1 Background, Problem and objectives

In many scientific and practical situations the signs or characteristics presented by a case (a medical problem, intelligence information, a traveler at a border or airport) are imperfectly related to the condition of interest (a disease, a threat, a smuggler, etc). A prompt decision must be made as to how to deal with the case at hand. In particular, we learn nothing about the true situation without some additional and expensive processing (diagnostic surgery, a detailed review by an analyst, a search an interview). When we incur the expense of collecting this information, we learn something not only about a particular case, but also about how well a particular set of attributes correlates with a positive outcome (the patient has a disease, the cargo represents a threat). The decision to collect this information, then, must balance the cost (and, Probabilistically, the value) of having the information now, and the value of the improving our ability to make better predictions in the future, even if we are "wrong" about this particular case.

These problems are important. And there may be thousands of patients, hundreds of thousands of containers and millions of documents. It is not possible to determine whether every particular case is actionable. Yet, misdiagnosing a single case (overlooking cancer in a patient, bypassing a container carrying a nuclear bomb, overlooking the critical document with instructions to terrorists) can have devastating consequences.

Whatever the current state of our rule or algorithm, eachcase has some estimated immediate value, and some potential to improve our future decisions. This is a key challenge for robust intelligence: reasoning about uncertain situations, in the presence of costs for learning, and with evanescent opportunities to learn. We will address this using dynamic programming problem. Specifically, the "state of the system" is its state of knowledge relating the attributes of the event (a patient, cargo coming through a port, a website) to the likelihood of a positive outcome (the patient has a disease, the cargo is carrying dangerous cargo, the website has valuable information). Of course, the dynamic program in its full form is computationally intractable.

We seek policies for learning rules. That is, we seek effective heuristics, or approximate algorithms. We also will rigorously assess the *distribution of the effectiveness* of any specific heuristic. A related problem is the "multi-armed bandit" problem, a rare case that has been solved optimally in a practical way, using the Gittins indices. The difficulty of computing Gittins indices has spawned a number of heuristics. But this literature has largely ignored a critical feature of real problems. Specifically, in real problems, examining a case with specific attributes will reveal something about cases with nearby or related attributes. This fact is fundamental to the scientific method. When we conclude that a high body-mass index is associated with certain health problems, we explicitly acknowledge that what is true for a specific value of this variable should be approximately true for nearby values. To put it in somewhat more mathematical terms, the property of interest is taken to be a smoothly varying function of the observable information, features or labels.

The specific **objectives of the proposed research** are: (a) to define rigorous models of the relation between features and the properties of interest (b) to solve these models, in the cases where features are uncorrelated (c) to solve them when features are correlated (d) to define and assess heuristics, with reference to these models (e) to extend the models to deal with changing relations between features and the underlying "value" (f) to develop models of the range of value that will be achieved, by applying these heuristics to the model problems (g) to assess and validate all of this work using the TREC Adaptive Filtering collections and tasks. The products of this research will be algorithms and heuristics, experiments run using them on the TREC materials, publications and

conference reports in several relevant venues.

2 Previous related work

This work has three foundations. One is analytical, and draws on prior work in: approximate solution of dynamic programming problems; analysis of change point detection, using Markov models; and assessment of the performance of detection systems, using ROC analysis, and specific models of the behavior of algorithms. The second foundation is a large body of labeled and *evaluated* material, which will serve as testing ground for all the techniques to be developed. These are the TREC materials, as developed and adapted for studying the Adaptive Filtering problem. The third, is feature engineering, specific to the problems of dealing with text materials. We will draw on the large array of feature engineering techniques that have been developed over seven years of TREC effort on this specific problem.

2.1 Prior work on optimal information collection

The problem of optimally collecting information is an old one, but progress has generally been made in narrow problem classes. Early work on optimal decisions in a statistical setting generally addressed problems such as sequential hypothesis testing (see Bechhofer et al. (1968) and DeGroot (1970) for reviews), ranking and selection (Goldsman & Nelson (1994)) and sequential design of experiments (Bechhofer et al. (1995)). Cohn et al. (1996) provides a method for optimally collecting information in a machine learning setting. The classic bandit problems for making choices while learning from the experience of making each choice saw a resurgence in interest with the discovery of Gittins indices (Gittins & Jones (1974), Gittins (1989)). There have been many attempts to generalize the basic model, but the results remain quite limited.

The simulation community has addressed this problem in the context of choosing parameters to control a simulation (see Fu (2002) and Swisher et al. (2003)). Each particular parameter setting usually produces a noisy response, and the time required to measure the performance of a single parameter setting may be significant. The literature on optimal computing budget allocation (OCBA) has focused on finding methods for collecting this information as quickly as possible (see Chen et al. (2000), Thorsley & Teneketzis (2007) and He et al. (2007) for examples and references).

Most of the literature on the "exploration vs. exploitation" problem (as it is often referred to) is relatively heuristic in nature (a review of these techniques is given in Powell (2007), Chapter 10). Standard techniques involve mixing exploration vs. exploitation in a fixed, pre-specified ratio, a declining ratio (epsilon-greedy optimization), and Boltzmann exploration (which explores decisions with a weight that is proportional to the attractiveness of a decision). Other techniques can be viewed as heuristic variations of Gittins indices, including interval exploration (Kaelbling (1993)) and upper confidence bounding (Chang et al. (2007)). Bickel & Smith (2006) illustrate optimal learning in the context of a binary choice model, motivated by an application of drilling oil wells.

Warren: POSSIBLE MISPLACED PARAGRAPH FOLLOWS

Gittins indices can be applied heuristically, and we can use procedures such as the knowledge gradient which has nice theoretical properties but is not generally optimal. Information collection problems can be formulated as dynamic programs, but these cannot be solved to optimality. Duff & Barto (1997) proposes a method for discrete state, discrete action dynamic programs to learn optimally, but this strategy has not been widely adopted. We will investigate the structure of optimal policies using classical theory, and we propose to use approximate dynamic programming to compute near-optimal policies, if only for the purpose of evaluating more convenient heuristics. ADP

has proven successful in a range of resource allocation problems (Topaloglu & Powell (2006), Powell et al. (2004), Powell (2007)), and we propose to use these methods for the problem of managing sensor resources, which exhibits some of the same structural properties (such as concavity).

2.2 Prior Work on Change Point Detection

Learning with costs is more difficult when the relation between labels and the value of examining the cases changes. Precisely this situation arises when there is an emerging threat such as changing political environment, a new bacterial infection, etc. Specifically for the example of monitoring message traffic for emerging threats, when an adversary is discussing a plan of attack, the traffic in relevant documents will increase. Similarly, the number of money transfers to and from rogue states may increase (as there are caps on the amounts that can be wired in one time, large amounts can be transferred only in numerous smaller chunks). When a hacker gains access to a computer, he quickly executes quickly several typical commands (for example, "change directory" and "list" commands as he navigates to the password files.)

Thus, there will be on-periods and off-periods when the arrival rates of documents with interesting labels are high and low, respectively. The start and end points of these periods are not labeled, and it is crucial to be able to identify each on-period quickly and elevate threat-awareness during those periods (for example, by adding temporarily more staff to critical tasks or by alerting officials or even the public.) The problem is complicated by the fact that a feature or label which connotes harmlessness for a period of time, may become significant quite suddenly.

Sequential change-point detection algorithms originate with for fault detection and isolation in industrial processes, target detection and identification in national defense, and in radar and sonar processing, speech and image analysis, and bio-surveillance; see, for example, **?**, **?**, and **?** for an extensive overview. Our own previous work on this topic is summarized below in section 3.3

2.3 Prior work On Adaptive Document Filtering

Adaptive document filtering has been studied in information retrieval. **?** first asked when a searcher with a well-defined task should give up perusing a list of documents, and built a complex Bayesian model involving Gaussian distributions. (**?**) simplified the model using the conjugate Beta and binomial distributions. Another old literature considers selective dissemination of information (SDI) which grew to focus on the construction of complex Boolean queries to describe the interests of specific clients. With the growth of the WWW, interest in Boolean queries declined, although systems such as Verity®are used in government agencies, where complex "standing weighted Boolean queries" help route information to the analysts best qualified to assess it. These systems evolved when all agreed that the end-user was designing a system to meet his or her own specific needs, and was capable of making an implicit decision about how much time should be spent refining the query. It was also presumed that the flow of incoming material (the scientific literature, on the one hand; message traffic of some kind, on the other) did in fact contain enough material of interest, that some of it should be routed to the recipient, fairly often.

In the mid 1990's using the TREC (Text Retrieval Conference), the Intelligence Community posed the important and realistic problem of assessing these assumptions. The first instances of filtering at TREC (TREC4,5) were essentially binary text classification exercises, which later came to be called batch filtering. Adaptive filtering recognized that the true "filtering" problem must not only assign a value to every relevant item routed to the end-user, but must also recognize that there is a cost associated to every item that the end-user must examine Lewis (1997) . The system must budget items sent to the end-user, which are both deliverables, and probes for information about the

end-users specific needs. Thus this is precisely an instance of the problem we address. In TREC5, three metrics were considered, corresponding to three different values of the ratio v/c, $1/2; 1; 3$ where c is the cost, v is the value of a relevant item, as defined below.

In TREC6 (Hull (1998)), more realistically, "Adaptive Filtering," recognized that the system (or the agency) must pay for the information that it uses to build a model. This was formulated as a constant horizon task ($H = 1000$ documents) and the total score for any particular solution was computed without discounting, as $V(P) = \sum_{i=1}^{min(S(R),1000)}[v_i - c]$.

In this equation, R labels the decision rule, and $S(R)$ is the stopping point of the rule. This evaluation formula recognizes that often an optimal policy stops sending documents, having "decided" either that there is not enough value in the stream, or that "it cannot figure out what the end-user wants." The corpus was drawn from FBIS (the Foreign Broadcast Information Service) with 130,000 training documents and 130,000 test documents. Thirty-eight of the topics have extensive (TREC-style) evaluations; others have sparser evaluations based only on the top 100 documents found by the NIST retrieval system. Cost and value were set so that the critical value of the probability of relevance p, was 0.2 or 0.4. The task is so challenging that in the first year of the task, the best system could barely justify its own existence (Hull (1998)). the problem was formulated as: selecting the threshold to balance the immediate cost of presenting an irrelevant document against the information gained by learning about its irrelevance.

Almost all participants filter using a ranked retrieval system, with thresholding of the document score. some converted score to probability using logistic regression. There was more variation in the features used to represent a document. AT & T used terms, phrases (adjacent pairs) and non-adjacent pairs based on weights derived from the Rocchio expansion Singhal (1998). ANU uses terms and phrases from training documents(**?**), and weight terms based on contrasting the probability they occur in the relevant documents and nonrelevant documents. City University ordered terms and adjacent pairs of words, according to a Bayesian model, with iterations to improve the quality of the fit (Walker et al. (1998)). CLARITECH used a Bayesian model in to select the terms, and a Rocchio algorithm in a second pass. Their CLROUTE used a similar method, while CLCOMM used two training sets, retaining only terms identified in both training sets (Zhai et al. (1999)).

In sum, these approaches, all based on linear classifiers or naive Bayes, were aggressive in extending the set of features beyond the traditional word-based vectors, to include: term pairs, k-grams, and selective weighting of document parts. Some reduced the resulting space using a variant of Singular Value Decomposition. All sought, via machine learning or statistical techniques, to improve both the linear classifier and the threshold setting. In a sense, all of theses are more sophisticated extensions of the basic stopping model, which simply asks for a determination that the line of investigation will not pay off. None of this ingenious array of attacks on the problem explicitly considered either the value of learning, or the specified finite horizon. The top six systems were (after correction of some errors in official submission data) indistinguishable by a multiple comparison Neumann-Kulls test.

The specific values of the parameters (v, c) or the threshold probabilities, and the topic tasks were adjusted in succeeding years, and scores became more often positive. The challenge, called the Adaptive Filtering Task, was continued for several years, with slowly improving results (although results at TREC are not, in general, comparable from year to year) until it was decided by the program committee that the problem was not suitable for the TREC venue. Only a small amount of the work ever took into account the dynamic and horizon-driven nature of the work. Specifically, Zhang et al. (2003) Zhang & Callan (2001) considered a single step look-ahead. In the final year, the best performance was achieved by a team from the Chinese Academy of Sciences (Xu et al. (2002)).

A subsequent examination of the task, using homotopic techniques **?**) for studying model parameters **?** determined that the specific parameter settings selected for that entry were extremely close to the optimal values of those parameters for that data set, raising the possibility that the selection of the parameters had been inadvertently guided by exposure to the test data.

For this work, the TREC corpus is representative of the problems of interest, and offers a large body of publicly available data.

3 Plan of Research

Our approach involves these activities: (1) effective approximate solution of the dynamic programming problem (2) exploitation of the smoothness of the value function, to learn from correlation among classes, features or labels (3) change point detection and (4) feature engineering and (5) validation on Adaptive Filtering, by an array of experiments with on features, heuristics, and parameter settings.

There are four major problem classes. The base case covers stationary applications, with stable relationships between the features and uncorrelated features. For documents such models are inadequate. No two documents have exactly the same features. We must consider interdependent or "correlated" features or labels. The extension to change point arises, e.g. when there is an emerging threat. The complete problem therefore considers correlated labels with nonstationary value function.

Below, we sketch how we would handle these three broad model classes. The first is a binary decision problem. Cases arrive sequentially. Each case has a label x_t, based on which we decide between: select further evaluation — discard. The second model problem class presents several cases at once (batched arrivals) and we choose a subset to evaluate. We use this to begin the study of correlated features. Finally, we will add transient or change-point behavior, such as would arise in the presence of emerging threats.

3.1 A Model with Unrelated Labels

We must, here, learn which among a discrete set of labels provides sufficient expected value to be worth sending for further analysis. A label may be a vector of attributes $x = (x_1, x_2, \ldots, x_m)$. Let $p(x)$ be the probability that a case with the label x will turn out, on inspection, to be a positive one. If the value of finding a positive is v and the cost of the detailed examination is c then the condition for being immediately worth sending is, $E(x) = vp(x) - c > 0$. We will deal with time by using a discount factor γ, rather than the artifacts introduced by the use of a finite horizon. However, in developing heuristics and algorithms, it is useful to consider a finite horizon T^{PH}, which makes it possible to formulate the dynamic programming problem.

3.1.1 Optimization model for a binary selection problem

Our most basic model posits a sequence of cases, each of which must either be discarded, or selected for further evaluation *at the time it is presented*. Let t index the cases, and let X_t be the random variable representing the label for the case with index t. We then make a decision Y_t where $Y_t = 1$ if we choose the case for further evaluation and 0 otherwise. Finally, we let $Z_t = 1$ if we find that the case is actionable, and 0 otherwise. As noted, we only observe Z_t if Y_t is 1. The overall process is represented by the sequence of random variables $(X_t, Y_t, Z_t)_{t=1}^{\infty}$. We further suppose examining a document has a cost c and, if the document is interesting, there is a reward $v = 1$ (that is, the value is taken as the numeraire). We suppose an infinite horizon with a discount factor γ. The infinite horizon discounted reward is related to the random variables by the equation:

$$\sum_{t=0}^{\infty} \gamma^t Y_t (Z_t - c).$$

The behavior of a "solution" to this problem will depend on the "underlying reality" of the stream of cases. We give a general probabilistic framework governing X_t and Z_t. Let α be the underlying rate at which interesting documents arrive. Specifically, we assume that the sequence $(Z_t)_{t=1}^{\infty}$ is a sequence of independent Bernoulli random variables with success probability α. The labels that we can see are governed by two (conditional) probability distributions on the space of labels: P_0 and P_1. Together they generate an unconditional distribution: $P_{01} := \alpha P_1 + (1-\alpha)P_0$ to be the unconditional distribution of the X_t. We let $\mathbb{P}$ be the set of the three distributions (P_0, P_1, P_{01}). Now we define a filtration $\{\mathfrak{F}_t\}_{t=0}^{\infty}$ by letting $\mathfrak{F}_t$, $t \geq 1$, be the sigma-algebra generated by $X_1, Y_1, Y_1 Z_1, \ldots, X_t, Y_t, Y_t Z_t$. We will require that Y_{t+1} be measurable with respect to the sigma-algebra generated by $\mathfrak{F}_t$ and X_{t+1}. This sets a mathematical framework, except that we have not specified how any particular decision rule, whether algorithm or heuristic, is to be represented. A *policy* π will be a rule for obtaining Y_{t+1} from $\mathfrak{F}_t$ and X_{t+1}. Mathematically our problem becomes one of finding a policy π that achieves the supremum

$$\sup_{\pi} \mathbb{E}^{\pi} \sum_{t=0}^{\infty} \gamma^t Y_t (-c + Y_t). \tag{1}$$

If α, P_0, and P_1 were known perfectly, and hence P_{01} as well, we could choose Y_t by computing the odds ratio

$$\begin{aligned} \text{Prob}\{Z_t = 1 \mid X_t = x\} &= \text{Prob}\{Z_t = 1\} \, \text{Prob}\{X_t \in dx \mid Z_t = 1\} / \, \text{Prob}\{X_t \in dx\} \\ &= \alpha P_1(dx) / P_{01}(dx) \end{aligned}$$

and then setting $Y_t = I_{\{\text{Prob}\{Z_t=1|X_t\}>c\}}$. However, the distributions P_0, and P_1, and the value of α, are generally unknown. So the rule for choosing the action Y_t must reflect the need to learn them from data. To develop a theoretical framework we formalize the precise way in which α,P_0, and P_1 are unknown using a Bayesian approach. Let us suppose α, P_0, and P_1 are themselves random, under some prior distribution, with α taking values in $[0,1]$ and the P_0 and P_1 taking values in some measurable family of measures. This family of measures may be a parametric family such as the family of normal distributions, or it may be an empirically parameterized family, admitting a much broader class of priors, perhaps at the cost of increased complexity.

3.1.2 Dynamic programming formulation

In principle, we may solve the general problem using dynamic programming. We formulate the solution abstractly, and then specialize to a specific case. Let $\mathbb{S}$ be the space of all possible joint posterior distributions on the random variables α, P_0, and P_1. Let S_0 be the prior distribution on (α, P_0, P_1) under $\mathbb{P}$. Our measurements give a sequence of posterior distributions $(S_n)_{n=0}^{\infty}$ which may be thought of as conditional distributions on (α, P_0, P_1) given $(\mathfrak{F}_n)_{n=0}^{\infty}$. These conditional distributions may be obtained recursively using Bayes rule. It can be shown that the supremum in (1) remains unchanged if we restrict the policy space to stationary policies of the form $\pi : \mathbb{S} \times \mathcal{X} \mapsto \{0,1\}$, where $Y_{t+1} = \pi(S^n, X^{n+1})$ under π. For each such stationary policy let us define the value function for that policy $V^{\pi} : \mathbb{S} \mapsto \mathbb{R}$ as

$$V^{\pi}(S_0) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \pi(S_t, X_{t+1})(-c + Z_{t+1})\right].$$

$V(s) = \sup_{\pi} V^{\pi}(s)$. Then the value function satisfies Bellman's recursion

$$\begin{aligned} V(S_t) &= \sup_{\pi} \mathbb{E}_t \left[\gamma V(S_{t+1}) + \pi(S_t, X_{t+1})(-c + Z_{t+1})\right] \\ &= \mathbb{E}_t \left[\max\{Q(S_t, X_{t+1}, 0), Q(S_t, X_{t+1}, 1)\}\right], \end{aligned}$$

where we define the Q-factor $Q : \mathbb{S} \times \mathcal{X} \times \{0,1\} \mapsto \mathbb{R}$ by

$$Q(S_t, X_{t+1}, y) = \mathbb{E}_t \left[y(-c + Z_{t+1}) + \gamma V(S_{t+1}) \mid X_{t+1}, Y_{t+1} = y \right]$$

Given unlimited computational power, we could compute the value function V as the fixed point of the Bellman recursion using an algorithm such as value iteration. In practice, the size of the state space $\mathbb{S}$ prevents this, or at least makes it very difficult. If the value function can be computed, however, we can use it to find an optimal policy π^* according to the formula $\pi^*(s,x) = I_{\{Q(s,x,1) \geq Q(s,x,0)\}}$. Since the sigma-algebra $\mathfrak{F}_t \vee \sigma(X_{t+1}, Z_{t+1})$ resulting from choosing $Y_{t+1} = 1$ contains the sigma-algebra $\mathfrak{F}_t \vee \sigma(X_{t+1})$ resulting from choosing $Y_{t+1} = 0$, Jensen's inequality and the convexity of the value function imply that

$$\mathbb{E}_t \left[V(S_{t+1}) \mid X_{t+1}, Y_{t+1} = 1 \right] - \mathbb{E}_t \left[V(S_{t+1}) \mid X_{t+1}, Y_{t+1} = 0 \right] \geq 0. \tag{2}$$

Thus, an optimal policy may always pass the document along to the analyst if $\text{Prob}_t\{Z_{t+1} = 1 \mid X_{t+1}\} \geq c$, that is, if the expected one-period reward is nonnegative (note that we have scaled the costs by assuming that the reward is equal to 1). More significantly, an *optimal* policy will sometimes choose to pass the document along even in situations in which the expected one-period reward is negative because the immediate one-period cost is offset by the term (2). This term may be thought of as a learning bonus, or as the value of the information gained from the analyst's feedback. The tradeoff between the learning bonus and a negative one-period reward is an example of the classic tradeoff between exploration and exploitation. In the next section we present a specific example for which the Bellman recursion may be solved numerically and the optimal tradeoff between exploration and exploitation found.

The problem is easily illustrated for the case where the unconditional document distribution is known. It is possible to show that the Q factors are given by

$$\begin{aligned} Q(S_t, X_{t+1}, 0) &= \gamma V(S_t) \\ Q(S_t, X_{t+1} = x, 1) &= -c + \frac{(a_{t,x+1})\left(1 + \gamma V(a_t + e_x, b_t)\right) + (b_{tx} + 1)\gamma V(a_t, b_t + e_x)}{a_{tx} + b_{tx} + 2}. \end{aligned}$$

This gives us Bellman's recursion as

$$V(a,b) = \sum_{x \in \mathcal{X}} P_{01}(x) \max \left\{ \gamma V(a,b), -c + \frac{(a_x + 1)\left(1 + \gamma V(a + e_x, b)\right) + (b_x + 1)\gamma V(a, b + e_x)}{a_x + b_x + 2} \right\}. \tag{3}$$

The posterior on p_x becomes progressively sharper as the sum $a_x + b_x$ increases. In the limit as $a_x + b_x \to \infty$, we have $p_x = \mathbb{E}p_x = (a_x + 1)/(a_x + b_x + 2)$. If we know p_x exactly and we see $X^t = x$ we know that the value of choosing $Y^t = 1$ is $\mathbb{E}\left[Z_{t+1} \mid X_{t+1} = x, p_x \right] = p_x$, with no learning bonus. We know then that the optimal decision is $Y_t = I_{\{p_x > c\}}$. In the limit as $\min_x a_x + b_x \to \infty$, we know every p_x exactly and the expected reward obtained at time t is $\mathbb{E}\left[(-c + p_{X_t})^+ \mid p \right] = \sum_{x \in \mathcal{X}} P_{01}(x)(-c + p_x)^+$. Then in the limit as $\min_x a_x + b_x \to \infty$, we set $p_x = \lim (a_x + 1)/(a_x + b_x + 2)$ and we have

$$\lim_{\min_x a_x + b_x \to \infty} V(a,b) = \sum_{t=1}^{\infty} \sum_{x \in \mathcal{X}} P_{01}(x)(-c + p_x)^+ = \frac{1}{1-\gamma} \sum_{x \in \mathcal{X}} P_{01}(x)(-c + p_x)^+. \tag{4}$$

3.1.3 A Value Iteration Algorithm

This value iteration algorithm approximates the value function, and serves to illustrate the principle, but will be extremely slow for all but very small values of d. The algorithm requires that we choose a parameter N which pays the role of ∞ [Warren - do I have this right? -paul], for which larger

values will increase accuracy but slow the running time. We now approximate $V(a,b)$ by (4) with $p_x = (a+1)/(a+b+2)$ for all a,b with $\min_x a_x + b_x \geq N$. This approximation defines a new optimization problem with value function $\widetilde{V}$ in which $\widetilde{V}(a,b)$ is defined according to (4) for a,b such that $\min_x a_x + b_x \geq N$, and obeys Bellman's recursion (3) for other a,b. We have the inequality $\widetilde{V} \geq V$, since the approximation (4) is an upper bound on V. As N increases to infinity, $\widetilde{V}$ decreases to V.

We can compute a sequence of approximations $\bar{v}^n$ decreasing to $\widetilde{V}$. To do this we must store the values of $\bar{v}^n(a,b)$ with a_x ranging from 0 to N and b_x ranging from 0 to $N - a_x$. We do not actually need to store $\bar{v}^n$ for those a, b with every x satisfying $a_x + b_x = N$, but doing so simplifies implementation without harming correctness, and only marginally degrades performance. B Begin with $\bar{v}^0(a,b)$ given by (4) for every a and b, not just those a and b with $\min_x a_x + b_x = N$. Then compute $\bar{v}^{n+1}$ from $\bar{v}^n$ according to the recursion

$$\bar{v}^{n+1}(a,b) = \sum_{x \in \mathcal{X}} P_{01}(x) \max \left\{ \gamma \bar{v}(a,b), -c + \frac{(a_x+1)\left(1+\gamma \bar{v}^n(a+e_x,b)\right) + (b_x+1)\gamma \bar{v}^n(a,b+e_x)}{a_x+b_x+2} \right\}.$$

At each iteration n, we compute this recursion for every a_x ranging from 1 to N and b_x ranging from 1 to $N - a_x$, assuming that $\bar{v}^n(a,b)$ is given by $\widetilde{V}(a,b)$ for (a,b) outside this range. We then advance to the next iteration until we are satisfied that $\bar{v}$ is "sufficiently close" to $\widetilde{V}$. The error can be characterized in a rigorous way in terms of the sup-norm.

[Warren – do we need to make the above claim?]

To store the function $\bar{v}^n$ for one value of n, the number of values we need to store is

$$|\{a, b \in \mathbb{N} : a + b \leq N\}|^d = \left| \sum_{a=1}^{N} N - a \right|^d = \left[N(N-1)/2 \right]^d .$$

A naive implementation would store $\bar{v}^n$ for two different values of n at a time, one for the previous iteration and one for the iteration being computed. A more sophisticated implementation could improve this by updating the $\bar{v}^n$ in place, but would still need to store at least $[N(N-1)/2]^d$ values at each step. With N set to 12, this amounts to 66^d values. Using 16-bit precision, and limiting ourselves to working entirely in RAM on a single computer with 3GB available memory, this will limit us to $d \leq \log\left(\frac{3\times 10^9 \text{bytes}}{2\text{bytes}}\right) / \log(66) \approx 5.04$.

This illustrates the principle. In practice, we can use approximate dynamic programming and exploit the convexity of the value function (see Powell (2007), chapter 11).

3.1.4 Dynamic ranking and selection

If we are in an on-line learning situation where many cases arrive at once, and we assume that the effect of the labels on the probability of "value" independent, we have, at each step the classic multi-armed bandit problem to which Gittins indices are suited. Alternatively, we may formulate an off-line learning problem, where we budget to learn as much as we can from a set of labels, after which we apply our knowledge to solve a problem. If we again assume that measurements are uncorrelated, we can use the recently-proposed knowledge-gradient algorithm (Frazier et al. (2007), see also Gupta & Miescke (1994)) which uses a one-step lookahead to estimate the value of a measurement. The approach of one-step lookahead has in fact been applied to the adaptive filtering problem in the TREC setting by Zhang and Callan Zhang et al. (2003) In this heuristic, the choice of label d to measure, indicated by $x_d = 1$, is determined using

$$X^{KG}(S^n) = \arg\max_{\{x|\sum_d x_d=1\}} \mathbb{E}\left[V(S^M(S^n,x,W^{n+1})) - V(S^n)|S^n\right].$$

where S^n is the current state of knowledge about the relationship between the label and its value, $x_d = 1$ if we are choosing label d, W^{n+1} is the information gained from the next measurement, and $S^M(S^n, x, W^{n+1})$ is the updated state of knowledge (we propose to use Bayesian updating). $V(S^n)$ captures the value of the current state of knowledge. The incremental value of a measurement is called the knowledge-gradient index, and it is given by first computing

$$\zeta_d^n = -\left|\frac{\bar{\theta}_d^n - \max_{d' \neq d} \bar{\theta}_{d'}^n}{\tilde{\sigma}_d^n}\right|.$$

where $\bar{\theta}_d^n$ is the current estimate of the value of label d after n observations, and $\tilde{\sigma}_d^n$ is the change in the *variance* resulting from measuring label d (recursive formulas for $\bar{\theta}_d^n$ and $\tilde{\sigma}_d^n$ are easily derived from Bayesian concepts).

ζ_d^n is called the *normalized influence* of decision d. It measures the number of standard deviations from the current estimate of the value of decision d, given by $\bar{\theta}_d^n$, and the best alternative other than decision d. We then find

$$f(\zeta) = \zeta\Phi(\zeta) + \phi(\zeta),$$

where $\Phi(\zeta)$ and $\phi(\zeta)$ are, respectively, the cumulative standard normal distribution and the standard normal density. The knowledge gradient algorithm chooses the decision d with the largest value of $\nu_d^{KG,n}$ given by

$$\nu_d^{KG,n} = \tilde{\sigma}_d^n f(\zeta_d^n).$$

The knowledge gradient algorithm is particularly easy to implement for independent measurements. Frazier et al. (2007) shows that this method is asymptotically optimal, fullly optimal for certain special cases and has an error bound. Experimental work shows that it compares very favorably to a range of other heuristics, including the most sophisticated OCBA algorithms.

3.2 Ranking and selection with *correlated* measurements

We have recently found that, in contrast to Gittins indices, the knowledge gradient can be extended to handle the case of correlated measurements. Let

$$f(S^n) = \sum_{i=1}^{M} a_i(\Phi(c_i) - \Phi(c_{i-1})) + b_i(\phi(c_i) - \phi(c_{i-1}))$$

where a_i, b_i and c_i are given [Warren - are they given or computed?] constants that are computed for the i^{th} label. Again, S^n is our "state of knowledge," consisting of the vector of all the current means μ^n, and the covariance matrix relating the different types of labels, Σ^n. Now let $\Sigma^n(x)$ be the updated covariance matrix that will result if make the measurement decision x (which indicates the type of label we are going to measure). The knowledge gradient policy for correlated rewards is then given by

$$X^{KG} = \arg\max_x f(\mu^n, \Sigma^n(x)).$$

This is fairly easy to compute for problems with tens of thousands of labels, but would be difficult to apply to populations of hundreds of thousands or millions of documents. [WARREN: do we mean features here?] For such applications, additional research is required on feature selection is needed. For example, the methods of Latent Semantic Indexing sharply reduce the dimensionality of the space of documents, which might then be binned into a reasonable number of orthants.

3.3 Learning with Costs, in a Changing Environment

We will address this problem terms of Markov models, and we sketch the method here. Let $\{H_t, t \geq 0\}$ represent some hidden Markov process with A state-space $\{0, 1\}$ and a one-step transition probability matrix P, where $H_t = 1$ (respectively, $H_t = 0$) means that at time t the relation between interesting cases and labels holds, and at the other times it does not. For the case of intrusion., we would say that a hacker is online at time t if $H_t = 1$ and offline if $H_t = 0$. Given the process $H_t, t \geq 0$, the labels Z_t, $t \geq 0$ are conditionally independent Bernoulli random variables with arrival rates α_{H_t}, $t \geq 0$, respectively. We also assume that the content X_t of document t has the conditional probability density f_0 (respectively, f_1) given that $Z_t = 0$ (respectively, $Z_t = 1$.) At each time t we make two decisions, Y_t and G_t, where $Y_t = 1$ if we choose the document for further evaluation and 0 otherwise, $G_t = 1$ if we believe that the source of interesting documents is active and 0 otherwise. The immediate cost and reward of forwarding a document to an expert are c and 1, respectively. Additionally, one incurs an immediate misclassification cost (c_{FN} for a false negative, and c_{FP} for a false positive) if at time t the status of the source is classified incorrectly. Now the expected total discounted reward becomes

$$\mathbb{E}\sum_{t=0}^{\infty}\gamma^t\left[Y_t(Z_t - c) - c_{FN}(H_t - G_t)^+ - c_{FP}(G_t - H_t)^+\right]. \tag{5}$$

As before, the objective is to find an admissible decision rule $(Y_t, G_t)_{t\geq 0}$ which maximizes the expected total discounted net reward. The solution of this partially observed Markov decision problem (POMDP) depends on the posterior-probability-distribution process

$$\Pi_t := \text{Prob}\{H_t = 1 \mid X_0, X_1, \ldots, X_{t-1}, X_t, Y_0, Y_1, \ldots, Y_{t-1}, Z_0Y_0, Z_1Y_1, \ldots, Z_{t-1}Y_{t-1}\}, \qquad t \geq 0,$$

which satisfies

$$\Pi_t \propto (1 - \Pi_{t-1})\,[\alpha_0]^{Y_{t-1}Z_{t-1}}\,[1 - \alpha_0]^{Y_{t-1} - Y_{t-1}Z_{t-1}}P_{01}f_1(X_t) + \Pi_{t-1}[\alpha_1]^{Y_{t-1}Z_{t-1}}\,[1 - \alpha_1]^{Y_{t-1} - Y_{t-1}Z_{t-1}}\,P_{11}f_1(X_t), \qquad t \geq 0$$

with the proportionality constant

$$(1 - \Pi_{t-1})\,[\alpha_0]^{Y_{t-1}Z_{t-1}}\,[1 - \alpha_0]^{Y_{t-1} - Y_{t-1}Z_{t-1}}\,[P_{00}f_0(X_t) + P_{01}f_1(X_t)] + \Pi_{t-1}[\alpha_1]^{Y_{t-1}Z_{t-1}}\,[1 - \alpha_1]^{Y_{t-1} - Y_{t-1}Z_{t-1}}\,[P_{10}f_0(X_t) + P_{11}f_1(X_t)], \qquad t \geq 0.$$

Note that the *sufficient statistic* for this problem, $\{\Pi_t, \mathfrak{F};\, t \geq 0\}$ is a controlled Markov chain on $[0, 1]$ adapted to the filtration

$$\mathfrak{F}_t \;=\; \sigma\{X_0, X_1, \ldots, X_{t-1}, X_t, Y_0, Y_1, \ldots, Y_{t-1}, Z_0Y_0, Z_1Y_1, \ldots, Z_{t-1}Y_{t-1}\}, \qquad t \geq 0.$$

The expectation in (5) can be rewritten as

$$\begin{aligned}
&\sum_{t=0}^{\infty}\gamma^t\mathbb{E}\left[Y_t\Big(\mathbb{E}[Z_t \mid \mathfrak{F}_t] - c\Big) - c_{FN}(1 - G_t)\,\text{Prob}\{H_t = 1 \mid \mathfrak{F}_t\} - c_{FP}G_t\,\text{Prob}\{H_t = 0 \mid \mathfrak{F}_t\}\right] \\
&\qquad = \sum_{t=0}^{\infty}\gamma^t\mathbb{E}\left[Y_t\Big((1 - \Pi_t)\alpha_0 + \Pi_t\alpha_1 - c\Big) - c_{FN}(1 - G_t)\Pi_t - c_{FP}G_t(1 - \Pi_t)\right] \\
&\qquad = \sum_{t=0}^{\infty}\gamma^t\mathbb{E}\left[Y_t\Big(\Pi_t(\alpha_0 - \alpha_1) - c + \alpha_0\Big) - c_{FN}\Pi_t - \Big(c_{FP} - (c_{FN} + c_{FP})\Pi_t\Big)G_t\right],
\end{aligned}$$

This, in turn, implies that the maximum expected total discounted net reward is attained by

$$Y_t^* = \left\{ \begin{array}{ll} 1, & \Pi_t \geq \dfrac{c-\alpha_0}{\alpha_1-\alpha_0} \\ 0, & \text{otherwise} \end{array} \right\} \qquad \text{and} \qquad G_t^* = \left\{ \begin{array}{ll} 1, & \Pi_t \geq \dfrac{c_{FP}}{c_{FP}+c_{FN}} \\ 0, & \text{otherwise} \end{array} \right\}. \tag{6}$$

Since we have said that when the hidden Markov process H is in state 1 (rather than 0), then interesting documents are produced at a higher rate, it makes sense to assume $\alpha_0 \leq \alpha_1$. Now note that in (6) we have $Y_t^* \equiv 0$ if $\alpha_0 \leq \alpha_1 \leq c$; that is, no documents are forwarded for examination if it is too expensive. Similarly, $Y_t^* \equiv 1$ if $c \leq \alpha_0 \leq \alpha_1$; that is, every document will be forwarded to an expert if examination is very cheap. The more interesting and realistic case is when $\alpha_0 < c < \alpha_1$, in which case the optimal strategy is given by (6).

To handle more realistic cases where arrival rates α_0 and α_1, transition probabilities $(P_{ij})_{i,j\in\{0,1\}}$, and densities f_0 and f_1 are all unknown, we propose to treat all of these unknowns as random variables with suitable prior probability distributions. Then we will derive the dynamics of the corresponding posterior-probability-distribution process, rewrite the expected total discounted costs in terms of this process, and use (approximate) dynamic programming techniques to solve it.

In the past we have successfully analyzed similar POMDPs. Dayanik et al. (2007*a*), Dayanik & Goulding (n.d.), Dayanik et al. (2007*b*), **?** studied stochastic systems which may undergo sudden changes at unknown and unobserved times and determined Bayesian quickest change detection and identification rules in discrete time. In continuous time, Bayraktar et al. (2006) have explicitly characterized the solution of adaptive Poisson disorder problem, while Dayanik & Sezer (2006) have proposed nearly-optimal online algorithms to detect a sudden unknown unobservable change in the arrival rate and mark distribution of compound Poisson processes.

3.4 Random Walk Models: the Distribution of Costs and Benefits

We speak of the expected value, which is to be optimized in the selection of a rule or policy. Let us now refine our terminology and refer to the "policies;; of hte preceeding sections as "rules". In the larger picture, the methods used to determine the rules, from the data, will be called "policies". Since the cases from which we learn arrive in random orders, the actual performance of any policy, and of the resulting rules will itself be a random variate. We will work to understand the range of variability of this value, in deciding whether to rely on a policy in important applications.

One particularly important class of models produce rules, for each label, of the precise form: "continue sending items with this label for evaluation until an irrevocable decision is reached to either send all, or send no more". The net present value of this rule can be computed precisely, for any given value of the cost c and the probability p that cases with this label are, indeed, "positive cases". For such rules one may compute the probability that the rule will fire when exactly k items with this label have been examined. With this information, one may compute not only the expected value, but also the complete distribution of the expected value of this policy, as a function of the , the probability p, and the cost, discount and/or horizon parameters.

An example of such a tractable policy class is: $D =$"decide to stop when the number of positive items seen, g, minus the number of negative items seen b falls below a preset threshold". The probability that a specific rule D, resulting from this policy, fires at step k can be computed using random walks with an absorbing barrier. Other stopping rules, corresponding to different degrees

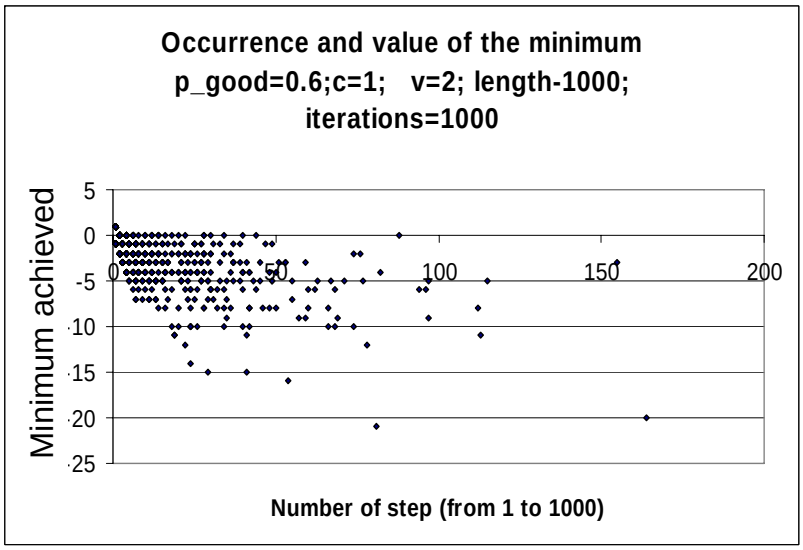


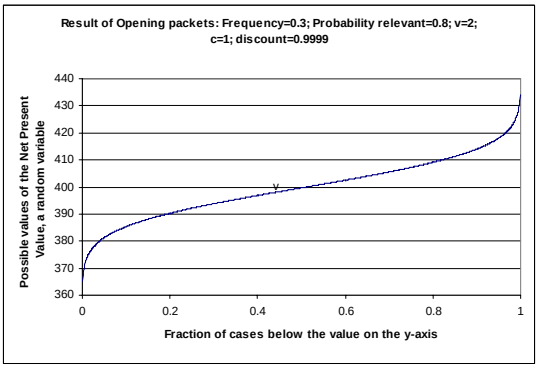
Figure 1: As the value of $g - b$ executes a random walk, it touches some minimum. The scatter plot shows the step at which the minimum is achieved, and the value of the minimum, for 1,000 pseudo-random sequences of length 1,000.

of certainty about the prior estimate of the probability of good outcomes, may depend on a more complicated procedure, such as the rules used in Sequential Analysis.

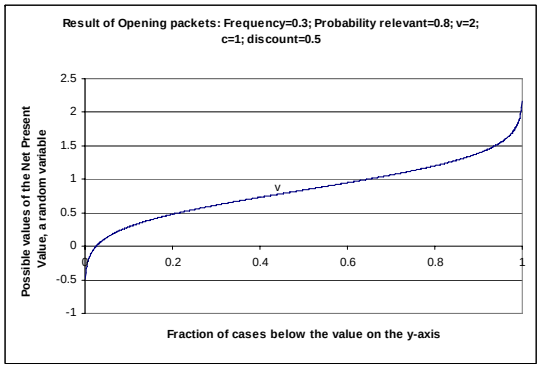
The probability that this rule results in a particular net present value can be expressed in closed form. Specifically, by the reflection principal, we can compute the distribution of the probability that the state will reach any particular value of $b-g$ at precisely the kth step. This is the difference of two binomial distributions, rescaled by a bias factor determined by the Radon-Nikodym derivative of the actual distribution with respect to the symmetric distribution. For large numbers of steps the result can be approximated by the normal distribution. For small number of steps, this approximation is unreliable. A general sense of the behavior of a rule can be found by simulating the process.

An example in 1, shows, for a particular choice of the parameters, the step at which the random walk reaches its minimum, and the value of that minimum. A simple stopping rule corresponds to a horizontal line on this plot. The proportion of the points that falls below the line is a measure of the chance that the corresponding rule leads to false rejection of the associated label. However, the lower we set that line, the greater the chance that a label that is not "sufficiently valuable" will also be accepted. Thus, this policy generates a parameterized family of rules (by the position of the horizontal line) and yields an Operating Characteristic (analogous to the ROC concept in Sensor Analysis) which summarizes the range of possible performance of the policy.

While the sign of the expected value will be the same for various discounting factors (for fixed cutoff rule), the ability of the rule to correctly discriminate among the labels will depend on the discounting. We show the value achieved using a cutoff rule of this type, for the case of a small discounting of future gains 3.4a and a large one 3.4b. When little benefit is realized from long term gains it is possible to get only a negative benefit, even when processing a label for which the long term expected value is positive.



3.4a: The results of a simulation for specific values of the parameters of the problem, for the rule: terminate cases that go negative in value. The discount factor is very close to 1, so that long term benefits are realized. It would be very rare to terminate the service.



3.4b: With a sharper discount, it is possible that the net present value will be negative, although this is still a rare event. In either case, this "label" ought to be identified as one worth submitting.

[WE WANT PETER's NEW GRAPH VALUE vs Horizon here

CAPTION. If the cost is equal to 0.5, and the prior distribution of the probability of relevance is uniform, then with no time to learn (Horizon-0) the expect value us 0. If the cost is lower, it is positive. But if the cost is higher, it would be negative. As the lower curve shows, it is only with sufficient time to learn, that the rule can distinguish among the good and bad situations, and learn whether to continue submitting for evaluation.)

THE BODY TEXT should say pretty much the same thing, but I am running out of words at this point.

The probability of false negatives can be computed for any specific value of the rate at which cases of interest are detected and sent for examination. These rigorous expressions can be convolved with the believed joint distribution of the probability of value, and the amount of value, for each label. Thus any specific calculatable rule can be joined with any assumed distribution of the probability of valuable items $\rho(p)$ to produce a precise estimate of the value achieved and missed, by applying the rule.

3.5 Experimentation, Application and Validation

3.5.1 Feature engineering

The TREC Adaptive Filtering collection will serve to test the ideas proposed here, but requires some processing and feature engineering to get beyond the established levels of performance. Proper selection of features, and such technical details as the specific monotone transform applied to the term frequencies, and corrections for document length are all important in optimizing performance. Based on the experiences reported in TREC, we anticipate that a rich and complex set of features will be needed. As discussed in section 2.3. These will include: terms; term pairs and phrases; term co-occurrences; character n-grams (4- or 5-grams have proven particularly effective) and additional weighting of lead terms or header terms.

At Rutgers we have considerable experience dealing with these texts, and have a suite of Perl

scripts for preprocessing. We will use the Lucene package to manage retrieval and filtering aspects, as it contains a number of the best-performing representation formulas. Finally, Rutgers is the developer of the BXR package, a powerful Bayesian multi-variate regression package that has proven particularly effective in reducing huge sets of features to manageable size, while achieving state-of-the-art performance in a number of arenas.

3.5.2 Experimentation and Validation

We will compare approaches, as they are developed, on cases having both independent and correlated features or labels. We will use simulation to test results, but, more significantly, we will validate those findings by testing the most promising algorithms on the TREC materials. There are many other applications, such as management of mobile sensors (e.g. sensing nuclear radiation), and spatial problems, where correlations will naturally correlations arise due to the relationships based on geographic proximity (e.g. measuring nuclear radiation or the flow of goods and people across a continuous border). We will seek other opportunities (and other sources of support) to extend these ideas to those cases.

4 Impacts

4.1 Impacts on science and technology

Advances in methods for collecting information arise in stochastic optimization algorithms (e.g. Spall (2003)), optimization of simulation (e.g. Chang et al. (2007)), sensor management (Castanon (1997)), sequential design of experience (Bechhofer et al. (1995)), and document processing. This research should both accelerate the performance of methods for solving these problems, as well as producing more robust solutions. This research has the potential of accelerating a broad class of Monte Carlo-based optimization algorithms, with specific potential for improving the process of identifying important documents.

4.2 Impacts on other scientific fields

This research will have scientific impacts far beyond the specific arena for which we have test collections that can support development and validation. Examples of such areas include: speedy medical diagnosis or differentiation of diseases (especially in threat of chemical or biological warfare). When central laboratories are a limiting resource, the quick decision of which samples to send for more detailed analysis is a precise replicate of the problem studied here. The problem of identifying events in noisy data occurs in fields ranging from seismology to meteorology, and the results obtained here will be translatable to those settings.

4.3 Impacts on science and technology education

The proposed research will directly contribute to the education of two graduate assistants, one at Rutgers, in the Department of Computer Science or Information Science, and one at Princeton in the Department of Operations Research and Financial Engineering. The PIs will have the opportunity to advance the education of several undergraduates, through the DIMACS Research Experience for Undergraduates. The work will flow into graduate courses at both institutions. At Princeton, the research will be incorporated directly into a new undergraduate course on Optimal Learning (ORF 418). Some of the resulting software will be suitable for educational use.

4.4 Impacts on society at large

The challenge of collecting information arises in a broad range of settings. This research may help with biomedical research through the intelligent design of expensive experiments. This research may

be put into practice at the National Ignition Facility at Lawrence Livermore, soon to become the world's largest laser facility (proposal pending), which faces a problem of determining when to use sensors to detect the state of lenses. The research may help improve national security by accelerating the process of identifying promising websites and documents.

There are many problems for which change-point detection, which is a component of this work, is important, and must be coupled with the problem of learning classifications. One notable case is spam-filtering, although the intense efforts in that area make it possible that general advances of the type sought here will have to be heavily re-engineered to be effective. Other problems for which many activities must be monitored and only a tiny fraction sent for expensive scrutiny are (a) computer intrusion detection [some data is available for this, from the 1999 KDD Cup]. (b) cell-phone network virus detection (c) credit card fraud and (d) calling-card fraud.

5 The research team

Paul Kantor is an Information Scientist, who has worked on the evaluation of information systems, and related problems, with support from the NSF, DARPA and ONR. Warren Powell is XXXXX. Savas Dayanik is an applied probabilist who is interested in the applications of sequential stochastic optimization to homeland security, biosurveillance, and finance.

The project is formulated as a "small project," but, because it involves three faculty investigators rather than one, it is budgeted for two years, rather than three. The principal investigators are currently working together on this problem class with modest support from the Department of Homeland Security through the DyDAn Center for Dynamic Data Analysis, based at Rutgers. The proposed work will complement and extend that work, and has no direct overlap with it.

6 Schedule of Work and Milestones

While we present the proposed research as a series of problems (uncorrelated labels; correlated labels; and change point detection) in fact work on all three aspects will begin at once. The product of the proposed work will be simulations, rigorous analytical results, demonstrations by application to data streams, drawn from the TREC adaptive filtering track, and extensions to other situations as data become available. Expected milestones for the four tracks of work are:

(1) approximate dynamic programming to develop rules for uncorrelated and correlated cases **(WP)**: uncorrelated case rules (6 months) correlated cases (12 m) integration with change point (18m) validation and writing (24m). (2) Change point detection **(SD)** (6m) (12m) Integration with ADP (18m) validation and writing (24m) (3) determination of the distribution of results using random walks and Martingales **(PK)**: Closed computable expressions for uncorrelated ADP rules (6m) Closed computable expressions for correlated ADP rules (12m) Closed computable expressions for change point rules (18m) validation and writing (24m) (4) validation with TREC data **(PK)** Preprocessing and data set up; feature engineering (6m) Test and validate uncorrelated rules (12m) Test and validate correlated rules (18m) Test and validate change point rules (24m)

References

Bayraktar, E., Dayanik, S. & Karatzas, I. (2006), 'Adaptive Poisson disorder problem', *Ann. Appl. Probab.* **16**(3), 1190–1261.

Bechhofer, R., Kiefer, J. & Sobel, M. (1968), *Sequential Identification and Ranking Procedures*, University of Chicago Press, Chicago.

Bechhofer, R., Santner, T. & Goldsman, D. (1995), *Design and Analysis of Experiments for Statistical Selection, Screening and Multiple Comparisons*, J.Wiley & Sons, New York.

Bickel, J. E. & Smith, J. E. (2006), 'Optimal sequential exploration: A binary learning model', *Decision Analysis* **3**(1), 16–32.

Castanon, D. (1997), 'Approximate dynamic programming for sensor management', *Decision and Control, 1997., Proceedings of the 36th IEEE Conference on*.

Chang, H., Fu, M., Hu, J. & Marcus, S. (2007), *Simulation-Based Algorithms for Markov Decision Processes*, Springer, Berlin.

Chen, C., Lin, J., Yücesan, E. & Chick, S. (2000), 'Simulation Budget Allocation for Further Enhancing the Efficiency of Ordinal Optimization', *Discrete Event Dynamic Systems* **10**(3), 251–270.

Cohn, D. A., Ghahramani, Z. & Jordan, M. I. (1996), 'Active learning with statistical models', *J. of Artificial Intelligence* **4**, 129–145.

Dayanik, S. & Goulding, C. (n.d.), Detection and identification of an unobservable change in the distribution of a markov-modulated random sequence, Submitted to *IEEE Transactions on Information Theory*. Preprint available at http://www.princeton.edu/~sdayanik/papers/markov.pdf.

Dayanik, S. & Sezer, S. O. (2006), 'Compound Poisson disorder problem', *Math. Oper. Res.* **31**(4), 649–672.

Dayanik, S., Goulding, C. & Poor, H. V. (2007*a*), Bayesian sequential change diagnosis, *Math. Oper. Res.* To appear. Preprint available at http://www.princeton.edu/~sdayanik/papers/diagnosis.pdf.

Dayanik, S., Goulding, C. & Poor, H. V. (2007*b*), Joint detection and identification of an unobservable change in the distribution of a random sequence, *in* 'CISS', IEEE, pp. 68–73.

DeGroot, M. H. (1970), *Optimal Statistical Decisions*, John Wiley and Sons.

Duff, M. & Barto, A. (1997), 'Local bandit approximation for optimal learning problems', *Advances in Neural Information Processing Systems* **9**, 1019.

Frazier, P., Powell, W. B. & Dayanik, S. (2007), A knowledge gradient policy for sequential information collection, Working Paper.

Fu, M. (2002), 'Optimization for simulation: Theory vs. practice', *INFORMS Journal on Computing* **14**(3), 192–215.

Gittins, J. (1989), *Multi-Armed Bandit Allocation Indices*, John Wiley and Sons, New York.

Gittins, J. C. & Jones, D. M. (1974), A dynamic allocation index for the sequential design of experiments, *in* J. Gani, ed., 'Progress in Statistics', pp. 241–266.

Goldsman, D. & Nelson, B. (1994), Ranking, selection and multiple comparisons in computer simulation, *in* J. D. Tew, S. Manivannan, D. A. Sadowski & A. F. Seila, eds, 'Proceedings of the 1994 Winter Simulation Conference'.

Gupta, S. & Miescke, K. (1994), 'Bayesian look ahead one stage sampling allocations for selecting the largest normal mean', *Statistical Papers* **35**, 169–177.

He, D., Chick, S. E. & Chen, C.-H. (2007), 'Opportunity cost and ocba selection procedures in ordinal optimization for a fixed number of alternative systems', *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews* **37**(5), 951–961.

Hull, D. A. (1998), The TREC-6 filtering track: Description and analysis, *in* E. M. Voorhees & D. K. Harman, eds, 'The Sixth Text REtrieval Conference (TREC 6): NIST Special Publication 500-240.', Department of Commerce, National Institute of Standards and Technology, U.S. Government Printing Office, Washington, DC, pp. 45–68.

Kaelbling, L. P. (1993), *Learning in Embedded Systems*, MIT Press, Cambridge, MA.

Lewis, D. D. (1997), The trec-5 filtering track, *in* E. M. Voorhees & D. K. Harman, eds, 'The Fifth Text REtrieval Conference (TREC 5): NIST Special Publication 500-238.', Department of Commerce, National Institute of Standards and Technology, U.S. Government Printing Office, Washington, DC, pp. 75–96.

Powell, W. B. (2007), *Approximate Dynamic Programming: Solving the curses of dimensionality*, John Wiley and Sons, New York.

Powell, W. B., Ruszczyński, A. & Topaloglu, H. (2004), 'Learning algorithms for separable approximations of stochastic optimization problems', *Mathematics of Operations Research* **29**(4), 814–836.

Singhal, A. (1998), AT&T at TREC-6, *in* E. M. Voorhees & D. K. Harman, eds, 'The Sixth Text REtrieval Conference (TREC 6): NIST Special Publication 500-240.', Department of Commerce, National Institute of Standards and Technology, U.S. Government Printing Office, Washington, DC, pp. 215–226.

Spall, J. C. (2003), *Introduction to Stochastic Search and Optimization: Estimation, Simulation and Control*, John Wiley & Sons, Hoboken, NJ.

Swisher, J., Jacobson, S. & Yücesan, E. (2003), 'Discrete-event simulation optimization using ranking, selection, and multiple comparison procedures: A survey', *ACM Transactions on Modeling and Computer Simulation (TOMACS)* **13**(2), 134–154.

Thorsley, D. & Teneketzis, D. (2007), 'Active acquisition of information for diagnosis and supervisory control of discrete event systems', *Discrete Event Dynamic Systems* **17**, 531–583.

Topaloglu, H. & Powell, W. B. (2006), 'Dynamic programming approximations for stochastic, time-staged integer multicommodity flow problems', *Informs Journal on Computing* **18**(1), 31–42.

Walker, S., Robertson, S. E., Boughanem, M., Jones, G. J. F. & Jones, K. S. (1998), Okapi at TREC-6 automatic ad hoc, vlc, routing, filtering and qsdr., *in* E. M. Voorhees & D. K. Harman, eds, 'The Sixth Text REtrieval Conference (TREC 6): NIST Special Publication 500-240.', Department of Commerce, National Institute of Standards and Technology, U.S. Government Printing Office, Washington, DC, pp. 125–136.

Xu, H., Yang, Z., Wang, B., Liu, B., Cheng, J., Liu, Y., Yang, Z., Cheng, X. & Bai, S. (2002), TREC 11 experiments at cas-ict: Filtering and web., *in* E. M. Voorhees & L. P. Buckland, eds, 'The Eleventh Text REtrieval Conference (TREC 11): NIST Special Publication 500-251', Department of Commerce, National Institute of Standards and Technology, U.S. Government Printing Office, Washington, DC.

Zhai, C., Jansen, P., Stoica, E., Grot, N. & Evans, D. A. (1999), Threshold calibration in clarit adaptive filtering., *in* E. M. Voorhees & D. K. Harman, eds, 'The Seventh Text REtrieval Conference (TREC 7): NIST Special Publication 500-242.', Department of Commerce, National Institute of Standards and Technology, U.S. Government Printing Office, Washington, DC, pp. 149–156.

Zhang, Y. & Callan, J. (2001), Maximum likelihood estimation for filtering thresholds, *in* 'In Proceedings of the 24st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval'.

Zhang, Y., Xu, W. & Callan, J. (2003), Exploration and exploitation in adaptive filtering based on bayesian active learning, *in* 'In Proceedings of the Twentieth International Conference on Machine Learning (ICML)'.