KIPS Tr. Software and Data Eng.
Vol.7, No.3 pp.99~106 pISSN: 2287-5905

https://doi.org/10.3745/KTSDE.2018.7.3.99

Comparison of Performance Between Incremental and Batch Learning Method for Information Analysis of Cyber Surveillance and Reconnaissance

Gyeong-Il Shin† · Hosang Yoon†† · DongIl Shin††† · DongKyoo Shin†††

ABSTRACT

In the process of acquiring information through the cyber ISR (Intelligence Surveillance Reconnaissance) and research into the agent to help decision-making, periodic communication between the C&C (Command and Control) server and the agent may not be possible. In this case, we have studied how to effectively surveillance and reconnaissance. Due to the network configuration, agents planted on infiltrated computers can not communicate seamlessly with C&C servers. In this case, the agent continues to collect data continuously, and in order to analyze the collected data within a short time in When communication is possible with the C&C server, it can utilize limited resources and time to continue its mission without being discovered. This research shows the superiority of incremental learning method over batch method through experiments. At an experiment with the restricted memory of 500 mega bytes, incremental learning method shows 10 times decrease in learning time. But at an experiment with the reuse of incorrectly classified data, the required time for relearn takes twice more.

Keywords : Cyber ISR, Incremental Learning Method, Batch Learning Method, AdaBoost

사이버 감시정찰의 정보 분석에 적용되는 점진적 학습 방법과 일괄 학습 방법의 성능 비교

신 경 일† · 윤 호 상†† · 신 동 일††† · 신 동 규†††

요 약

사이버 감시정찰은 공개된 인터넷, 아군 및 적군 네트워크에서 정보를 획득한다. 사이버 ISR에서 에이전트를 활용하여 데이터를 수집하고, 수집한 데이터를 C&C 서버에 전송하여 수집한 데이터를 분석 한 후 해당 분석 결과를 이용하여 의사결정에 도움을 줄 수 있다. 하지만 네트워크 구성에 따라 침투한 컴퓨터에 심어진 에이전트와 외부 네트워크에 존재하는 C&C 서버 간 정기적인 통신이 불가능하게 되는 경우가 존재한다. 이때 에이전트는 C&C 서버와 통신이 재개되는 짧은 순간에 데이터를 C&C 서버에 전달하고, 이를 받은 C&C 서버는 수집한 데이터를 분석한 후 다시 에이전트에게 명령을 내려야한다. 따라서 해당 문제를 해결하기 위해서는 짧은 시간 내에 빠르게 학습이 가능하며, 학습 과정에서 많은 자원을 소모하지 않고도 학습할 수 있어야한다. 본 연구에서는 점진적 학습 방법을 일괄 학습 방법과 비교하는 실험을 통해 우수성을 보여주고 있다. 점진적 학습 방법을 사용한 실험에서는 500M 이하의 메모리 리소스로 제한된 환경에서 학습소요시간을 10배 이상 단축시키는 결과를 보여 주었으나, 잘못 분류된 데이터를 재사용하여 학습 모델을 개선하는 실험에서는 재학습에 소요되는 시간이 200% 이상 증가하는 문제점이 발견되었다.

키워드 : 사이버 ISR, 점진적 학습 방법, 일괄 학습 방법, AdaBoost

1. 서 론

사이버전은 합동 전력의 일부로서 전통적인 물리전을 지휘하는 지휘관이나 참모가 사이버공간에서 원하는 정보를 적시에 접속하고 사용할 수 있도록 사이버 공간에서 '기동의 자유를 제공(providing freedom of maneuver)'하고 사이버 공간 내에서 또는 사이버 공간을 통한 '전력을 투사(projection of power)' 할 수 있도록 하는 연구가 필요하다.

※ 본 연구는 방위사업청과 국방과학연구소의 지원으로 수행되었음 (UD160066BD).
※ 이 논문은 2017년도 한국정보처리학회 추계학술발표대회에서 "사이버 ISR에서의 점진적 학습 방법과 일괄 학습"의 제목으로 발표된 논문을 확장한 것임.
† 준 회 원 : 세종대학교 컴퓨터공학과 석사과정
†† 비 회 원 : 국방과학연구소 책임연구원
††† 종신회원 : 세종대학교 컴퓨터공학과 정교수
Manuscript Received : December 15, 2017
First Revision : January 16, 2018
Accepted : January 28, 2018
* Corresponding Author : DongIl Shin(dshin@sejong.ac.kr)

이와 같은 연구를 수행하기 위해서는 사이버공간에서의 정보수집 기술을 비롯하여 물리전과 사이버전을 통합하기 위한 사이버 지휘통제(Cyber Command and Control, C2) 체계, 능동적인 사이버방어/공세적인 대응기술 및 사이버 전투피해평가 기술에 관한 운영 개념연구 및 기초연구가 시급하다.

국가/국방차원에서 사이버공간의 전투에서 승리를 하기 위해서는 사이버공간 감시정찰 → 사이버공간 지휘통제 → 사이버공간 방어/공세적 대응 → 사이버공간 전투피해평가 → 환류(feed-back)/재대응하는 절차를 보장할 수 있도록 사이버 공간에서의 작전능력을 발휘케 하는 일련의 연구가 반드시 필요하고 본 논문은 특히 사이버 감시정찰에서 정보의 신속한 분석에 관련된 최근의 연구결과를 서술하고 있다.

지휘명령 결정에 소요되는 정보는 전투 시 우위를 달성할 수 있는 매우 중요한 요소로 아군 측의 전반적인 상황 데이터와 함께 상대방의 데이터를 얼마나 보유하고 있느냐에 따라서 전투의 승패가 갈릴 수 있다. 특히 전쟁의 영역이 사이버 공간으로 확대된 현대전에서 정보는 매우 중요해졌으며, 데이터를 수집하여 분석하는 전체 과정을 사이버 ISR (Intelligence Surveillance Reconnaissance) 이라 지칭한다[1]. 본 논문에서는 이러한 정보를 취득하기 위해 정보를 수집하는 단계와 최종 분석하는 단계에서 기계학습 및 딥러닝을 이용하여 지휘명령 결정에 도움을 주는 에이전트 모델을 제안했다.

사이버 ISR은 물리적인 감시/정찰을 기반으로 하여 정보를 생성하는 방법인 전통적인 정보 감시정찰에 사이버 공간이 추가된 개념으로 사용된다. 사이버 공간에서의 감시정찰은 공간의 특성상 공공기관, 민간 기관, 정부와 상업, 군대와 비군사적 구분이 모호하며 대부분의 데이터가 익명성을 보유하는 등의 특성이 있으므로, 사이버 공간에서 발생되는 데이터에 대해서 수집하고 분석할 수 있는 별도의 운영 시스템이 필요하다. 일반적으로 적군 측 정보를 수집할 때 적군의 네트워크에 있는 에이전트가 지휘통제(Command and Control, C&C) 서버를 통하여 아군 측의 명령을 받고 작전을 수행할 수 있다. 에이전트는 적군 측의 데이터를 수집하여 정기적으로 지휘통제 서버에 수집한 데이터를 전송해주고, 지휘통제 서버는 수집된 데이터를 분석하여 새로운 명령을 내리거나 수집대상물을 교체하는 과정을 수행하게 된다. 이러한 일련의 과정을 고려하여 고안된 것이 사이버 감시정찰 운영 프로세스로서 전체 구성은 Fig. 1과 같다.

하지만 일부 네트워크 구성상 지휘통제 서버와 에이전트 간 정기적 통신이 불가능 한 경우가 있으므로, 에이전트가 지속적으로 데이터를 수집하면서 지휘통제 서버와 통신이 재개되는 순간 수집한 데이터를 지휘통제 서버에 전송하고 이를 받은 지휘통제 서버는 수집된 데이터를 빠르게 분석하는 모델이 필요하다. 해당 분석모델에는 이미 학습이 완료된 모델로 새로운 데이터가 들어오면 완료된 모델에 새로운 데이터를 추가하여 학습 업데이트가 가능한 학습법이 필요하다. 추가적으로 지휘통제 서버에서는 한 가지 분석 모델이 아닌 여러 가지의 분석 모델을 동시에 빠른 시간 내에 분석해야하므로 최소한의 리소스를 이용하도록 고안되어야 한다.

본 논문은 이러한 조건을 만족시키기 위해 점진적 학습 방법을 이용하여 분석 에이전트를 학습시키는 방안을 제안하고 이에 대한 실험 결과를 서술한다. 본 논문은 일괄처리 학습 방법과 점진적 학습 방법을 이용하여 학습하는 두 가지 방법을 여러 가지 측면에서 비교한 결과를 보여주고 있다[2, 3].

기존 분석 모델에서는 더 좋은 성능을 얻기 위하여 정답을

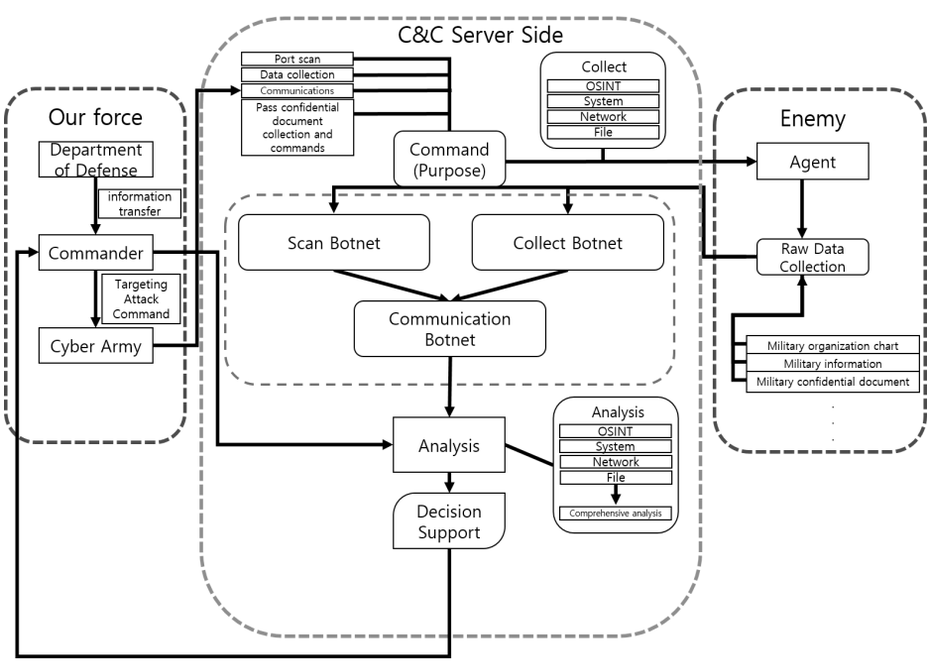

Fig. 1. Cyber Surveillance and Reconnaissance Operation Concepts

맞춘 데이터에 초점을 두고 분석을 한다. 하지만 본 연구에서 제안하는 아이디어는 잘못 분류된 데이터의 재사용에 대해 초점을 맞춰 모델을 개선하는 방향을 제안한다. 이는 모델이 정답을 맞힌 데이터에 가중치를 높게 부여하는 방법이 아닌 정답을 맞히지 못한 데이터에 가중치를 높게 부여하여 학습하는 방법으로 분석 모델이 왜 틀렸는지에 중점을 두고 분석하는 방법이다[4-6].

2. 관련 연구

점진적 학습 방법은 이용 가능한 자료들을 이용하며, 하나 이상의 개념 가설 형성을 하고 점차적으로 추가로 주어지는 예제들을 이용하여 가설을 개선한다. 현재 주어진 예와 반례로부터 지식을 생성하고 계속 새로운 예와 반례가 생길 때마다 점진적으로 현재의 지식을 수정하는 방향으로 진행되는 학습 방법이다[7]. 이는 인간의 개념 학습 방법과 매우 유사하며, 여러 개의 개념습득에 유용한 방법이다.

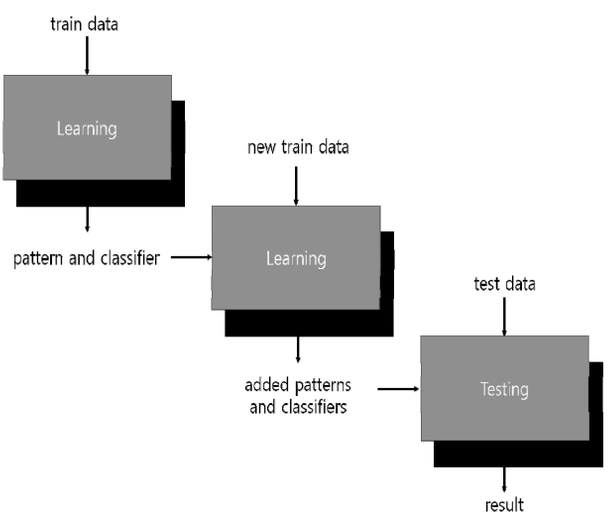

Fig. 2. Incremental Learning Method

점진적 학습 방법은 일괄 학습 방법과 달리 새로 들어오는 데이터를 학습할 때 기존의 학습이 끝난 모델에 이어서 학습을 이어서 하는 모델로 이전에 학습한 데이터를 제외하고 새로 생성된 데이터에 관해서만 학습을 진행한다. 그로 인하여 학습 시 사용되는 시간도 단축할 수 있으며, 사용되는 자원 또한 감소하게 된다[8].

일반적인 네트워크 데이터는 지속적으로 들어오게 되는 스트림형 데이터로 높은 수준의 정확성을 가지며, 실시간으로 탐지할 수 있는 기능이 요구되는 매우 처리하기 어려운 도메인에 속한다. 또한 기존의 침입탐지 시스템(IDS: Intrusion Detection System)의 일반적인 데이터 프로파일링 형태로는 지속적으로 변형되는 공격기법을 탐지하기는 더욱 어렵다. 따라서 최근 머신러닝 및 딥러닝을 이용한 탐지 시스템에 대한 연구가 많이 진행되고 있다[9, 10]. 특히 이러한 문제에는 KDD CUP 99 데이터가 사용된다.

먼저 기계학습 및 앙상블 기법 알고리즘을 적용한 Hasan's의 논문에서는 KDD CUP 99 데이터를 SVM과 Random Forest를 비교하는 실험을 다뤘는데, 실험결과 SVM의 성능이 +1.58% 약간 더 우수하지만 학습 경과 시간이 Random Forest에 비해 4배가량 많은 시간이 소요된다[11].

또한 딥러닝 기반의 알고리즘을 적용한 Aminato's의 논문에서는 KDD CUP 99 데이터를 인공 신경망(Artificial Neural Network)과 스택 오토 인코더(Stacked Auto Encoder)를 적용하여 모든 특징을 이용해 모델을 학습 시킨 것과 일부 특징만을 활용하여 모델을 학습시킨 결과 학습시간도 빠르며, 유사한 성능이 나오는걸 알 수 있었다[12]. Aminato's는 이러한 침입탐지 시스템 모델은 일반적인 침입탐지 시스템과 비교할만한 탐지율을 제공하며, 향후 연구로는 더 다양한 침입탐지 시스템 유형에 대해 실험할 예정이라 했다.

본 논문에서는 사이버 감시정찰 패킷에 대한 분석 데이터로 적합한 KDD CUP 99 데이터 셋을 이용하여 실험하였다. KDD CUP은 ACM에서 매년 열리는 데이터 마이닝 대회로 KDD CUP 99 데이터 셋은 KDD CUP 99년도에 쓰인 데이터이다. 해당 데이터는 네트워크 패킷 데이터이로, 앞에서 말 한 듯이 침입 탐지 시스템 관련 데이터 셋으로 자주 쓰이고 있다. 이는 주요 공격은 DoS(Denial of Service), R2L(unauthorized access from a remote machine), U2R(unauthorized access to local superuser privileges), Probe 총 4가지로 구별되며, Normal 라벨까지 총 5가지의 라벨이 존재한다. 또한 KDD Cup 학습 데이터에는 23개의 공격 유형이 포함되어 있고 평가 데이터는 학습 데이터에 없는 14개의 유형이 추가로 포함된다[13, 14].

3. 실 험

3.1 일괄 학습 방법 vs 점진적 학습 방법 비교

본 실험에서는 적군 측 네트워크의 구조에 의해 지휘통제 서버와 에이전트 간 정기적인 통신이 불가능하다는 가정 하에 실험하였다. 적군 컴퓨터에 침입한 에이전트가 지속적으로 데이터를 수집하며, 적군에게 발각되지 않도록 숨어 지내다가 지휘통제 서버와 통신이 재개되는 순간 에이전트는 지금까지 수집한 데이터를 지휘통제 서버로 전송을 한다. 이때 통신이 언제 끊길지 모르기에 지휘통제 서버에서는 짧은 시간 내에 빠르게 분석모델에 추가된 데이터를 학습을 하여 새로운 모델을 생성 및 업데이트한 후 에이전트에게 전송해주어야 한다. 또한 데이터들을 이용하여 한 개의 모델을 생성하는 경우가 아니라 다수의 모델을 생성해야할 가능성이 높기에 적은 리소스를 이용하여야 한다. 따라서 지휘통제 서버와의 통신이 재개되는 순간 빠르고 적은 리소스를 이용하여 데이터 분석을 시도하는 상황에 적합한 학습법을 찾기 위해 일괄 학습 방법

과 점진적 학습 방법을 비교하는 실험을 해보았다.

해당 실험은 멀티코어 CPU를 이용하여 학습 실험을 진행하였으며, 사용된 CPU는 Ryzen 7 1700X이며, 운영체제는 Windows 10 Pro 1708버전, 메모리는 64GB, Java 버전은 1.8.0_131, 사용한 에디터는 Eclipse, 사용한 기계학습 라이브러리는 MOA(Massive Online Analysis) 2016.04 버전을 이용하여 실험하였다. 그리고 실험에 사용된 데이터 셋은 KDD CUP 99 데이터의 학습 데이터와 평가 데이터를 이용하였으며, 두 개의 데이터 중 학습 데이터만을 가지고 학습을 해보고 두 번째로는 학습 데이터와 평가 데이터를 합쳐 학습하였다. 이는 데이터가 크면 클수록 더욱 성능의 차이가 확연하게 보이기에 학습 데이터와 평가 데이터를 합친 데이터를 생성하여 성능을 비교 해보았으며, 이와 동일하게 명확한 비교를 위하여 많은 리소스를 사용하는 트리 알고리즘을 이용하여 리소스의 사용량을 비교하였다. 가장 먼저 학습 시 사용되는 메모리양과 학습경과시간에 대해 비교를 하였다.

Table 1. Data Set Description

Data set	Number of data	Size of data
training	494,020	51.1MB
test	311,029	39.9MB
training + test	805,049	83.9MB

실험에 사용한 데이터는 본래 23개의 공격 유형으로 나누어져 있었다. 본 실험에서는 23개의 공격 유형 중 비슷한 유형끼리 묶기 위해 Kayacik's의 논문에서 각 공격을 범주 형태로 나눈 자료를 참고하여 상위 개념으로 표현하였다. 총 5가지의 상위개념으로 묶었으며, 이는 Normal, DoS, Probe, R2L, U2R이다.

먼저 일괄 학습 방법을 이용하는 알고리즘을 이용하여 실험을 해보았으며, 해당 실험에서 쓰인 알고리즘은 트리알고리즘의 하나인 의사결정 트리(Decision Tree)를 이용해보았다. 메모리양의 제한을 약 14545MB일 때 학습 데이터만을 학습할 경우 약 52.576초가 소요되었으며, 이때 총사용한 메모리의 양은 2884.5MB이다. 학습 데이터와 평가 데이터를 모두 학습한 경우는 앞의 시험과 동일하게 메모리양의 제한은 14545MB이었으며, 총 143.694초의 시간이 소요되었고, 총 사용된 메모리의 양은 4995MB이었다. 학습 데이터만 사용하였을 경우 초당 약 55MB를 사용하였고, 학습 데이터와 평가 데이터를 학습한 경우에는 초당 약 35MB를 사용하였다. 당연히 데이터가 커지면 커질수록 사용하는 메모리와 시간이 증가한다. 그렇다면 메모리를 1000MB 제한했을 경우에는 어떠한 결과가 나오는지 실험을 해보았다. 학습 데이터만을 학습시킬 때 937MB가 사용되었으며, 총 51.582초가 경과되었다. 학습 데이터와 평가 데이터를 합쳐 학습하였을 경우에는 heap 사이즈가 부족하여 학습을 하지 못하였다. 타겟 컴퓨터에 에이전트가 침투했을 경우 일괄 학습 방법을 이용하여 분

Table 2. Classify Attack Types by Category

Category	Attack type
Normal	normal
DoS	smurf
	neptune
	back
	teardrop
	pod
	land
Probe	satan
	ipsweep
	portsweep
	nmap
R2L	warezclient
	guess_passwd
	warezmaster
	imap
	ftp_write
	multihop
	phf
U2R	spy
	buffer_overflow
	rootkit
	loadmodule
	perl

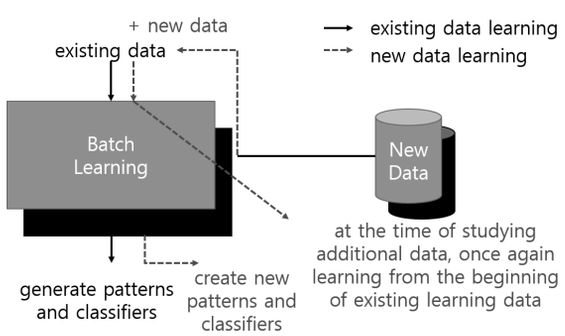

Fig. 3. Batch Learning Process

석을 할 때 메모리에 제한을 주지 않고 해당 학습을 진행하였다면 사용되는 메모리의 양이 너무 많아 적에게 발각될 수 있으며, 반대로 메모리를 제한하여 학습을 진행하면, 데이터가 너무 클 경우 학습 진행을 못하는 경우가 발생할 가능성이 높다. 이러한 이유로 제안한 사이버 감시정찰 모델에서는 일괄 학습 방법이 적합하지 않다고 판단했다. 그리하여 일괄 학습 방법의 문제점을 해결하기 위해 점진적 학습 방법을 적용해보았다.

먼저 일괄 학습 방법을 적용한 트리 알고리즘의 하나인 Hoeffding Tree를 사용하여 실험하였으며, 기존 ID3, C4.5과 같은 기존 의사 결정 트리 학습자의 저장소 제한 문제를 Hoeffding bound를 통해 해결할 수 있는 알고리즘으로 대용

량 데이터를 스트림으로 처리할 때 사용하게 된다. 앞서 의사결정 트리를 이용하여 실험한 것과 동일한 환경에서 실험을 진행하였다. 먼저 일괄 학습 방법으로 메모리양의 제한을 14545MB를 주었을 때 학습 경과시간은 18.541초가 소요되었으며 사용한 메모리의 양은 5295MB이다. 학습 데이터와과 평가 데이터를 합쳐 실험하였을 경우 학습 경과 시간은 33.192초이며, 사용한 메모리양은 4088MB이다.

두 번째로 점진적 학습 방법을 적용하여 Hoeffding Tree를 학습하는 실험해보았다. 메모리양의 제한을 14545MB로 주었을 경우 학습 데이터만을 학습하였을 때 경과 시간은 18.828초이며, 사용된 메모리양은 787MB이다. 학습 데이터와 평가 데이터를 합쳐 학습한 경우 경과 시간은 34.644초였으며 사용된 메모리양은 806MB이다.

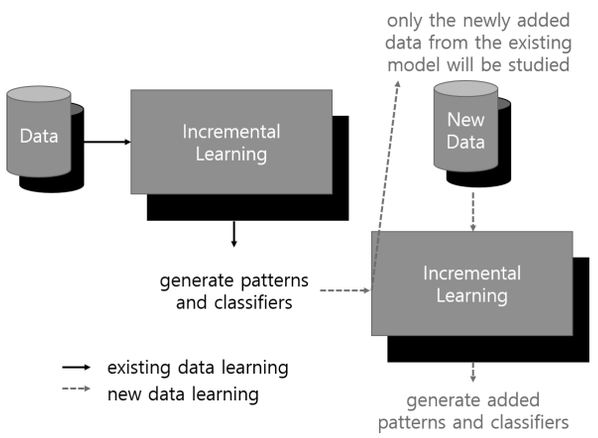

Fig. 4. Incremental Learning Process

Table 3. Comparison of Learning Elapsed Time and Amount of Memory Used by Model

Division	Model	Train Data	Limited amount of memory (MB)	Amount of memory used(MB)	Learning Time(sec)
Batch	Decision Tree	train	14545.0	2884.5	52.576
		all	14545.0	4995.0	143.694
		train	1000.0	937.0	51.582
		all	1000.0	Can not be measured due to lack of heap space	
	HoeffdingTree	train	14545.0	5295.0	18.541
		all	14545.0	4088.0	33.192
		train	1000.0	781.5	22.987
		all	1000.0	928.0	38.642
Incremental	HoeffdingTree	train	14545.0	787.0	18.828
		all	14545.0	806.0	34.644
		train	1000.0	761.5	18.951
		all	1000.0	778.0	29.554

Table 3를 보면 점진적 학습법을 사용한 Hoeffding Tree 모델이 가장 사용한 메모리의 양이 적었다. 또한 점진적 학습 방법을 사용한 경우 일괄학습을 사용한 경우보다 경과시간이 단축되었으며 이는 점진적 학습 방법이 더 적은 메모리를 사용하였는데도 학습 시간이 줄어든 것을 볼 수 있다.

Table 3에서 볼 수 있듯이 일반적인 의사결정 트리는 메모리가 1000MB로 제한된 경우에 메모리가 부족하여 실행이 불가능하였다. 본 논문은 추가적인 실험으로 얼마나 제한된 메모리상황에서도 Hoeffding Tree 모델 학습이 가능한지 알아보기 위해 매우 적은 메모리양을 이용하여 실험을 하였다.

먼저 학습 할 분석 모델은 Hoeffding Tree를 이용하였으며, 학습 데이터는 학습 데이터와 평가 데이터를 합친 데이터를 이용하였다. 그리고 이전 실험에서 제한 두었던 메모리양의 절반인 500MB부터 실험을 하였다.

Table 4. Comparison of Hoeffding Tree Model Learning with Very Limited Amount of Memory

Division	Model	Train data	Limited amount of memory (MB)	Learning Time(sec)
Batch	Hoeffding Tree	all	500	164.166
		all	400	469.408
		all	300	Can not be measured due to lack of heap space
		all	200	
		all	100	
Incremental	Hoeffding Tree	all	500	30.871
		all	400	31.340
		all	300	32.160
		all	200	31.477
		all	100	31.670

Table 4의 일괄 학습 방법을 적용하여 학습한 결과를 보면 메모리 제한이 500MB일 경우 경과 시간은 164.166초로 1,000MB일 경우 38.642초에 비해 상당히 많은 시간이 지체된 것을 볼 수 있다. 또한 메모리 제한이 300MB일 경우는 힙 공간 부족으로 인하여 실행이 불가능하다. 반면 점진적 학습 방법을 적용하여 학습한 결과를 보면 메모리 제한이 500MB인 경우 30.871초로 메모리 제한이 1000MB인 경우와 거의 유사한 학습시간을 보이며, 메모리 제한을 300MB으로 제한을 주어도 학습이 가능하였다. 이후 200MB, 100MB를 추가로 메모리 제한을 주어 실험한 결과 모두 학습이 가능한 것을 알 수 있었다. 100MB로 메모리 제한을 주었을 때 학습 시간은 31.670초로 매우 빠른 속도를 보여주었으며, 이는 일괄 학습 방법의 500MB로 메모리 제한을 주었을 경우보다 월등하게 빠른 속도를 보여준다.

이처럼 빠른 속도와 적은 리소스를 이용하는 점진적 학습 방법이 유용하지만 생성된 모델의 성능이 일괄 학습 방법을 적용하여 생성된 모델에 비해 좋지 않다면 이는 해당 문제에

적용하기에는 부적합한 방법이다. 따라서 성능을 검증하기 위하여 각 학습 방법을 적용하여 학습 데이터를 학습하여 모델을 생성 한 후 평가 데이터를 이용하여 모델의 성능을 평가하여 보았다. 두 학습 방법 비교 실험에 쓰인 데이터 모두 학습 데이터로는 이전 실험과 동일하게 KDD 99 학습데이터를 이용하였으며, 평가 데이터 또한 동일하게 KDD 99 평가 데이터를 이용하였다. 또한 일괄 학습과 점진적 학습 모두 동일하게 메모리 제한을 약 16기가로 설정했다.

Table 5. Model Performance Comparison Using Hoeffding Tree

Division	TP Rate	FP Rate	F-Measure	ROC Curve Area
Batch	76.8%	11.1%	0.799	0.864
Incremental	76.9%	11.2%	0.800	0.865

Table 5은 Hoeffding Tree를 이용하여 일괄 학습 방법과 점진적 학습 방법을 통해 생성된 모델의 성능을 비교하는 표이다. Table 5의 내용을 보면 일괄 학습 방법과 점진적 학습 방법을 통해 생성된 모델의 성능은 거의 유사하다고 볼 수 있다. 정답을 얼마나 잘 분류하였는지를 뜻하는 TP Rate의 경우 점진적 학습 방법이 0.1% 더 우수한 성능을 보였으며, 잘못 분류한 경우가 얼마나 많은 지를 뜻하는 FP Rate의 경우는 11.1%로 일괄학습 방법이 우수하였다. F-Measure 및 ROC Curve Area의 경우는 모두 점진적 학습 방법이 0.001 우수한 것을 알 수 있었다.

결과적으로 해당 문제에서 점진적 학습 방법은 일괄 학습 방법에 비해 빠른 속도로 학습이 가능하며, 매우 적은 리소스를 이용하지만 생성된 분류기의 성능은 일괄 학습 방법과 매우 유사한 성능을 보이는 것을 알 수 있었으며, 이는 해당 문제에는 점진적 학습 알고리즘을 적용하는 것이 매우 적합하다는 것을 알 수 있다.

3.2 잘못 분류된 데이터의 재사용에 대한 성능 비교

추가적인 실험으로 잘못 분류된 데이터를 재사용하여 모델의 성능을 개선하는 방법을 적용한 것과 적용하지 않은 것을 성능 비교하여 실험하였다. 실험에 사용된 알고리즘은 의사결정 트리와 앙상블 기법 중 하나인 AdaBoost이며, 데이터 셋은 위와 동일하게 KDD CUP 99 데이터 셋을 이용하였다. 실험방법으로는 일반적인 의사결정 트리와 AdaBoost를 적용한 의사결정 트리의 분석 모델을 생성한 후 평가 데이터를 이용하여 성능을 측정해보았으며, 최종적으로 제안한 방법과 일반적인 방법을 비교했다.

Table 6. Comparison of Proposed Method and General Model Performance

Model	TP Rate	FP Rate	F-Measure	ROC Area	Learning Time
Decision Tree	73.8%	2.0%	0.787	0.882	47.99sec
AdaBoost (Decision Tree)	74.5%	2.2%	0.794	0.955	1000.36sec

실험 결과 일반적인 의사결정 트리에 비해 잘못 분류된 데이터의 재사용에 초점을 맞추어 모델을 개선하는 방법으로 AdaBoost 알고리즘을 적용한 의사결정 트리가 높은 성능을 보여주었다. 하지만 ROC Area를 제외하고 모든 결과가 일반적인 의사결정 트리와 거의 유사하다. 또한 학습시간 같은 경우는 훨씬 오랜 시간이 소요되기에 통신이 재개된 순간 빠르게 학습을 요구하는 본 문제에는 적합하지 않은 것으로 분석되었다.

4. 결 론

본 논문에서는 두 가지 실험을 하였다. 먼저 적 네트워크에 침투한 에이전트가 지휘통제 서버와의 정기적 통신이 불가능할 때 통신이 재개되는 순간 지휘통제 서버에서 빠른 시간 내에 적은 리소스를 사용하여 데이터를 학습하는 방법으로 적합한 학습 방법을 찾기 위하여 일괄 학습 방법과 점진적 학습 방법 두 가지를 비교하여 보았으며, 두 번째로는 잘못 분류한 데이터를 재사용한 학습 결과와 일반적인 학습 결과의 성능을 비교했다.

첫 번째 실험 결과 점진적 학습 방법은 일괄 학습 방법에 비해 적은 메모리를 사용하여 더 적은 학습 시간이 소요된다는 것을 알 수 있었다. 또한 매우 적은 자원을 활용할 때 일괄 학습 방법은 자원의 부족으로 인하여 학습이 진행되지 않는 반면 점진적 학습경우는 매우 작은 자원을 활용해서 학습이 가능하며, 일괄 처리에 비해 매우 짧은 시간 내에 학습을 완료하는 것을 알 수 있었다. 그리고 각 방법 및 평가 데이터를 적용하여 분류기를 생성하였을 때, 일괄 학습과 점진적 학습 방법은 유사한 성능을 보여주었다. 결과적으로 점진적 학습 방법은 일괄 학습 방법보다 적은 자원을 소모하지만 빠른 학습 속도를 보여주며, 생성된 분류기의 성능은 유사하기에 해당 문제에 적합한 학습 방법이다.

두 번째 실험인 잘못 분류한 데이터를 재사용한 학습에서는 일반적인 학습에 비해서 제안한 방법이 미세하지만 더 좋은 결과를 보여주었다. 하지만 학습시간이 더 소요되는 결과를 얻었다.

이러한 실험 결과를 통해 사이버 감시정찰의 정보 분석 모델에는 점진적 학습 방법을 사용하는 것이 월등하게 우수하다는 것을 알 수 있었으며, 잘못 분류된 데이터의 재사용한 학습 모델은 약간의 정확도 상승의 결과를 보여주었지만 학습 시간이 오래 걸리기에 해당 도메인에는 적합하지 않은 방법이라는 것을 알 수 있었다. 해당 실험 결과에는 기재하지 않았지만 의사 결정 트리의 종류가 아닌 나이브 베이즈를 이용하였을 경우에도 100MB의 메모리 제한을 설정했을 때 일괄 학습 방법은 학습을 진행하지 못하지만 점진적 학습 방법은 학습을 진행할 수 있는 것을 알 수 있었다. 따라서 점진적

학습 방법은 결정 트리 종류 알고리즘 이외에도 적은 리소스를 사용하는 것을 알 수 있었으며, 제한된 환경에서 사용하기 적합한 방법인 것을 알 수 있다.

Fig. 1은 현재 연구 중인 사이버 감시 정찰 개념모델로 내부에 수많은 지능적 수집 기법 및 분석 기법들이 필요로 하다. 따라서 향후 연구로는 연구 중인 사이버 감시 정찰에서 필요로 한 지능적 수집 기법 및 분석 기법들을 연구할 예정이며, 이번 연구에서 연구된 점진적 학습 방법을 실제로 지능적 수집 기법 및 분석 기법에 적용하여 지능적이며 효율적인 사이버 감시 정찰 모델을 연구할 계획이다.

References

[1] Matthew M. Hurley, "For and from cyberspace: Conceptualizing cyber intelligence, surveillance, and reconnaissance," *Air & Space Power Journal*, Vol.26, No.6, pp.12-33, 2012.

[2] Hey-Jung Baek and Young-Tack Park, "The Study on Improvement of Cohesion of Clustering in Incremental Concept Learning," *The KIPS Transactions: Part B*, Vol.10, No.3, pp.297-304, 2003.

[3] P. Fuangkhon and T. Tanprasert, "An incremental learning algorithm for supervised neural network with contour preserving classification," *Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, 2009. ECTI-CON 2009. 6th International Conference on*, Vol.2, pp.740-743, 2009.

[4] Y. Freund, and R. E. Schapire, "Experiments with a new boosting algorithm," *Icml*, Vol.96, pp.148-156, 1996.

[5] W., Hu, W., Hu, and S. Maybank, "Adaboost-based algorithm for network intrusion detection," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, Vol.38, No.2, pp.577-583, 2008.

[6] T. G. Dietterich, "Ensemble methods in machine learning," *Multiple Classifier Systems*, Vol.1857 pp.1-15, 2000.

[7] A. Bifet, G. Holmes, R. Kirkby, and B. Pfahringer, "Moa: Massive online analysis," *Journal of Machine Learning Research*, Vol.11, pp.1601-1604, 2010.

[8] R. R. Ade and P. R. Deshmukh, "Methods for incremental learning: a survey," *International Journal of Data Mining & Knowledge Management Process*, Vol.3, No.4, pp.119-125, 2013.

[9] G. I. Shin, D. I. Shin, D. K. Shin, and H. S. Yooun, "An Comparative Research of the Detection Rate of Intrusion Detection System Algorithms," in *Proceedings of the Korea Information Processing Society Review 2017 Spring Conference*, Vol.24, No.1, pp.223-226, 2017.

[10] H. J. Ji, D. K. Shin, D. I. Shin, Y. H. Kim, and D. H. Kim, "A Study on comparison of KDD CUP 99 and NSL-KDD using artificial neural network," in *Proceedings of the Korea Information Processing Society Review 2017 Spring Conference*, Vol.24, No.1, pp.211-213, 2017.

[11] M. A. M. Hasan, M. Nasser, B. Pal, and S. Ahmad, "Support vector machine and random forest modeling for intrusion detection system (IDS)," *Journal of Intelligent Learning Systems and Applications*, Vol.6, No.1, pp.45-52, 2014.

[12] M. E. Aminanto and K. Kim, "Deep learning-based feature selection for intrusion detection system in transport layer," in *Proceedings of the Korea Institutes of Information Security and Cryptology Conference 2016 Summer*, Vol.26, No.1, pp.740-743, 2016.

[13] A. A. Olusola, A. S. Oladele, and D. O. Abosede, "Analysis of KDD'99 intrusion detection dataset for selection of relevance features," *Proceedings of the World Congress on Engineering and Computer Science*, Vol.1, pp.20-22, 2010.

[14] H. G. Kayacik, A. N. Zincir-Heywood, and M. I. Heywood, "Selecting features for intrusion detection: A feature relevance analysis on KDD 99 intrusion detection datasets," in *Proceedings of the Third Annual Conference on Privacy, Security and Trust*, 2005.

신 경 일

http://orcid.org/0000-0001-6498-0480
e-mail : sgi@gce.sejong.ac.kr
2017년 조선대학교 컴퓨터공학과(공학사)
2017년~현 재 세종대학교 컴퓨터공학과 석사과정
관심분야 : 기계학습, 딥러닝

윤 호 상

http://orcid.org/0000-0003-2723-3376
e-mail : yun_hosang@add.re.kr
1987년 고려대학교 수학과(학사)
1990년 고려대학교 전산학과(석사)
2003년 한국과학기술원(전산학박사)
1990년~1992년 한국국방연구원 연구원
1993년~1998년 국방정보체계연구소 선임연구원
1999년~현 재 국방과학연구소 책임연구원
2010년~현 재 국방과학연구소 2본부 3부 4팀장
관심분야 : 사이버전 침입탐지 기술, 사이버전 능동대응 기술

신 동 일

http://orcid.org/0000-0002-8621-715X
e-mail : dshin@sejong.ac.kr
1986년 연세대학교 전산과학과(학사)
1992년 Washington State University, Computer Science, M.S
1997년 University of North Texas, Computer Science, Ph.D.
1997년~1998년 시스템공학연구소, 선임연구원
1998년~현 재 세종대학교 컴퓨터공학과 정교수
관심분야 : 기계학습(딥러닝), HCI

신 동 규

http://orcid.org/0000-0002-2665-3339
e-mail : shindk@sejong.ac.kr
1986년 서울대학교 계산통계학과(학사)
1992년 Illinois Institute of Technology 컴퓨터과학과(석사)
1997년 Texas A&M University 컴퓨터공학과(박사)
1986년~1991년 한국국방연구원 전산체계연구부 연구원
1997년~1998년 현대전자 멀티미디어연구소 책임연구원
1998년~현 재 세종대학교 컴퓨터공학과 정교수
관심분야 : 상황인식 미들웨어, 정보 보안, 데이터마이닝