEL$_{reset}$ can be combined with other Bayesian approaches for similar drift-tracking improvement.

We now revisit the research questions and summarize our findings.

1. **Are existing high-performance algorithms vulnerable to time-varying expertise? If yes, how to track expertise drift in referral networks?**
   Yes, they are vulnerable. In fact, DIEL's (the known state-of-the-art on stationary setting) drift tracking performance was worse than existing Thompson Sampling variants in presence of biased drift.
2. **Previous literature provides experimental evidence DIEL's early learning advantage and Thompson Sampling's (TS) competitive performance due to Bayesian exploration. Can DIEL or a variant of DIEL be effectively combined with TS variants for superior drift-tracking?**
   Yes, we answered this question constructively by proposing and testing Hybrid, a new combination of TS and DIEL$_{reset}$. Across a wide range of drift settings, Hybrid consistently outperformed all challenger algorithms we considered.
3. **Is the newly-proposed Hybrid algorithm robust to topic-dependent drifts and expertise-level-dependent drifts?**
   Yes. Additionally, Hybrid is robust to a drift setting where drift behavior can vary across expert-and-topic pairs concurrently.
4. **How crucial is the variance reset for improved performance?**
   Our empirical evaluations suggested all components of Hybrid were necessary for its superior performance. Not only is variance reset necessary, but also effectively randomizing between DIEL$_{reset}$ and a Bayesian TS algorithm was critical for its performance.
5. **Can DIEL with variance reset be combined with other Bayesian approaches for improved performance?**
   Yes. We conducted experiments with three additional TS variants and found that the resulting Hybrid algorithms outperformed their respective TS component alone. In fact, Hybrid$_{\text{Dynamic TS}}$, combination of DIEL$_{reset}$ and Dynamic Thompson Sampling, slightly outperformed Hybrid.

Our work can be extended in multiple ways, including the following:

1. **Biasing Hybrid to the more successful component:** Our design for randomization between two component algorithms is rather simple. Instead of randomizing between two components with equal probability, a performance-dependent guided selection approach or meta-learning may lead to an additional performance boost. A deeper exploration of this idea can be an interesting future line of research.
2. **Adaptive mechanism to set the w parameter of** DIEL$_{reset}$: We have shown that the performance of Hybrid is not overly sensitive to the choice of w. Moreover, the parameter can be configured w.r.t. the individual *subnetwork* size. However, a more principled approach to set w motivated by some recent progress in adaptively setting parameter values could prove fruitful.
3. **Theoretical analysis of Hybrid:** Even in the stationary expertise setting, the analysis of Thompson Sampling, one of the two components of Hybrid was a long-standing research problem that remained unsolved for decades. To the best of our knowledge,

DIEL does not have known finite-time regret bound in the static setting either. Hence, theoretical analysis of Hybrid in a time-varying setting is a doubly-challenging research problem. However, considering the practical performance, a thorough theoretical analysis of Hybrid would be a worthy research objective.

4. **Relaxing the crisp boundary and independence assumptions of topics:** In this work and also in previous research on referral networks, the typical assumption is that topics have clear boundaries and expertise in one particular topic is independent of expertise in other topics. These assumptions do not strictly hold in the real-world; skilled experts on a particular topic are likely to succeed in related topics and instances may belong to multiple overlapping topics (e.g., a multi-task learning topic can also be a continual learning topic). Devising drift-tracking algorithms in a learning setting where topics have overlapping boundaries and expertise on similar topics are correlated can be an interesting future research direction.

References

1. Abdallah, S., & Kaisers, M. (2016). Addressing environment non-stationarity by repeating Q-learning updates. *The Journal of Machine Learning Research*, *17*(1), 1582–1612.
2. Agrawal, R. (1995). Sample mean based index policies with O (log n) regret for the multi-armed bandit problem. *Advances in Applied Probability*, *27*, 1054–1078.
3. Agrawal, S., & Goyal, N. (2012). Analysis of Thompson sampling for the multi-armed bandit problem. In *COLT* (pp. 39–1).
4. Akakpo, N. (2008). *Detecting change-points in a discrete distribution via model selection*. arXiv preprint arXiv:0801.0970.
5. Allesiardo, R., & Féraud, R. (2015). Exp3 with drift detection for the switching bandit problem. In *IEEE international conference on data science and advanced analytics (DSAA)* (pp. 1–7). IEEE.
6. Audibert, J. Y., Munos, R., & Szepesvári, C. (2007). Tuning bandit algorithms in stochastic environments. In *International conference on algorithmic learning theory* (pp. 150–165). Springer.
7. Auer, P., Cesa-Bianchi, N., & Fischer, P. (2002). Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, *47*(2–3), 235–256.
8. Auer, P., Cesa-Bianchi, N., Freund, Y., & Schapire, R. E. (2002). The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing*, *32*(1), 48–77.
9. Babaioff, M., Sharma, Y., & Slivkins, A. (2014). Characterizing truthful multi-armed bandit mechanisms. *SIAM Journal on Computing*, *43*(1), 194–230.
10. Barabási, A. L., & Albert, R. (1999). Emergence of scaling in random networks. *Science*, *286*(5439), 509–512.
11. Baram, Y., Yaniv, R. E., & Luz, K. (2004). Online choice of active learning algorithms. *Journal of Machine Learning Research*, *5*(Mar), 255–291.
12. Berry, D. A., & Fristedt, B. (1985). *Bandit problems: Sequential allocation of experiments (monographs on statistics and applied probability)* (Vol. 12). Berlin: Springer.
13. Bertsimas, D., & Niño-Mora, J. (2000). Restless bandits, linear programming relaxations, and a primal–dual index heuristic. *Operations Research*, *48*(1), 80–90.
14. Bouneffouf, D., & Feraud, R. (2016). Multi-armed bandit problem with known trend. *Neurocomputing*, *205*, 16–21.
15. Bowling, M., & Veloso, M. (2001). Rational and convergent learning in stochastic games. In *International joint conference on artificial intelligence* (Vol. 17, pp. 1021–1026). Lawrence Erlbaum Associates Ltd.
16. Burtini, G., Loeppky, J., & Lawrence, R. (2015). *A survey of online experiment design with the stochastic multi-armed bandit*. arXiv preprint arXiv:1510.00757.
17. Cesa-Bianchi, N., & Lugosi, G. (2006). *Prediction, learning, and games*. Cambridge: Cambridge University Press.
18. Chapelle, O., & Li, L. (2011). An empirical evaluation of thompson sampling. In J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira, K. Q. Weinberger (Eds.), *Advances in neural information processing systems* (pp. 2249–2257).

19. Donmez, P., Carbonell, J., & Schneider, J. (2010). A probabilistic framework to learn from multiple annotators with time-varying accuracy. In *Proceedings of the 2010 SIAM international conference on data mining* (pp. 826–837). SIAM.
20. Donmez, P., Carbonell, J. G., & Bennett, P. N. (2007) Dual strategy active learning. In *Machine learning ECML* (pp. 116–127).
21. Donmez, P., Carbonell, J. G., & Schneider, J. (2009). Efficiently learning the accuracy of labeling sources for selective sampling. In *Proceedings of the 15th ACM international conference on knowledge discovery and data mining* (p. 259).
22. Dragoni, N. (2006). Fault tolerant knowledge level inter-agent communication in open multi-agent systems. *AI Communications*, *19*(4), 385–387.
23. Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M., & Bouchachia, A. (2014). A survey on concept drift adaptation. *ACM Computing Surveys (CSUR)*, *46*(4), 44.
24. Garivier, A., & Moulines, E. (2008). *On upper-confidence bound policies for non-stationary bandit problems*. arXiv preprint arXiv:0805.3415.
25. Gittins, J. C., & Jones, D. M. (1974). A dynamic allocation indices for the sequential design of experiments. In J. Gani (Ed.), *Progress in statistics, European meeting of statisticians* (Vol. 1, pp. 241–266).
26. Gorner, J. M. (2011). *Advisor networks and referrals for improved trust modelling in multi-agent systems*. Master's thesis, University of Waterloo
27. Graepel, T., Candela, J. Q., Borchert, T., & Herbrich, R. (2010). 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).
28. Granmo, O. C. (2010). Solving two-armed Bernoulli bandit problems using a Bayesian learning automaton. *International Journal of Intelligent Computing and Cybernetics*, *3*(2), 207–234.
29. Guha, S., Munagala, K., & Shi, P. (2010). Approximation algorithms for restless bandit problems. *Journal of the ACM (JACM)*, *58*(1), 3.
30. Gupta, N., Granmo, O. C., & Agrawala, A. (2011). Thompson sampling for dynamic multi-armed bandits. In *10th International conference on machine learning and applications and workshops (ICMLA)* (Vol. 1, pp. 484–489). IEEE.
31. Hartland, C., Gelly, S., Baskiotis, N., Teytaud, O., & Sebag, M. (2006). Multi-armed bandit, dynamic environments and meta-bandits. In *NIPS-2006 workshop, Online trading between exploration and exploitation*. Whistler, Canada.
32. Hasselt, H. V. (2010). Double Q-learning. In *Advances in neural information processing systems* (pp. 2613–2621).
33. Holme, P., & Kim, B. J. (2002). Growing scale-free networks with tunable clustering. *Physical Review E*, *65*(2), 026,107.
34. Huang, L., Joseph, A. D., Nelson, B., Rubinstein, B. I., & Tygar, J. (2011). Adversarial machine learning. In *Proceedings of the 4th ACM workshop on security and artificial intelligence* (pp. 43–58). ACM.
35. Kaelbling, L. P. (1993). *Learning in embedded systems*. Cambridge: MIT Press.
36. Kaisers, M., & Tuyls, K. (2010). Frequency adjusted multi-agent Q-learning. In Proceedings of the 9th international conference on autonomous agents and multiagent systems (Vol. 1, pp. 309–316). International Foundation for Autonomous Agents and Multiagent Systems.
37. Kaufmann, E., Korda, N., & Munos, R. (2012). Thompson sampling: An asymptotically optimal finite-time analysis. In *International conference on algorithmic learning theory* (pp. 199–213). Springer.
38. Kautz, H., Selman, B., & Milewski, A. (1996). *Agent amplified communication* (pp. 3–9).
39. KhudaBukhsh, A. R., & Carbonell, J. G. (2018). Expertise drift in referral networks. In *Proceedings of the 17th international conference on autonomous agents and multiagent systems* (pp. 425–433). International Foundation for Autonomous Agents and Multiagent Systems.
40. KhudaBukhsh, A. R., Carbonell, J. G., & Jansen, P. J. (2016). Proactive-DIEL in evolving referral networks. In *European conference on multi-agent systems* (pp. 148–156). Springer.
41. KhudaBukhsh, A. R., Carbonell, J. G., & Jansen, P. J. (2016). Proactive skill posting in referral networks. In *Australasian joint conference on artificial intelligence* (pp. 585–596). Springer.
42. KhudaBukhsh, A. R., Carbonell, J. G., & Jansen, P. J. (2017). Incentive compatible proactive skill posting in referral networks. In *European conference on multi-agent systems*. Springer.
43. KhudaBukhsh, A. R., Carbonell, J. G., & Jansen, P. J. (2017). Robust learning in expert networks: A comparative analysis. In *International symposium on methodologies for intelligent systems (ISMIS)* (pp. 292–301). Springer.
44. KhudaBukhsh, A. R., Carbonell, J. G., & Jansen, P. J. (2018). Robust learning in expert networks: A comparative analysis. *Journal of Intelligent Information Systems*, *51*(2), 207–234.
45. KhudaBukhsh, A. R., Jansen, P. J., & Carbonell, J. G. (2016). Distributed learning in expert referral networks. In *European conference on artificial intelligence (ECAI)* (pp. 1620–1621).

46. Lai, T. L. (2001). *Sequential analysis*. New York: Wiley Online Library.
47. Lai, T. L., & Robbins, H. (1985). Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, *6*(1), 4–22.
48. Langford, J., Strehl, A., & Wortman, J. (2008). Exploration scavenging. In *Proceedings of the 25th international conference on Machine learning* (pp. 528–535). ACM.
49. Levine, N., Crammer, K., & Mannor, S. (2017). Rotting bandits. In *Advances in neural information processing systems* (pp. 3074–3083).
50. Liu, K., & Zhao, Q. (2010). Indexability of restless bandit problems and optimality of whittle index for dynamic multichannel access. *IEEE Transactions on Information Theory*, *56*(11), 5547–5567.
51. Lu, X., Adams, N., & Kantas, N. (2017). *On adaptive estimation for dynamic Bernoulli bandits*. arXiv preprint arXiv:1712.03134.
52. May, B. C., Korda, N., Lee, A., & Leslie, D. S. (2012). Optimistic Bayesian sampling in contextual-bandit problems. *Journal of Machine Learning Research*, *13*(Jun), 2069–2106.
53. Noda, I. (2009). Recursive adaptation of stepsize parameter for non-stationary environments. In *ALA* (pp. 74–90). Springer.
54. Raj, V., & Kalyani, S. (2017). *Taming non-stationary bandits: A Bayesian approach*. arXiv preprint arXiv:1707.09727.
55. Shivaswamy, P. K., & Joachims, T. (2012). Multi-armed bandit problems with history. In N. D. Lawrence & M. Girolami (Eds.), *International Conference on Artificial Intelligence and Statistics* (pp. 1046–1054).
56. Silva, B. C. D., Basso, E. W., Bazzan, A., & Engel, P. M. (2006). Dealing with non-stationary environments using context detection. In *Proceedings of the 23rd international conference on machine learning* (pp. 217–224). ACM.
57. Slivkins, A., & Upfal, E. (2008). Adapting to a changing environment: The Brownian restless bandits. In *COLT* (pp. 343–354).
58. Tekin, C., & Liu, M. (2012). Online learning of rested and restless bandits. *IEEE Transactions on Information Theory*, *58*(8), 5588–5611.
59. Thompson, W. R. (1933). On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, *25*(3/4), 285–294.
60. Tsymbal, A. (2004). The problem of concept drift: Definitions and related work. *Computer Science Department, Trinity College Dublin*, *106*(2), 58.
61. Watkins, C. J., & Dayan, P. (1992). Q-learning. *Machine Learning*, *8*(3–4), 279–292.
62. Watts, D. J., & Strogatz, S. H. (1998). Collective dynamics of 'small-world' networks. *Nature*, *393*(6684), 440–442.
63. Weber, R. R., & Weiss, G. (1990). On an index policy for restless bandits. *Journal of Applied Probability*, *27*(3), 637–648.
64. Wei, W., Li, C. M., & Zhang, H. (2008). A switching criterion for intensification, and diversification in local search for SAT. *Journal on Satisfiability, Boolean Modeling and Computation*, *4*, 219–237.
65. Whittle, P. (1988). Restless bandits: Activity allocation in a changing world. *Journal of Applied Probability*, *25*(A), 287–298.
66. Wiering, M., & Schmidhuber, J. (1998). Efficient model-based exploration. In *Proceedings of the fifth international conference on simulation of adaptive behavior (SAB'98)* (pp. 223–228).
67. Yolum, P., & Singh, M. P. (2003). Dynamic communities in referral networks. *Web Intelligence and Agent Systems*, *1*(2), 105–116.
68. Yolum, P., & Singh, M. P. (2003). Emergent properties of referral systems. In *Proceedings of the second international joint conference on autonomous agents and multiagent systems* (pp. 592–599). ACM.
69. Yu, B. (2002). *Emergence and evolution of agent-based referral networks*. Ph.D. thesis, North Carolina State University.
70. Yu, B., Venkatraman, M., & Singh, M. P. (2003). An adaptive social network for information access: Theoretical and experimental results. *Applied Artificial Intelligence*, *17*, 21–38.
71. Yu, J. Y., & Mannor, S. (2009). Piecewise-stationary bandit problems with side observations. In *Proceedings of the 26th annual international conference on machine learning* (pp. 1177–1184). ACM.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.