퍼지 및 지능시스템학회 논문지 2005, Vol. 15, No. 4, pp. 395-399

강화학습을 이용한 주제별 웹 탐색

Topic directed Web Spidering using Reinforcement Learning

임수연

SooYeon Lim

경북대학교 컴퓨터공학과

요 약

본 논문에서는 특정 주제에 관한 웹 문서들을 더욱 빠르고 정확하게 탐색하기 위하여 강화학습을 이용한 HIGH-Q 학습 알고리즘을 제안한다. 강화학습의 목적은 환경으로부터 주어지는 보상(reward)을 최대화하는 것이며, 강화학습 에이전트는 외부에 존재하는 환경과 시행착오를 통하여 상호작용하면서 학습한다. 제안한 알고리즘이 주어진 환경에서 빠르고 효율적임을 보이기 위하여 넓이 우선 탐색과 비교하는 실험을 수행하고 이를 평가하였다. 실험한 결과로부터 우리는 미래의 할인된 보상을 이용하는 강화학습 방법이 정답을 찾기 위한 탐색 페이지의 수를 줄여줌으로써 더욱 정확하고 빠른 검색을 수행할 수 있음을 알 수 있었다.

Abstract

In this paper, we presents HIGH-Q learning algorithm with reinforcement learning for more fast and exact topic-directed web spidering. The purpose of reinforcement learning is to maximize rewards from environment, and reinforcement learning agents learn by interacting with external environment through trial and error. We performed experiments that compared the proposed method using reinforcement learning with breath first search method for searching the web pages. In result, reinforcement learning method using future discounted rewards searched a small number of pages to find result pages.

Key words : 강화학습, Q-학습, 에이전트, 웹 탐색, 보상,

1. 서 론

에이전트(agent)는 사용자가 제공한 매개변수를 사용하여 인터넷의 전부 또는 일부를 검색하여 사용자가 관심을 가지고 있는 분야의 정보를 수집하고, 이것을 매일 또는 정해진 주기로 제공한다. 웹사이트들을 방문하여 웹문서 및 기타 여러 가지 정보를 읽어오는 에이전트 프로그램을 스파이더라고 부른다. 스파이더는 사이트의 웹문서들을 여러 가지 방식으로 탐색할 수 있는데, 그중 한 방법은 각 문서에 정의되어 있는 모든 하이퍼텍스트 링크를 따라 모든 문서를 읽을 때까지 탐색하는 것이다.

웹 전체를 대상으로 하여 구별되는 많은 웹문서들을 찾는 것이 목적인 스파이더링(spidering)은 여러 가지 방식으로 탐색할 수 있는데, 그중 한 방법은 각 페이지에 정의되어 있는 모든 하이퍼텍스트 링크를 따라 모든 페이지를 읽을 때까지 탐색하는 것이다. 이와 같은 목적에 적합한 탐색 전략은 넓이 우선 탐색(Breath First Search) 방법이다. 그러나 특정 주제(topic)나 종류에 대한 웹 페이지들을 찾는 것이 목적이라면 주제와 관련이 없는 하이퍼링크를 따라가는 것이 상당히 소모적인 일일 것이다. 왜냐하면, 사용자가 관심이 있는 문서로 유입되는 링크가 집중되어 있을 가능성이 높기 때문이다[8].

지능적인 스파이더(intelligent spider)라면 주제와 관련이 있는 문서들과 연결되어 있는 문서들을 집중적으로 탐색할 것이다. 또한 원하는 정보를 빠르고 정확하게 찾을 때 에이전트는 사회 환경 속에서의 시행착오와 즉각적으로 얻을 수 있는 정보, 학습된 지식, 특별한 경험들을 통하여 학습할 수 있다.

강화학습은 최적의 해를 찾기 위해 순차적인 판단을 하는 기계 학습(machine learning)의 한 종류이며, 감독학습(supervised learning)과 구별되는 점은 학습자(learner)가 어떤 특별한 상태에 대한 올바른 행동인지를 알려주는 것이 아니라 선택한 행동이 얼마나 좋은지를 나타내는 스칼라(scalar) 형태의 보상 값으로 알려준다는 것이다. 강화학습은 사전 지식이 필요 없고, 예제가 아닌 경험과 관찰을 통해서 학습한다. 강화학습 에이전트는 각각의 유용한 행동들을 수행함으로써 기대할 수 있는 미래의 할인된 보상 값들의 합을 나타내는 스칼라 값을 지연된(delayed) 이득으로 표현한다. 강화학습의 장점들 중의 하나는 즉각적인 이득은 주지 않더라도 미래에 이득이 되는 행동(action)들에 대하여 유용한 정도를 측정하는 형식을 제공한다는 것이다.

본 논문은 강화학습을 이용하여 특정 주제에 관한 웹문서들을 더욱 빠르고 정확하게 탐색하는 스파이더를 구축하는 것을 목적으로 한다. 또한 제안한 학습 알고리즘이 주어진 환경에서 빠르고 효율적으로 정답을 찾는 것을 보이기 위한 실험을 수행하고 이를 평가하고자 한다. 실험대상은 MIT (Massachusetts Institute of Technology)의 전자전기 컴퓨터 공학부(Department of Electrical Engineering and Computer Science) 홈페이지들로 정하였다.

접수일자 : 2005년 7월 5일
완료일자 : 2005년 7월 29일

2. 관련연구

웹을 탐색할 때 사용자의 질의에 대하여 지능적으로 정보를 추출하거나 최적의 행동을 보이기 위한 많은 연구가 이루어지고 있다.

Letizia[5]는 MIT Media Lab에서 개발된 웹 브라우징을 도와주는 사용자 인터페이스 에이전트이며, 사용자의 브라우징 행태로부터 사용자가 무엇에 관심을 가지는지를 추론하는 휴리스틱에 의해 확장된 넓이 우선 탐색에 기반한 브라우징 전략을 자동으로 실현한다.

Laser[6]는 높은 검색 순위를 갖는 페이지가 검색엔진이 관련정보를 찾기 위하여 검색을 시작하는 시작점이 될 수 있다는 생각에서 출발하여 웹 페이지들을 인덱싱할 때 기계학습 방법을 적용한 시스템이다. 다른 문서 집합과는 달리 웹 페이지들은 하이퍼텍스트들의 그래프로 이루어져 있다. 웹에 있는 정보를 찾기 위하여 검색엔진은 하이퍼링크들을 따라 페이지들 사이를 이동하고, 하나의 고립된 문서가 아닌 이웃한 문서들에 대한 정보들까지 고려하여야 한다. 따라서 검색엔진이 찾는 정보와 관련된 페이지를 발견했을 때 확실한 보상을 주는 강화학습은 유용한 방법이 될 것이다. 검색함수는 TFIDF 벡터 공간 모델에 기반을 두고 있으며, 가중치의 계산에 HTML문서의 마크업 정보를 이용하였다. 검색성능을 향상시키기 위하여 파라미터의 수를 자동적으로 최적화시키는 이 검색엔진은 카네기멜론 대학 컴퓨터공학부 웹사이트를 실험대상으로 하였다.

ARACHNID[4]는 동적이며 분산된 웹 공간에 있는 정보들을 찾기 위해 다수의 에이전트들을 사용하는 분산 시스템으로, 브리태니커 백과사전 코퍼스를 실험 대상으로 하였다. 다중 에이전트 시스템의 가장 큰 장점은 독립적인 응용 프로그램의 집합으로는 해결할 수 없는 보다 복잡한 서비스를 다른 에이전트와의 협력을 통해 제공할 수 있다는 점이다.

WebWatcher[13]는 CMU Learning Lab에서 개발된 여행정보안내 시스템으로 다른 사용자가 방문한 웹페이지 중에 사용자가 관심을 가질 만한 문서를 제안하는 기능을 가지고 있다. 이 시스템은 사용자가 취할 수 있는 행동 공간을 현재 연결되어 있는 하이퍼링크들로 한정시켰으며, 감독학습과 하이퍼링크가 존재하는 단어들의 값을 학습하는 강화학습을 결합한 방법을 사용했다.

J. Rennie[8]는 특정주제에 관한 웹 페이지들을 찾는 웹 스파이더를 구축하였다. 이 때 강화학습을 이용하여 목적 페이지(target page)가 소수인 경우와 아닌 경우를 구분하여 실험을 행하고, 즉각적인 보상과 미래에 얻어지는 보상이 탐색 결과에 미치는 영향을 분석하였다.

본 논문에서 제안한 웹 정보 검색 시스템은 강화학습을 이용하여 질의에 대한 최적의 정책을 수행하는 것을 목적으로 하며, 사용자의 관심도를 나타내는 하이퍼링크를 따라 갔을 때 나타나는 문서와 질의와의 유사도를 이용하는 HIGH-Q 학습 알고리즘을 제안한다.

3. 강화학습을 이용한 웹 문서 탐색

강화학습을 웹 탐색에 적용할 때의 문제점은 상태와 행동을 나타낼 공간이 너무나 크다는 것이다. 웹에서 주제와 관련된 문서의 수와 유입되는 링크를 가지고 있는 URL의 수는 지수적으로 늘어나고 있으므로 효율적인 웹 탐색을 위해서는 문제를 단순화하여 생각해볼 필요가 있다. 이를 위하여 본 논문에서는 상태들은 서로 독립이며 하이퍼링크로 연결된 문서들이 관련되어 있다고 가정하며 최적의 정책에 대한 해를 정의한다.

3.1 강화학습

어떤 상태에 있는 에이전트가 한 행동을 수행하고 환경으로부터 보상을 받는 것을 결정 프로세스(decision process)라고 하며, 새로운 상태가 단지 현재 상태와 행동으로 결정되는 결정 프로세스를 MDP(Markov Decision Process)라 한다. 특히 환경으로부터 받는 보상을 기반으로 수행할 행동을 배우는 학습을 강화학습이라고 한다.

강화학습은 동적 프로그래밍과 교사학습을 혼합한 형태의 학습으로서 학습을 수행하는 에이전트는 에이전트의 외부에 존재하는 환경(environment)과 시행착오(trial and error)를 통해 상호작용(interaction)하면서 학습한다. 이와 같이 스스로 작동하게 내버려두지만 원하는 행동을 하면 보상하고 그렇지 않으면 벌을 준다. 이는 많은 시행착오를 시뮬레이션(simulation)하는 학습 과정이 필요하지만, 점진적으로 성능을 향상시킬 수 있다는 장점을 가지고 있다.

각각의 Q값에 대해 학습하기 위해, Q-학습 에이전트들은 그 자신의 Q값들을 위해 n개의 Q-테이블(n×n)을 유지하며, Q-학습된 에이전트는 장기적인 행동들에 대한 결과를 예상할 수 있는 능력을 가지게 된다. 즉, 학습을 수행하는 동안 주어진 환경에서 취할 수 있는 행동을 시도하며, 외부 환경으로부터 에이전트가 선택한 행동에 대한 평가로서 스칼라형의 강화값을 받아 강화(reinforce)된다. 이는 자신이 수행한 행동에 대하여 보상값을 받아서 조금씩 좋은 방향으로 행동을 강화시키는 학습방법으로 MDP방식에 기반하고 있다. 이때, 환경은 상태(state)들로 이루어져 있으며, 행동(action)을 취함으로써 상태의 변경이 이루어진다. 특히 목표에 도달하기 위해 취한 행동들을 정책(policy)이라고 하며 에이전트의 임무는 이 제어정책(control policy)을 학습하는 것이다. 이 에이전트의 스케쥴, 즉 제어정책을 π로 나타내었을 때 우리가 풀 문제는 가장 좋은 π를 찾는 것이며 이는 가장 많은 획득 축적된 값(accumulated reward)을 얻게 하는 π를 의미한다.

에이전트는 현재의 상태에서 다음 행동을 선택하기 위하여 정책 π : S→A를 학습하게 된다. 임의의 초기상태 st에서 임의의 정책 π를 취함으로써 획득 축적된 값은 다음과 같이 정의된다.

$$V^{\pi}(s_t) \equiv r_t + \gamma\, r_{t+1} + \gamma^2 r_{t+2} + \ldots \equiv \sum_{i=0}^{\infty} \gamma^i r_{t+i} \qquad (1)$$

식(1)에서 s_t는 초기 상태, r_t는 시간 t에서의 보상이다. 보상을 합산하는 여러 가지 방법이 있는데, 여기서는 하나의 정책을 수행함으로써 얻을 수 있는 각 보상에 시간에 따른 할인율을 적용하고 합산하는 방법을 사용한다. $\gamma \in [0,1]$는 즉각적인 보상 값(immediate reward)이 아닌 지연된 값들을 결정하는 할인상수(discount factor)이다. 즉 현재 즉각적으로 받는 보상 값의 할인율은 "1"로서, 이는 멀리 있는 보상 값보다 가까이 있는 보상 값을 더 선호한다는 것을 나타낸다.

MDP에서 에이전트의 목적은 모든 상태 s에 대하여 축적된 보상을 최대로 하는 정책 π를 학습하는 것이다.

그러한 정책을 최적 정책(optimal policy)라 하고 π^*로 나타내며, 축적된 보상은 $V^*(s)$로 나타낸다.

$$\pi^* \equiv \arg\max_{\pi} V^{\pi}(s), (\forall s) \quad (2)$$

$$V^{\pi^*}(s) \Rightarrow V^*(s) \quad (3)$$

Watkins[3]가 제안한 Q-학습은 가장 널리 이용되고 있는 비모델 강화학습 방법으로서 통계적 동적 프로그래밍(stochastic dynamic programming)에 근거하고 있으며, 학습자가 선택한 행동이 환경에 어떤 영향을 미치는지에 대한 사전 지식이 없을 때 적용될 수 있다.

Q-학습은 시간 변화에 따른 적합도 차이를 학습에 이용하는 TD-학습의 한 종류이며, 리턴되는 기대값의 추정자로 TD(0)를 사용한다. 그리고 행동을 선택할 때는 현재 Q값들 중 가장 큰 추정치를 가진 행동을 선택하는 탐욕(greedy)정책을 사용한다. Q-학습 에이전트는 학습 과정에서 최적의 정책을 결정하는 Q값의 수렴을 위하여 일련의 상태-행동을 선택하는 것이 필요하다. 이 알고리즘의 학습 결정 전략은 각각의 상태-행동 쌍에 대한 장기적인 할인된 보상을 측정하는 상태-행동 가치 함수 Q에 의하여 결정된다. 따라서 상태 공간에 있는 모든 상태-행동 쌍을 반복 경험하면서, 평가값인 Q값을 기반으로 하여 환경이 주는 보상 값을 각각의 상태-행동 쌍에 대한 전략을 학습한다. 이 때, 모든 상태-행동 쌍의 값을 저장하기 위하여 룩업 테이블(lookup table)을 이용하며 Q값은 학습하지 않는 동안 감소하지 않는다.

Q-학습의 기본 개념은 다음과 같이 표현될 수 있다.

$$Q(s,a) \equiv r(s,a) + \gamma V^*(\delta(s,a)) \quad (4)$$

이 때, $Q(s,a)$는 상태 s부터 시작해서 첫 번째 행동으로 a를 적용하고, 그 후에 최적의 정책을 따라가면서 획득할 수 있는 할인 축적된 보상을 의미한다. 만일 $Q(s,a)$가 구해진다면 최적의 정책을 발견할 수 있게 되는 것이다.

$$\pi^*(s) = \arg\max_{a} Q(s,a) \quad (5)$$

3.2 학습 전략

하나의 웹 페이지 내에는 다른 웹 페이지와 연결시켜주는 하이퍼링크들이 존재한다. 사용자들은 원하는 웹 페이지들을 찾기 위하여 관련된 하이퍼링크들을 따라 탐색을 하게 된다. 웹상에서 사용자가 원하는 문서들을 찾는 문제에 강화학습을 적용할 때 각각의 웹 페이지들은 상태로, 하이퍼링크들을 따라 문서들 간을 이동해 가는 것을 행동으로 정의할 수 있다.

본 논문에서는 강화학습을 적용한 웹 탐색 알고리즘으로 Q-학습 알고리즘을 사용한다. 이 알고리즘의 학습 전략은 각각의 상태-행동 쌍에 대한 장기적인 할인 보상을 측정하는 상태-행동 가치 함수인 Q에 의해 결정된다. 에이전트가 경험하는 지각, 행동, 보상으로 이루어지는 사건 열을 에피소드(episode)라고 하며, 이는 학습을 수행하는 에이전트와 외부 환경과의 상호작용이 자연스럽게 끝나는 것을 의미한다. 여러 번의 에피소드들을 통한 Q값의 갱신은 최적 해에 수렴함은 이미 증명되어 있다[14].

Q-학습 알고리즘은 더 이상 진행할 수 없는 노드에 이른 다음에는 플레이 결과를 플레이 과정에 반영시켜서 가치 함수를 갱신하는 것으로 최종 방문한 노드의 가치 함수 값을 일정 수준(전파 상수)만큼 바로 이전의 노드들로 전파시키는 방법이다. 검색 에이전트는 각 상태에서 특정 하이퍼링크를 따라가는 행동을 취했을 때, 그 행동에 따른 보상 값을 받는 것과 함께 그 행동이 정답집합으로 가는데 얼마나 유용한가를 가치 함수를 이용하여 계산한다. 가치 함수 값인 Q값은 어떤 웹문서에서 사용자가 관심을 가지고 따라가는 행동들에 대한 할인된 보상 값의 합으로 정의된다.

가치 함수 Q값을 구하기 위하여 우선 사용자의 질의와 웹 문서들을 전처리 과정을 통하여 벡터모델로 표현하고, (코사인) 유사도를 계산하였다. 이렇게 학습된 결과를 이용하여 질의와의 관련 Q값이 가장 큰 하이퍼링크들을 따라간다면 더욱 효율적인 검색을 할 수 있을 것이다.

Q 값의 수렴을 위해서는 일련의 상태-행동을 선택하는 것이 필요하다. 이를 위하여 임의 선택방법(random selection)을 사용할 경우에는 최적 값의 수렴을 위해 Q테이블에 접근하는 횟수가 크게 증가하므로 실제 환경의 온라인 학습에 이용하기에는 부적절하다. 강화학습에서 발생하는 문제들 중 하나는 탐색(exploration)과 이용(exploitation) 사이의 균형 문제이다. 탐색은 새로운 정보를 모으기 위하여 알려지지 않은 상태와 행동을 선택하는 것을 말하며 이용은 이미 학습된 상태와 행동을 취하는 것을 말한다. 이용은 모든 가능한 상태-행동들이 Q-학습 수렴법칙을 만족하기 위하여 충분히 탐색되는 것을 보장하며, 탐색은 탐욕 정책을 적용하고 있다[11]. 보상을 얻기 위해서는 이미 알고 있는 것을 이용해야 하지만, 미래에 더 좋은 행동을 선택하기 위해서는 탐색도 중요하다. 탐색과 이용의 균형은 강화학습에서 매우 중요한 문제 중의 하나이며 여러 가지 요인에 의해 판단되어진다.

Q값 외에 기타 정보를 이용하여 탐색에 도움을 주기 위한 여러 가지 시도가 있어왔다. 기본적인 Q-학습에서는 Q값을 하나의 값으로 저장하는 것과는 달리 Q값의 분포[9,10]를 이용하거나 명시적인 탐색 보너스[12]를 이용하기도 한다. 전자의 경우 Q값의 불확실도가 높은 곳을 집중 탐색하게 함으로써 전역적 탐색을 가능하게 하지만 계산 부담이 높으며, 후자의 경우에는 상태의 방문 횟수나 예측되는 오차가 큰 상태를 더 탐색하는 방법으로 비교적 구현하기가 쉽다.

강화학습에서의 탐색과 이용의 균형 문제를 해결하기 위한 방문자 리스트는 지금까지 방문한 모든 페이지들의 하이퍼링크들을 모아놓은 것이다. 탐색 시 매 번의 실행은 시행의 순열로 이루어져 있으며 이를 에피소드라고 부른다. 한 에피소드가 끝나면 지금까지 거쳐 온 웹 페이지들의 Q값을 역으로 전파시키면서 이전의 Q값들을 갱신해준다. 하나의 에피소드가 끝나면 다음 에피소드의 출발 노드를 정하기 위해서 방문자 리스트에 있는 노드들 중 가장 큰 유사도를 가지는 노드를 선택해서 탐색을 실시한다. 이는 방문자리스트에서 다음에 탐색할 노드를 선택한다는 것이 스파이더가 마지막으로 방문한 페이지에서 나오는 하이퍼링크를 선택하는 게 아니라는 것을 의미한다.

다음의 그림 1은 본 논문에서 제안한 HIGH-Q 학습 알고리즘으로 방문자 리스트(visitant list)와 유사도를 이용하여 웹 페이지들을 탐색한다.

```
initialize QValue[]=0;
pick random node s1;
pick link for s1 to follow;
while 정답을 모두 발견
   1. while (s1.depth<= bound) and (s1.child가 존재)
        s2=s1.child;
        QValue[s1,a]=Similarity(s2, Query);
        pick node s2 with Max(QValue[s1,a]);
        s1=s2;
   2. for (from 마지막 노드 to 출발 노드)
        QValue[nPath[last-1]] = QValue[nPath[last-1]]
                              + (QValue[nPath[last]]*Rate);
    // Q값 갱신 : 한 에피소드에 대한 reward를 반영
```

그림 1. HIGH-Q 학습 알고리즘의 pseudo-code

4. 실험 및 평가

실험을 위하여 먼저 웹 문서들을 수집하고, 이미 수집한 문서들과 하이퍼링크들을 이용하여 오프라인으로 학습을 시켰다. 실험 대상은 MIT의 전자전기 컴퓨터 공학부 홈페이지들을 대상으로 총 1,181개의 문서와 이와 관련된 하이퍼링크 7,198개를 수집하였다. 수집한 코퍼스 내의 문서들은 해당 학부에 대한 안내, 행사, 연구에 관한 정보 등으로 구성되어 있었으며, 문서를 수집할 때 홈으로 다시 연결된 링크들은 제외하였다.

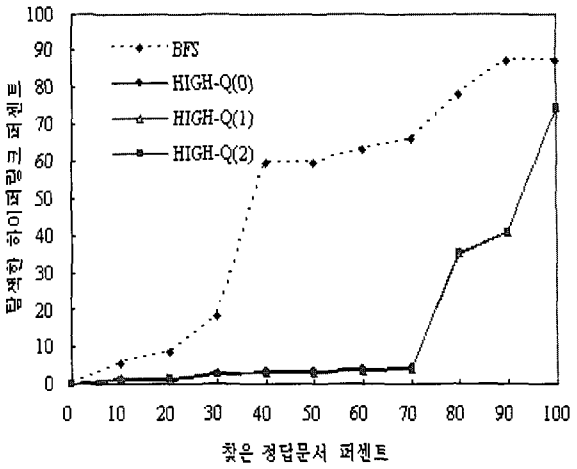
본 논문에서는 데이터 마이닝(data mining)이나 정보 추출(information extraction)과 관련된 출판물(publication)에 대한 웹 문서들을 찾는 것을 목적으로 실험을 수행하였으며, 이 때 수집한 웹 문서들에 존재하는 하이퍼링크들 중에서 pdf나 ps파일을 연결해주는 하이퍼링크와 상위노드로 회귀하는 링크들은 실험의 정확성을 위하여 제외하였다. 각 문서들에 존재하는 하이퍼링크의 평균수는 6개 정도로 나타났으며, 실험대상이 된 학부의 특성을 살펴본 결과 정답을 찾기 위해 탐색할 패스의 깊이는 약 7~8개를 넘지 않는 것으로 나타났다. 따라서 우리는 노드의 전진을 깊이 10으로 정하였다.

실험의 성능을 평가하기 위해서 입력 질의에 대한 정답 집합을 구성하고, 이 정답 집합을 얼마나 빨리 그리고 정확하게 찾는가에 대하여 조사하였다. 정답 집합은 벡터모델에 기반한 코사인 유사도를 이용하여 유사문서들을 추출하고 이들을 전문가들이 재검토한 결과물로 구성하였다.

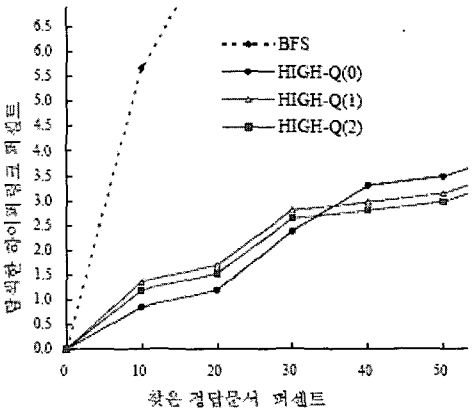
실험은 데이터 마이닝이나 정보추출과 관련된 홈페이지나 출판물을 검색하는 것을 목적으로 하였으며 나타난 하이퍼링크들을 순서대로 따라가는 넓이 우선 탐색(BFS)과 미래의 보상을 단계별로 고려한 Q-학습(HIGH-Q(0), HIGH-Q(1), HIGH-Q(2))를 비교 실험하였다. 이에 대한 실험 결과가 다음에 나타나있다. HIGH-Q(0)는 노드를 선택할 때 미래의 보상에 대한 고려없이 즉각적인 보상 값만을 이용하는 방법이고, HIGH-Q(1), HIGH-Q(2)는 각각 한 단계 혹은 두 단계의 할인된 미래의 보상 값을 고려한다는 것을 의미한다. 다음의 그림 2는 이에 대한 실험결과들을 보여주고 있다.

그림 2(a)는 정답문서집합을 찾기 위하여 탐색한 하이퍼링크의 비율을 나타낸다.

성능을 비교하기 위하여 정답문서의 80%가 찾아질 때까지의 탐색한 하이퍼링크의 수를 조사하였으며 구체적인 결과가 다음의 표 1에 나타나있다.



(a)



(b)

그림 2. 넓이우선탐색과 강화학습이용 탐색 방법의 비교

표 1. 80%의 정답문서를 찾을 때까지 탐색한 노드의 수

BFS	925
HIGH-Q(0)	418
HIGH-Q(1)	417
HIGH-Q(2)	412

표 1에서 정답을 찾기 위해 탐색한 페이지의 비율은 전체 페이지의 넓이 우선 탐색은 78.3%, 강화학습을 적용한 탐색의 경우에는 35.2%로 나타났다. 많은 하이퍼링크들을 따라가다 보면 결국 찾고자 하는 정답집합을 모든 탐색 방법이 찾을 수는 있다. 하지만 강화학습을 이용한 탐색 방법이 너비 우선 탐색 방법보다 약 2배 이상의 더욱 적은 하이퍼링크를 따라가고도 정확하게 원하는 정보를 찾을 수 있다는 것을 알 수 있었다. 이 때 탐색한 평균 에피소드의 수는 756개였다.

그림 2(b)는 미래의 보상을 고려한 강화학습을 적용한 탐색을 자세히 비교하기 위하여 위의 그림을 확대한 것이다. 이로부터 우리는 초기 30%의 웹페이지를 찾기까지는 HIGH-Q(0)가 나은 성능을 보이지만 그 이후부터는 HIGH-Q(2)가 HIGH-Q(1), HIGH-Q(0) 보다 나은 성능을 보이기 시작

하는 것을 알 수 있었다. 이는 미래의 보상 값을 고려하는 것이 탐색에 영향을 미침을 보여주는 것이다.

5. 결 론

본 논문은 웹 문서들을 더욱 빠르고 정확하게 탐색하기 위하여 강화학습을 이용한 탐색 알고리즘을 제안하였다. 특히 주제별 웹 탐색은 전 시간에 걸쳐 보상을 측정할 수 있고 지연된 보상을 갖는다는 것이 특징이며, 이득이 될 수 있는 미래의 보상을 모델링할 수 있다는 점이 장점이다.

제안한 HIGH-Q 학습 알고리즘이 주어진 환경에서 빠르고 효율적임을 보이기 위하여 정답을 찾기 위해 탐색하는 노드 수에 대한 실험을 수행하고 이를 범용적인 탐색 방법인 넓이 우선 탐색과 비교, 평가하였다. 실험 결과로부터 우리는 제안한 방법이 보다 더 정확하고 빠른 검색을 수행할 수 있었고, 미래의 할인된 보상을 이용하는 강화학습 이 탐색 페이지의 수를 줄이는데 효과적임을 알 수 있었다. 이는 웹 페이지의 수가 늘어날수록 더욱 효과적일 것이다. 다만, 미래의 보상 값을 고려할 때, 참조 단계를 무작정 늘이는 것은 좋은 방법은 아닐 것이다 참조하는 미래의 보상 값이 늘어날수록 그에 대한 부하가 또한 커지기 때문이다. 따라서 참조단계를 적절히 조정하는 연구가 이루어져야 할 것이다.

일반적인 강화학습 방법들의 경우 큰 상태공간에 대한 문제점이 발생한다. 큰 상태공간은 보다 많이 탐험을 해야 하며 현재 경험한 상태-행동 쌍의 Q값만을 갱신하므로 학습속도가 느려진다는 것을 의미한다. 따라서 룩업 테이블에 의한 표현 방식은 적절한 방법이라고 볼 수 없으며, 에이전트가 학습을 위하여 작은 행동 공간을 사용하는 방안에 대한 연구가 필요할 것이다. 또한 실험 대상이 된 컴퓨터공학부 홈페이지는 학부 소개나 행사와 관련된 문서와의 연결이 많았고 검색하고자 하는 논문자료들과 연결되지 않은 것이 문제점이었고 외부로 나가는 하이퍼링크에 대한 처리가 미흡했으므로 이에 대한 보완과 더불어 코퍼스의 확장이 필요하다.

참 고 문 헌

[1] 박찬건, 양성봉, "강화 학습에서의 탐색과 이용의 균형을 통한 범용적 온라인 Q-학습이 적용된 에이전트의 구현," 정보과학회 논문지(B), Vol. 30, No. 7, pp. 672-680, 2003.

[2] 정태진, 장병탁, "강화 학습을 이용한 웹 정보 검색," 정보과학회 제 28회 추계학술대회, Vol. 28, No. 2, pp. 94-96, 2001.

[3] C. J. Watkins and P. Dayan, "Technical note : Q-Learning," Machine Learning, 8, pp .279-292, 1992.

[4] F. Menczer, "ARACHNID: Adaptive retrieval agents choosing heuristic neighborhoods for information discovery," In *proceedings of 14th International Conference on Machine Learning*, pp. 227-235, 1997.

[5] H. Lieberman, "Letizia: An agent that assists web browsing," In *Proocedings of the International Joint Conference on Arti cial Intelligence (IJCAI95)*, pp. 924-929, 1995.

[6] J. Boyan, D. Freitag, and T. Joachimas, "A machine learning architecture for optimizing web search engines," In *proceedings of AAAI workshop on Internet-Based Information Systems*, pp. 1-8, 1996.

[7] J. Peng, and R. Williams, "Incremental multi-step Q-learning," *Machine Learning*, vol. 22, pp. 283-290, 1996.

[8] J. Rennie and A. McCallum, "Using Reinforcement Learning to Spider the Web Efficiently," In *proceedings of the 16th International Conference on Machine Learning(ICML-99)*, pp. 335-343, 1999.

[9] L. P. Kaelbling, "Learning in Embedded System," PhD thesis, Departmenr of Computer Science, Stanford University, 1990.

[10] R. Dearden, N. Friedman and S. Russell, "Bayesian Q-Learning," In *proceedings of AAA-98*, 1989.

[11] R. S. Sutton and A. G. Barto, *Reinforcement Learning : An Introduction*. The MIT Press, 1998.

[12] S. B. Thrun, "The role of exploration in learning control," *Handbook of Intelligent Control:Neural, Fussy and Adaptive Approaches*. 1992.

[13] T. Joachims, D. Freitag, and T. M. Mitchell. "A WebWatcher: A Tour Guide for the World Wide Web," In *Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence (IJCAI'97)*, pp. 770-777, 1997.

[14] T. M. Mitchell, *Machine Learning*, McGraw-Hill, 1997.

[15] M. Tan, Multi-agent reinforcement learning: Independent vs. cooperative agents. In *Proc. of the Tenth International Conf. on Machine Learning*, pp. 330.337, 1993.

저 자 소 개

임수연(SooYeon Lim)

제15권 1호 (2005년 2월호) 참조