한국빅데이터학회지
제7권 제1호, 2022, pp. 113-124 https://doi.org/10.36498/kbigdt.2022.7.1.113

데이터센터 장애 예방을 위한 인프라 이상징후 분석: RRCF와 Prophet Ensemble 분석 기반

Infrastructure Anomaly Analysis for Data-center Failure Prevention: Based on RRCF and Prophet Ensemble Analysis

신현종 · 김성근 · 천병환 · 진경복 · 양승정†

한화시스템 IDC운영혁신팀

요 약

데이터센터의 장애 예방을 위해 머신러닝과 빅데이터를 활용한 다양한 방법들이 적용되어 왔다. 그러나 개별 장비 기반의 성능지표를 참조하거나, 인프라 운영환경을 고려하지 않은 접근방법으로 실제 활용되는 데에는 많은 한계가 있었다. 이에 본 연구에서는 개별 인프라 장비들의 성능지표를 통합 모니터링하며, 다양한 장비들의 성능지표를 구간화, 등급화 하여 단일수치화를 진행한다. 인프라 운영에 대한 경험치 기반으로 데이터 전처리를 수행하며, RRCF(Robust Random Cut Forest)분석과 Prophet 분석 모델을 앙상블하여 이상징후 검출에 신뢰도 있는 분석결과를 도출하였다. 데이터센터 내 운영담당자들의 접근을 용이하게 하기 위해 장애분석시스템을 구현하여 데이터센터 장애의 선제 대응과 적정한 튜닝시점을 제시할 수 있다.

■ 중심어 : IDC장애분석, 인프라 장애 예방, RRCF, Prophet, 이상징후 감지, Anomaly Detection, 통합 모니터링

Abstract

Various methods using machine learning and big data have been applied to prevent failures in Data Centers. However, there are many limitations to referencing individual equipment-based performance indicators or to being practically utilized as an approach that does not consider the infrastructure operating environment. In this study, the performance indicators of individual infrastructure equipment are integrated monitoring and the performance indicators of various equipment are segmented and graded to make a single numerical value. Data pre-processing based on experience in infrastructure operation. And an ensemble of RRCF (Robust Random Cut Forest) analysis and Prophet analysis model led to reliable analysis results in detecting anomalies. A failure analysis system was implemented to facilitate the use of Data Center operators. It can provide a preemptive response to Data Center failures and an appropriate tuning time.

■ Keyword : IDC Integrated Monitoring System, RRCF, Prophet, Anomaly Detection, Infrastructure failure prevention

2022년 05월 12일 접수; 2022년 05월 27일 수정본 접수; 2022년 05월 27일 게재 확정.
† 교신저자 (yeod0301@hanwha.com)

Ⅰ. 서 론

데이터센터 내에서 발생하는 인프라 장애를 예방하기 위해서 설비통합 모니터링 시스템 연구[11], 실시간 모니터링을 통한 장애예측 시스템[10], 빅데이터 분석 기반의 시계열 분석과 머신러닝 기법을 활용한 장애 예측[17] 등에 대한 다양한 접근이 시도되었고, 개발/소개되었다. 기존 접근방법에서는 데이터센터에서 발생하는 인프라 장애를 사후 즉시 조치가 아닌, 예측을 통한 사전 예방에 초점을 두고 있으며, 빅데이터와 머신러닝 기법을 활용하여 예측값을 생성하거나, 이상패턴에 대한 유무를 분석하여 선제적인 인프라 장애 예측이 가능함을 설명하고 있다.

이는 서버나 데이터베이스 등에 해당하는 각 인프라 장비로부터 CPU사용량, 메모리 사용량, 동시접속자 수, WAS 응답시간, DB Active Session 수 등 해당 인프라의 개별 성능지표 데이터들을 수집하여 장애 예측분석에 활용하는 것을 기본으로 하고 있다.

장애예측 방법에는 장애 데이터를 분석하여 예측값을 생성하는 것과 이상패턴을 학습하여 예측을 하는 방법이 있다.

빅데이터와 머신러닝 기법을 활용하여 예측값을 생성한 뒤 장애 발생 여부를 예측하는 방법은, 인프라 성능지표들을 활용하여 시계열 분석 등을 통해 특정 시간에서의 성능지표 값을 미리 알아낼 수 있는 것을 말하며, 이상패턴에 대한 분석 방법은 기존(또는 과거)에 인프라 시스템이 정상 또는 비정상이었을 때의 성능지표 패턴을 학습하여 이후에 보여 지는 성능지표의 패턴이 정상 또는 비정상인지에 대한 여부를 판단하게 된다.

그러나 실제로 데이터센터의 다양한 인프라에서 발생하게 되는 장애는 단일 인프라 성능 데이터에 의한 경우보다 훨씬 복잡하다. 또한 다양한 이벤트로 인해 발생하는 경우가 대부분이어서 단일수치 형식의 분석에는 한계가 있으며, 장애 예방에 대한 오류가 발생하게 된다. 그리고 인프라 성능지표의 정상과 비정상에 대한 패턴이 일정하지 않아 학습에 한계가 있으며, 이러한 문제를 해결하기 위해서 Critical한 상황과 관련된 항목 또는 지표 중심으로만 학습하거나 모니터링 하게 되어, 임계수치 결정이 어려운 항목은 분석 과정 중에 개별 분석하게 되는 경우가 발생하게 된다. 결과적으로 임계치 기반의 모니터링과 분석이 가능한 항목 위주로만 분석이 이뤄지게 되는 것이다.

기본적으로 데이터센터에서 운영하는 인프라는 업무 및 관련 시스템 종류가 상당히 다양하여 기존의 단순 성능지표를 이용한 예측 분석의 개념을 활용해서는 효율적으로 장애 예방을 할 수가 없다.

이에 본 연구에서는 실제 운영담당자들의 운영노하우를 반영하여 인프라 성능지표들을 다양한 관점으로 해석하여 데이터를 확장하고, 각 데이터 그룹별로 등급화 하여 수치를 단일화 하는 과정을 반복하여 성능 지표 데이터 전처리를 진행하였다.

성능지표 전처리를 위해서, 각 인프라의 성능지표 대상을 정의하고, 각 성능지표별로 운영업무현황에 맞게 해석한 뒤, 업무서비스 상태를 대변하는 복잡한 데이터를 단일 데이터로 변환한다.

결과에 대한 신뢰도를 높이기 위해 단일 학습 모델이 아닌, RRCF와 Prophet 분석 방법을 Ensemble하였다. 두 개의 모델에서 모두 이상치라고 판단하게 될 경우 최종적으로 장애 가능성이 높다고 판단하게 된다.

Ⅱ. 연구 방법

데이터센터 내 인프라 장비의 성능지표를 활용한 장애 예방 시스템을 구현을 위한 타당성 및

기술구현 방안 선정을 위해서 확증적 데이터분석 프로세스를 참조하였다.

단계1) 문제 정의 : 현 상황의 명확한 정의와 해결을 위한 방안 제시

단계2) 데이터 수집 : 장애 분석을 위한 대상 선정과 분석 데이터 수집

단계 3) 데이터 전처리 : 정의된 문제의 명확한 해결과 수집된 인프라 데이터의 운영 현황에 맞게 전처리 수행. 즉, 인프라 모니터링 수치를 다양한 관점으로 분석하여 데이터를 확장하고 각 데이터 그룹별로 등급화하여 성능 수치 데이터 전처리 진행

단계 4) 데이터 모델링 : 업무 서비스별 특성에 맞게 전처리된 데이터를 학습모델링

단계 5) 시각화 및 탐색 : 최종 인프라 운영자가 용이하게 사용할 수 있도록 다양한 각도로 시각화된 화면 서비스 제공

2.1 문제 정의

기존 데이터센터 내 인프라 장비의 장애관리를 위해서는 업무별, 장비별 임계치 관리 위주로 장애 예측 분석을 시도해 왔었다.

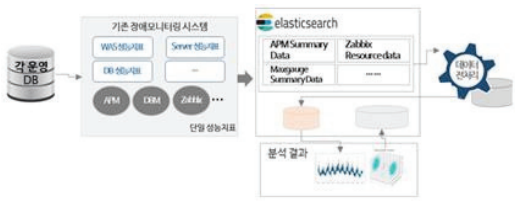
데이터센터의 운영 관점에서 살펴보면, 운영해야 하는 인프라 장비의 다양화와 이에 따른 모니터링과 업무의 복잡도가 증가하고 있으나, 이에 반해 현 상황은 각 인프라 요소별 분석에 국한되어 있어 활용에 한계가 있다. 또한, 실제 운영에 따른 인프라 관련 이벤트 및 성능 수치값의 종류가 다양하여 단일 수치 형식의 분석만으로는 장애현상에 대한 명확한 분석과 해석이 어렵다. 이를 해결하기 위해 Critical한 상황과 관련된 항목과 성능지표를 정의한 뒤 그 기준으로 분석을 하게 되며, 이는 그 외의 항목에 대한 대응을 더욱 어렵게 만드는 원인이 되기도 한다. 인프라 운영 환경에 따라 발생하게 되는 현상이 통합적으로 적용되는 데에 한계가 있으며, 변동폭이 매우 크거나, 일정 패턴이 없는 수치의 경우에는 적정 임계치 판단이 용이하지 않게 된다. 같은 인프라 장비라 하더라도, 업무와 관련 시스템에 따라 발생하는 현상이 다르며, 이에 따른 사전 정의가 되어 있지 않는다면 기존의 단순 수치 값의 변동 분석만으로는 이슈 여부 판단 및 유의미한 결과를 도출하는 데에 한계가 따른다.

이를 해결하기 위해서 우선, 인프라 운영 지식(운영 경험치) 기반으로 성능지표 전처리 대상을 선별한다. 기존의 수치 데이터를 기반으로 2차 활용이 가능한 이벤트를 생성하여 활용한다.

또한, 각 업무 인프라별로 존재하는 다양한 수치(약 90여종 이상)를 단일화 규칙에 따라 단일 데이터로 변환하여 단일화 수치 패턴을 분석하게 된다.

2.2 데이터 수집

본 연구에서 배경이 되는 데이터센터의 경우, 기존에 장애현황에 대한 실시간 모니터링 서비스를 제공하고 있으며, 제공되는 성능 데이터는 운영 인프라에서 발생하는 기본 운영 로그데이터와 성격이 다르다. 각 인프라 운영 모니터링 시스템에서 제공하는 Summary Data를 1차 가공하여 활용하게 되며, 이에 대한 데이터 수집 개념도는 (그림 1)에서 설명하고 있다.



〈그림 1〉 데이터 수집 개념도

다음 <표 1>은 Elasticsearch Index별 데이터 명세의 일부를 보여주고 있다.

〈표 1〉 Elasticsearch Index 별 데이터 명세 (예시)

name	field name	expected value	os
CPU 사용률	log_type	zabbix_metric	
	type	os_zabbix_cpu_usage	
	server_name	[hostname]	
	@timestamp	[데이터 시점 날짜]	
	created_mil	[데이터 시점 날짜 millisecondes]	
	os_cpu	[전체 CPU 사용률]	Linux AIX HP-UX Solaris Windows
	cpu_user_time	[user CPU 사용률]	
	cpu_idle_time	[Idle CPU 사용률]	
	cpu_system_time	[System CPU 사용률]	
	cpu_steal_time	[기타 Linux CPU 사용률]	Linux
	cpu_softirq_time		
	cpu_nice_time		
	cpu_interrupt_time		
Active Session	log_type	active sessions	
	server_name	[hostname]	
	instance_name	[instance_name]	
	server_instance_name	/[serer_name]/[instance_name]	
	@timestamp	[데이터시점 날짜]	
	value	value	
...	...	...	...

2.3 데이터 전처리

성능데이터 전처리를 위해 주요 모니터링 지표<표 2>와 비활용 지표<표 3>를 선정한다.

주요 모니터링 지표란, 각 인프라 요소별로 잘 알려진 지표로 실시간 모니터링이 되는 지표 수치들을 말한다. 예를 들어 서버의 CPU, Memory, DB의 Lock count, WAS의 Active Service 등을 말하며 인프라 장애 모니터링뿐만 아니라 임계치 기반의 장애 분석을 진행할 때 기본적으로 활용이 되는 지표들이다.

반면, 비활용 지표는 직관적인 해석이 어렵기 때문에 실시간으로 모니터링 되지 않는 지표를 말한다. 실시간 모니터링이 되지 않고 있으나, 장애나 이슈가 발생했을 경우 다른 지표 항목들과 교차 해석이 필요한 지표들로 장애의 주요 원인이 되기도 한다. 예를 들어, 서버의 Wait I/O, WAS의 GC Time, DB의 Sort Row 등이 비활용 지표에 해당된다.

〈표 2〉 주요 모니터링 지표(예)

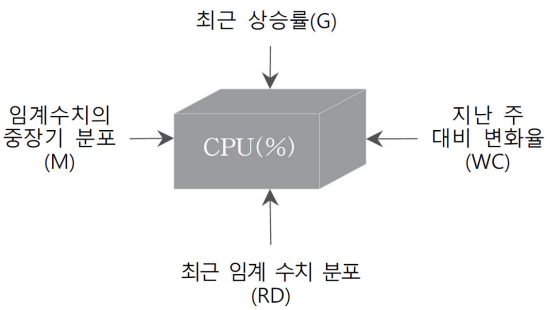
그룹	성능수치
Server	CPU, Memory, Disk...
WAS	Active Service, Response Time, ...
DB	Tablespace, Active Session, Lock Count ...
...	...

〈표 3〉 비활용 지표(예)

그룹	성능수치
Server	Wait I/O, Socket 등
WAS	GC Time, 동시사용자 등
DB	Hard Parse, Execute Count, Sort Row 등
...	...

선정된 성능지표들은 인프라 운영에 기본이 되는 가장 중요한 항목들이며, 장애 발생 시 가장 영향을 많이 주는 지표들이다.

개별 성능수치들을 4가지 관점으로 분석하여 데이터를 확장한다. 다음 (그림 2)는 운영 경험치 기반의 4가지 관점을 나타내고 있다. 대부분의 성능수치는 발생하는 시점의 현재값 자체만으로 장애나 이슈 여부를 판단하는 데에 한계가 있으므로 운영자들은 기존의 패턴을 비교하며 이상 유무에 대해서 판단하게 된다. 이때, 기존 운영 시 발생하는 일시적인 특이사항(일시적인 성능 수치의 상승/하락)에 대해서 이상 유무를 판단하



〈그림 2〉 4가지 관점의 개별수치 분석 (CPU사례)

기 위해서는 일정 시간이 반영된 수치의 흐름, 또는 과거 시점(동일 요일/유사한 시간대)의 성능수치 흐름 패턴과의 차이점을 살펴볼 필요가 있다. (그림 2)의 4가지 관점 중 최근 임계수치 분포(RD)는 최근 5분간의 데이터 분포를 설명하고 있다. 최근 임계 수치 분포는 일시적으로 수치의 상승/하락을 자주 보일 수 있으므로 민감하게 대응하지 않으며 최근 5분 데이터의 분포 중 상위 70% 값을 추출하여 적용한다. 이는 시스템의 성능 수치를 반영하는 평균, 최소값, 최대값 보다는 시스템의 전반적인 상태를 효과적으로 반영하다고 판단하여 백분위수로 적용하게 된다.

운영자들은 지표의 수치가 평소와 다른 값으로 10분 이상 유지되는 경우, 이상이 있다고 체크하며 인프라 상태에 변화가 발생했다고 판단한다. 이러한 현상을 반영하기 위해서 임계수치의 중장기 분포(M) 관점에서 최근 10분 데이터의 분포를 추출하여 1부터 10까지 레벨값을 부여한다.

최근 상승률(G)은 수치의 기울기를 보는 것으로, 일시적으로 상승하게 되는 수치에 대한 해석을 하고자 함이다.

지난 주 대비 변화율(WC)는 지난 주 같은 요일, 동일 시간대와 수치를 비교하게 되며, 최근 10분의 평균과 지난 주 동일 시간대의 평균 수치의 비율을 비교하여 수치구간에 레벨값을 부여한다.

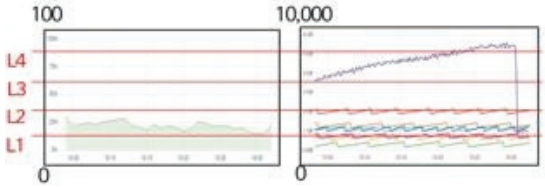
개별 성능수치들을 1)최근 임계 수치 분포, 2)임계수치의 중장기 분포, 3)최근 상승률, 4)지난 주 대비 변화율 등을 활용해 운영환경에 맞게 조정한다. 각 관점에 대해서 수치의 Level을 10가지로 정의한다 [표 4].

〈표 4〉 성능수치 Level 처리 (예시)

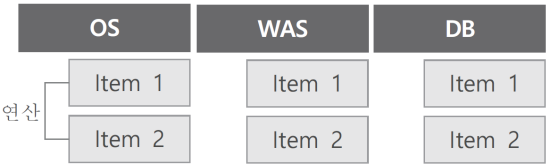
항목	이름	분석 구분	구간 정의										분석 요건 최소 수치	비고
			L1	L2	L3	L4	L5	L6	L7	L8	L9	L10		
Server	CPU 사용률	최근 분포	30	40	50	60	70	75	80	85	90	95	30	최근 5분 70% 값
		중장 기분포	30	40	50	90	70	75	80	85	90	95	30	최근 10분 70% 값
		최근 상승률	0.1	0.3	0.5	1	1.5	2	2.5	3	3.5	4	50	최근 5분 경사도
		전주 비교	1.5	2	3	4	5	6	7	8	9	10	50	최근 10분 평균 vs 前주 10분 평균 상승률
	CPU Wait IO 사용률	최근 분포	10	20	30	40	50	60	70	80	90	99	10	최근 5분 70% 값
		중장 기분포	10	20	30	40	50	60	70	80	90	99	10	최근 10분 70% 값
		최근 상승률	0.1	0.3	0.5	1	1.5	2	2.5	3	3.5	4	10	최근 5분 경사도
		전주 비교	1.5	2	3	4	5	6	7	8	9	10	10	최근 10분 평균 vs 前주 10분 평균 상승률
	socket 개수	최근 분포	100	200	400	600	1000	2000	5000	10000	20000	30000	100	최근 5분 70% 값
		중장 기분포	100	200	400	600	1000	2000	5000	10000	20000	30000	100	최근 10분 70% 값
		최근 상승률	0.1	0.3	0.5	1	1.5	2	2.5	3	3.5	4	400	최근 5분 경사도
		전주 비교	1.5	2	3	4	5	6	7	8	9	10	400	최근 10분 평균 vs 前주 10분 평균 상승률

항목	이름	분석 구분	구간 정의										분석 요건 최소 수치	비고
			L1	L2	L3	L4	L5	L6	L7	L8	L9	L10		
WAS	WAS Active Service 개수	최근 분포	5	10	15	20	30	40	50	60	80	100	5	최근 5분 70% 값
		중장 기분포	5	10	15	20	30	40	50	60	80	100	5	최근 10분 70% 값
		최근 상승률	0.1	0.3	0.5	1	1.5	2	2.5	3	3.5	4	5	최근 5분 경사도
		전주 비교	1.5	2	3	4	5	6	7	8	9	10	5	최근 10분 평균 vs 前주 10분 평균 상승률
	WAS 응답시간	최근분포	1000	2000	3000	6000	9000	15000	30000	60000	120000	600000	1000	최근 5분 70% 값
		중장 기분포	1000	2000	3000	6000	9000	15000	30000	60000	120000	600000	1000	최근 10분 70% 값
		최근 상승률	0.1	0.3	0.5	1	1.5	2	2.5	3	3.5	4	1000	최근 5분 경사도
		전주 비교	1.5	2	3	4	5	6	7	8	9	10	1000	최근 10분 평균 vs 前주 10분 평균 상승률
	WAS GC Time	최근 분포	500	1000	2000	3000	5000	8000	10000	15000	30000	60000	500	최근 5분 70% 값
		중장 기분포	500	1000	2000	3000	5000	8000	10000	15000	30000	60000	500	최근 10분 70% 값
		최근 상승률	0.1	0.3	0.5	1	1.5	2	2.5	3	3.5	4	500	최근 5분 경사도
		전주 비교	1.5	2	3	4	5	6	7	8	9	10	500	최근 10분 평균 vs 前주 10분 평균 상승률
	con_user	최근 분포	100	200	300	500	800	1200	1600	2500	3500	5000	100	최근 5분 70% 값
		중장 기분포	100	200	300	500	800	1200	1600	2500	3500	5000	100	최근 10분 70% 값
		최근 상승률	0.1	0.3	0.5	1	1.5	2	2.5	3	3.5	4	100	최근 5분 경사도
		전주 비교	1.5	2	3	4	5	6	7	8	9	10	100	최근 10분 평균 vs 前주 10분 평균 상승률
DB	DB Lock수	최근 분포	1	3	5	7	10	15	20	30	40	50	1	최근 5분 70% 값
		중장 기분포	1	3	5	7	10	15	20	30	40	5	1	최근 10분 70% 값
		최근 상승률	0.1	0.3	0.5	1	1.5	2	2.5	3	3.5	4	1	최근 5분 경사도
		전주 비교	1.5	2	3	4	5	6	7	8	9	10	1	최근 10분 평균 vs 前주 10분 평균 상승률
	DB Active Session수	최근 분포	50	100	150	200	300	400	500	600	700	800	50	최근 5분 70% 값
		중장 기분포	50	100	150	200	300	400	500	600	700	800	50	최근 10분 70% 값
		최근 상승률	0.1	0.3	0.5	1	1.5	2	2.5	3	3.5	4	50	최근 5분 경사도
		전주 비교	1.5	2	3	4	5	6	7	8	9	10	50	최근 10분 평균 vs 前주 10분 평균 상승률

항목	이름	분석 구분	구간 정의										분석 요건 최소 수치	비고
			L1	L2	L3	L4	L5	L6	L7	L8	L9	L10		
	DB Sort Rows수	최근 분포	100000	200000	300000	1000000	4000000	8000000	10000000	20000000	30000000	50000000	100000	최근 5분 70% 값
		중장 기분포	100000	200000	300000	1000000	4000000	8000000	10000000	20000000	30000000	50000000	100000	최근 10분 70% 값
		최근 상승률	0.1	0.3	0.5	1	1.5	2	2.5	3	3.5	4	100000	최근 5분 경사도
		전주 비교	1.5	2	3	4	5	6	7	8	9	10	100000	최근 10분 평균 vs 前주 10분 평균 상승률
	total session	최근 분포	300	600	900	1200	1600	2000	4000	6000	8000	10000	300	최근 5분 70% 값
		중장 기분포	300	600	900	1200	1600	2000	4000	6000	8000	10000	300	최근 10분 70% 값
		최근 상승률	0.1	0.3	0.5	1	1.5	2	2.5	3	3.5	4	300	최근 5분 경사도
		전주 비교	1.5	2	3	4	5	6	7	8	9	10	300	최근 10분 평균 vs 前주 10분 평균 상승률



〈그림 3〉 성능수치 등급화(예시)



〈그림 4〉 수치 그룹화

4가지 관점의 개별수치를 항목별로 성능수치별 운영에 맞는 등급을 부여하게 된다 (그림 3). 이때 부여되는 등급의 구간과 범위 등은 운영자의 운영 경험치를 기반으로 하며, 등급부여에 따라 서로 다른 성격의 수치 연산이 가능하게 된다.

각 성능수치 등급화가 완료되면, 인프라 요소별 즉, Server, WAS, DB, SAP 등의 요소별로 수치들을 그룹화 하여 연산하게 된다 (그림 4).

그룹별로 연산된 수치들을 분단위(또는 일정 단위별로)로 합산하여 하나의 수치로 단일화하여 추이를 분석하게 된다 (그림 5).

<표 4>는 다양한 성능 수치를 하나로 통합하기 위해 동일한 기준으로 등급화 하여 간소화하는 과정을 나타내고 있다.

위에서 설명한 데이터 전처리 과정을 정리하면 다음과 같다.

〈그림 5〉 수치 단일화 개념도

첫째, 운영 경험치 기반의 4가지 관점을 반영한 데이터 확장을 예를 들어 살펴보면 다음과 같다.

최근 임계수치 분포(RD) 또한 최근 5분간의

CPU(%) 상승률(기울기)을 계산하여 사전에 기정의한 10개 구간 중 계산한 값이 위치한 구간의 Level 값을 구한다.

임계수치 중장기 분포(M) 분석은 최근 10분간의 CPU(%) 데이터 분포에서 상위 70%를 샘플링한다. 사전에 정의한 10개 구간 중 샘플링 한 값이 위치한 구간의 Level값을 기록한다.

최근 상승률(G)의 경우, 최근 5분간의 CPU(%) 데이터 분포에서 상위 70% 값을 샘플링한 뒤 사전에 정의한 10개 구간 중 샘플링 한 값이 위치한 구간의 Level 값을 기록한다.

지난 주 대비 변화율(WC)은 (최근 10분간의 CPU(%) 평균)을 지난 주 동일시간 10분 평균으로 나누어 계산한다. 사전에 정의한 10개 구간 중 계산 값이 위치한 구간의 Level값을 기록한다.

둘째, 이렇게 기록한 각 관점의 점수들을 활용하여 업무서비스 내 모든 장비의 점수를 합산하게 된다. 단, 그룹별(서버, WAS, DBMS 그룹별)로 합산을 한다. 이때, 서버점수의 합산은 CPU점수와 Memory 점수, WIO점수, Socket 점수를 합산한 것이 된다.

셋째, 학습기간 기준으로 그룹별 수치를 재계산하여 특정시점의 단일화된 성능 지표 수치를 도출하게 된다. 예를 들어 특정시점의 WAS 최종점수는 WAS평균에 표준편차를 더한 뒤 이 값으로 WAS 점수를 나누어 산출한다.

넷째, 최종 단일화된 수치는 이렇게 산출된 특정시점의 각 인프라 별 점수를 모두 합산하여 산출한다.

2.4 데이터 모델링

전처리 과정을 거쳐 수치 단일화된 각 인프라의 성능지표를 기반으로 업무서비스별 시간 흐름에 따른 이상패턴에 대한 학습을 진행하게 된다.

본 연구에서는 두 가지 학습 모델을 활용하였으며, 두 모델의 분석 결과 모두 이상패턴으로 판단할 경우 최종적으로 이상 패턴으로 판단하게 된다.

첫 번째 학습모델로는 Facebook에서 제공하는 예측모델인 Prophet 모델을 활용하였다.

Prophet 모델은 비주기적 변화를 반영하는 트렌드(growth, g(t)), 주기적인 변화를 나타내는 계절성(seasonality, s(t)), 불규칙 이벤트를 반영하는 휴일(holiday, h(t))의 3가지 주요 항목으로 이루어져 있다. [14]

데이터센터 내 인프라 장비의 성능지표를 기반으로 하는 장애분석의 경우, 이러한 세가지 항목들을 모두 반영할 수 있으며, Prophet 모델은 curve-fitting으로 예측문제를 해결할 수 있어 활용하기에 용이하다. 성능지표가 학습한 추이의 동적 임계치를 넘어가면 이상치로 판정하게 된다.

두 번째 학습모델로는 스트리밍 데이터의 이상감지를 위한 분석 모형인 RRCF(Robust Random Cut Forest)로 AWS에서 Sagemaker의 모듈로 제공하고 있다[5]. RRCF는 데이터셋에서 이상치를 감지하는 비지도학습모델로 데이터센터의 장애발생과 같이 발생 빈도가 아주 희박한 상황에 대한 예측 분석의 학습모델에 활용하기 적합하다[4].

전처리된 데이터를 랜덤하게 샘플링한 뒤 다수의 Tree를 생성하게 되며, 루트노드와 가까울수록 이상 패턴에 가깝다고 판단하게 된다. 본 연구에서는 이러한 RRCF 학습모델을 활용하여 인프라의 성능지표를 단일 수치화 한 뒤, 랜덤하게 feature를 추출하여 여러 개의 tree를 구성한다. 현재 데이터와 과거 일정 기간의 데이터를 하나의 입력데이터로 사용하며, 새로운 데이터가 학습한 이상치의 3σ를 적용하여 이상패턴을 감지하게 된다.

운영 시스템에 학습 모델을 적용한 후 실제 장애 또는 성능 저하 현상이 있었던 여러 사례들을 살펴보면, 두 개의 모델에 대한 이탈률이 모두 20% 이상인 경우가 다수로 분석되었다.

그러나 고려해야 하는 특이사항 중, 운영서비스를 일시적으로 중단했을 경우, Prophet 모델의 경우 이탈율이 0으로 나오는 반면, RRCF 모델의

경우 상당히 큰 이탈율을 보이게 된다. 이러한 현상을 반영하기 위해서 두 모델의 평균이 30% 이상인 경우도 이상현상의 고려대상이라 판단하게 된다.

이에 기본 이상치 탐지 조건을 다음과 같이 정의하였다.

1) Prophet 모델 : 예측 범위 상단을 20% 이상 초과하면 이상현상으로 판단
2) RRCF 모델 : Cutoff 점수를 20% 이상 초과하면 이상현상으로 판단
3) 두 모델의 초과비율 평균이 30% 이상 초과하면 이상현상으로 판단

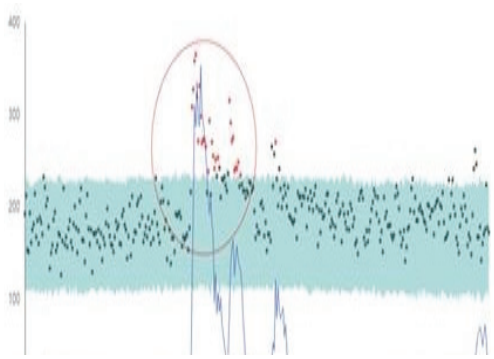
결론적으로 최종 이상현상 판단 기준은, (Prophet>20%) AND (RRCF>20%) AND [(Prophet and RRCF) >30%] = 이상현상

다시 설명하면, RRCF와 Prophet 모델이 모두 이상치라고 판단하고, 두 모델의 평균값 비율이 30% 이상에 해당하는 경우에만 장애와 유사한 이상징후 상황이라고 결론을 내리게 된다.

2.5 시각화 및 탐색

구현된 학습모델에 의해 이상현상이 검출되었을 경우, 시각화를 통해 이상징후 분석 후 장애사전 예방을 위한 탐색 작업을 진행하게 된다.

예를 들어, 다음 (그림 6)과 같이 RRCF와 Prophet 학습 결과 이상징후가 감지되었을 경우, 우선 운영담당자에게 통보를 보내게 된다.



〈그림 6〉 이상징후 감지 사례

다음 (그림 7)과 같이 통합모니터링을 통해 이상징후 감지 시점에서의 성능지표 흐름을 확인할 수 있다.

서버와 WAS 등에서 이상패턴을 보이는 지점을 확인할 수 있으며, 개별 인프라 항목이 아닌 통합모니터링이 가능하다.

〈그림 7〉 이상징후 통합 모니터링 화면

(그림 7)의 첫 번째 차트는 서버, WAS, DB 전체의 성능지표에서 이상징후를 보이는 시점을 분석할 수 있으며, 차례대로 서버와 WAS 각각의 지표 흐름을 보여주고 있다. 그림에서 보는 바와 같이 이상징후가 보였던 시점에 서버와 WAS 성능지표 또한 이상치를 보이고 있다. 좀 더 자세히 성능지표들을 탐색하기 위해 각 세부성능들을 상세히 보여주고 있는 상세화면을 살펴보면 다름 (그림 8)과 같다.

〈그림 8〉 성능지표 상세 분석 화면

(그림 8)은 서버, WAS, DB 각각의 성능을 상세하게 분석할 수 있는 화면으로, 이상징후가 감지된 시점에 각 인프라 영역의 성능지표들을 분석할 수 있다.

성능 상세에 대한 분석에 따라 이상징후를 보이는 원인 파악이 가능해지며, 이는 운영담당자의 운영에 대한 경험치와 시스템에서 설명하고 있는 현상들을 종합적으로 분석하여 결과를 도출하게 되는 것이다.

이러한 결과는 장애가 발생하기 전에 이상징후를 감지할 수 있게 되어 장애 사전 대응이 가능하며, 항목들의 튜닝 시점을 결정하는 데 도움을 줄 수가 있다. 이는 인프라 운영에 매우 중요한 것으로, 다양하고 복잡한 인프라 구성으로 인해 개별 항목들의 튜닝시점 파악이 쉽지 않다, 그로 인해 결정적인 장애의 원인이 되기도 하는데, 그 시점에 대한 가이드를 제공하게 된다는 것은 결정적인 장애를 사전에 방지할 수 있게 된다는 것이다.

이러한 분석 결과의 활용도를 높이고 데이터센터의 장애예방을 위한 장애분석 시스템 서비스를 구현하였다 (그림 9).

〈그림 9〉 장애분석 시스템 화면

Ⅲ. 결 론

데이터센터에서 발생하는 인프라 장애는 단일 시스템 성능지표 분석만으로는 사전 예방 뿐만 아니라 신속한 대응도 어렵다. 기존에는 단일 시스템 성능지표 분석 위주로 장애예측과 장애이상 패턴분석이 이루어 졌으나, 실제 데이터센터 내 인프라 운영에 대한 환경이 매우 다양하고, 복잡하여 실제 운영에 대한 경험치가 반영된 데이터 처리가 핵심이 된다.

이에 본 연구에서는 데이터센터 장애 예방을 위해 현황에 맞는 문제정의를 하고, 인프라 장비들의 성능지표들을 수집, 운영자의 경험치를 반영하여 그룹, 등급화 한 뒤 수치단일화를 진행하였다. 이는 기존의 단일 인프라 항목을 기반으로 하는 장애 예측 방법 보다 데이터센터의 복잡하고 다양한 장애와 이상현상을 설명하는데 용이하며 정확한 결과를 도출 할 수가 있다. 또한 단일 인프라 수치를 적용한 장애 예측의 결과는 현실적으로 적용이 무의미한 것에 비해, 본 연구에서 접근한 단일화된 운영데이터 기반의 장애 이상치 검출은 실제 운영자들에게 도움이 되고 있는 것을 실험적으로 알 수가 있다.

신뢰성 있는 결과 도출을 위해 시계열 예측과 이상징후 탐지를 위해 RRCF와 Prophet 모델을 Ensemble하여 활용하였으며, 인프라 운영담당자들의 용이한 활용을 위해 장애분석시스템 서비스를 구현하였다.

참 고 문 헌

[1] 이택현, 국광호, “RRCF 알고리즘을 활용한 RAN 장비 이상 검출에 관한 연구”, 한국정보통신학회 춘계 종합학술대회 논문집, pp.581-583, 2021.

[2] 이현용, 김낙우, 이준기, 이병탁, “효과적인 이상 진단을 위한 클러스터링의 타당성 연구”, 한국정보처리학회 학술대회논문집, pp.428-430, 2020.

[3] 오민지, 최은선, 노경우, 김재성, 조완섭, “제조설비 이상탐지를 위한 지도학습 및 비지도학습 모델 설계에 관한 연구”, 한국빅데이터학회지

제6권 제1호, pp.23-35, 2021.

[4] 조준모, “빅데이터의 정규화 전처리과정이 기계학습의 성능에 미치는 영향”, Journal of the KIECS, Vol.14, No.3, pp.547-552, 2019.

[5] S. Guha, N.Mishra, G.Roy, O.Schrijvers, “Robust Random Cur Forest Based Anomaly Detection On Streams”, Proceedings of the 33rd International Conference on Machine Learning, New York, JMLR: W&CP Vol.48, 2016.

[6] M.D.Bartos, A.Mullapudi, S.C.Troutman, “Implementation of the Robust Random Cut Forest algorithm for anomaly detection on streams”, The Journal of Open Source Software, 4(35), 2019.

[7] D.Hendrycks, M.Mazeika, T.Dietterich, “Deep Anomaly Detection with Outlier Exposure”, conference paper at ICLR, 2019.

[8] 류승택, “Interactive Data Visualization Based Realtime Monitoring and Fault Detection System”, Journal of Knowledge Information Technology and Systems, Vol.13, No.4, pp.421-428, 2018.

[9] 한무명초, 이충권, Kim Yang Sok, “제조공정에서 센서와 머신러닝을 활용한 불량예측 방안에 대한 연구”, Entrue Journal of Information Technology, Vol.17, No.1, pp.89-98, 2019.

[10] 나성일, 김형중, “빅데이터 기반의 IoT 이상 장애 탐지 시스템 설계”, Journal of Digital Contents Society, Vol.19, No.2, pp.377-383, 2018.

[11] 임복출, 김순곤, “서버 성능 관리를 위한 장애 예측 시스템”, 한국정보전자통신기술학회논문지, Vol.11, No.6, pp.684-690, 2018.

[12] 최우형, 황현숙, 김창수, “데이터센터의 설비 통합 모니터링 시스템 설계에 관한 연구”, Journal of the Korea Institute of Information and Communication Engineering, Vol.19, No.4, pp . 909-916, 2015.

[13] 임선열, 최효근, 이규열, 이태훈, 유헌창, “기계학습을 활용한 IoT 플랫폼의 이상감지 시스템”, 한국정보처리학회 추계학술발표대회 논문집, 제28권, 제2호, pp.1001-1004, 2018.

[14] G.Pang, C.Shen, L.Cao, A.Hengel, “Deep Learning for Anomaly Detection:A Review”, ACM Comput. Surv., Vol.1, No.1, 2020.

[15] S.Taylor, B.Letham, “Forecasting at Scale”, The American Statistician, Vol.72, pp.37-45, 2018.

[16] Jong Min Kim, Jaiwook Baik, “Anomaly Detection in Sensor Data”, 신뢰성응용연구, 제18권, 제1호, pp.20-32, 2018.

[17] 천강민, 양재경, “앙상블 모델 기반의 기계 고장 예측 방법”, J.Soc.Korea Ind. Syst. Eng., Vol.43, No.1, pp.123-131, 2020.

[18] 김얼, 김다연, 채윤주, 신동렬, “빅데이터 분석 기반 서버관리 플랫폼 설계”, 한국정보과학회 학술발표논문집, pp.233-234, 2015.

[19] 고경철, 이양원, “평균과 표준편차를 이용한 자동 임계치-결정 알고리즘,”한국컴퓨터교육학회 논문지 “, 제8권, 제6호, pp.103-111, 2005.

[20] 신현승, 유승주, “서버시스템에서의 메모리 불량 현상 분석 및 해결방법”, 전기전자학회논문지, 21(4), pp.353-357, 2017.

[21] 장한나, 윤이삭, 전예은, 김장원, “A study on Patent Invention Trend Analysis using Prophet”, 한국정보처리학회 추계학술대회, pp.752-753, 2019.

[22] S.Kwon, “Anomaly Detection of Big Time Series Data using Machine Learning”, Journal of Soc. Korea. Ind. Syst. Eng.,Vol.43, No2, pp.33-38, 2020.

[23] I.Golan, R.E.Yaniv, “Deep Anomaly Detection using Geometric Transformations”, 32nd Conference on Neural Information Processing Systems (NeurIPS), 2018.

저 자 소 개

신 현 종 (Hyun-Jong Kim)
·2006년 8월 : 동국대학교 컴퓨터공학과(공학사)
·2013년 8월~현재 : 한화시스템 재직 중
·관심분야 : 데이터센터 통합 모니터링, 빅데이터 분석, 인공지능

김 성 근 (Sung-Keun Kim)
·2007년 2월 : 인천대학교 컴퓨터공학과(공학사)
·2013년 7월~현재 : 한화시스템 재직 중
·관심분야 : 빅데이터 수집/분석, 인공지능, 미들웨어

천 병 환 (Byoung-Whan Chun)
·1998년 2월 : 숭실대학교 컴퓨터학부(공학사)
·1998년 2월~현재 : 한화시스템 재직 중
·관심분야 : IT아키텍처, 클라우드, 데이터분석, 인공지능

진 경 복 (Kyong-Bog, Jin)
·1999년 2월 : 강릉원주대학교 정보통계학과(이학사)
·2000년 3월~현재 : 한화시스템 재직 중
·관심분야 : IDC구축・운영, 업무만족도/생산성 향상, DT전략/KPI 수립

양 승 정 (Seung-Jeong Yang)
·1997년 8월 : 동국대학교 산업공학과(공학박사)
·2011년 1월~현재 : 한화시스템 재직 중
·관심분야 : 빅데이터 분석, 인공지능, 퍼지시스템