네트워크 성능 분석을 위한 통계적 가설 검정의 구현과 평가

Implementation and Evaluation of Statistical Hypothesis Testing for Network Performance Analysis

박성훈

January 15, 2023

Abstract

네트워크 서비스에서 사고 분석(incident diagnosis)는 서비스 지연 혹은 실패 시간을 최소화하고 높은 수준의 서비스를 제공하기 위한 핵심적인 주제로 나타나고 있다. 앞선 보고서에서는 연속적 시계열 데이터(continuous time series)와 시간적 사건(temporal event)들 사이의 상관성을 분석하고자 한 CETS(Correlating Events with Time Series for Incident Diagnosis)[1]와 근본 원인 분석(Root Cause Analysis, RCA)에 대한 연구인 FluxRank [2] 및 ϵ-Diagnosis [3]에 대하여 비교하고 분석한 바 있다. 본 연구에서는 그 중 준수한 정확도를 보이고, 상관성의 방향과 단조 효과를 파악할 수 있는 CETS를 구현하고 그 성능을 확인해보고자 한다.

1 서론

네트워크 서비스에서는 예측치 못한 간섭이나 서비스 중단과 같은 사고들을 피할 수 없는 경우가 있으며 이는 상당한 경제적 손실과 같은 문제들을 발생시킨다. 따라서, 이러한 사고들의 진단 효율을 개선하기 위한 노력이 이루어져 왔다. 이러한 서비스 사고 분석은 주로 서비스 실행 시간에 수집된 텔레메트리 데이터(telemetry data)에 의존하고 있다. 서비스 단위 로그, 성능 카운터(performance counter), 장치/프로세스/서비스 단위 사건 등의 텔레메트리 데이터는 사고 진단을 위한 충분한 정보를 제공하기도 하며 크게 두 가지 항목으로 구분할 수 있다. 바로 연속적 시계열 데이터와 시간적 사건 데이터이다.

데이터 기반 사고 진단에서 텔레메트리 데이터와 시스템 상태 사이의 상관 분석은 주요한 역할을 하는데 이는 상관관계가 인과 분석의 단서를 제공하기 때문이다. 실제 상황에서 이러한 상관된 척도(metric)는 그 자체로 근본 원인이 되는 것은 아니지만 사고 조사를 위한 초기 척도 집합 생성에 유용하게 사용 될 수 있다. 이렇게 시계열 데이터 사이의 상관성이나 시스템 사건들 사이의 상관성을 파악하기 위한 연구는 많이 이루어져 왔으나, 시계열과 사건 시퀀스(event sequence) 사이의 상관성을 평가하는 연구는 이루어지지 않았다. 또한, 서로 다른 두 형태의 데이터의 이형적 특성(heterogeneous property) 때문에 전통적인 상관 분석 방식인 Pearson correlation과 Spearman correlation은 만족스럽지 못한 결과를 보이는 경우가 많다.

CETS 연구에서는 시계열과 사건 시퀀스(event sequence) 사이의 상관성 분석을 다변량 2샘플 가설 검정 문제로 생각하고 최근접 이웃 알고리즘에 기반한 검정 통계량으로 가설을 검정하여 상관관계를 파악하고자 시도하였다. 본 연구에서는 CETS에 기반하여 상관관계 파악 알고리즘을 구현하고, 이상 현상 감지(anomaly detection) 연구 [4]에서 사용된 SMD(Server Machine Dataset) 데이터셋 [5]으로 그 성능을 확인하고자 한다. 또한, SMD 데이터셋에 CETS 알고리즘을 적용함에 있어 발생하는 문제들의 원인을 파악하고 CETS 알고리즘을 일부 개선해보고자 한다.

2 문제 정의와 수식화

2.1 문제 정의

통계적으로 상관성이란 확률적으로 독립이 아닌 두 확률 변수 사이의 통계적 관계를 의미한다. Figure 1는 두 형태의 사건과 시계열 사이의 상관관계를 도시하고 있다. 여기서 CPU 성능 카운터가 CPU intensive program에 대해서 상당히 증가하므로 상관성을 보인다고 할 수 있다. 반대로 disk intensive

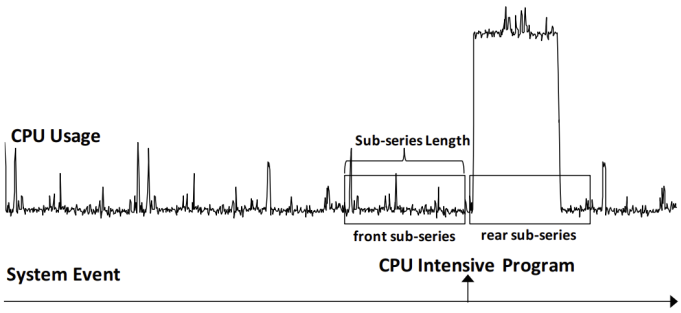

Figure 1: Example of front sub-series and rear-series

program은 상관성을 보이지 않는다. 일반적으로 서비스의 사건 분석에서는 다음의 상관성의 세 측면을 고려한다.

상관성의 존재 - 사건 시퀀스와 시계열 사이에 상관성이 존재하는가? 통계적 상관성이 인과성을 입증하지는 못하나, 인과관계 파악에 유용하게 사용될 수 있다.

의존성의 시간적 순서 - 의존적 데이터 쌍에 대하여 상관관계의 방향은 인과 분석을 위하여 유용하게 사용될 수 있다.

의존성의 단조 효과 - 많은 경우에서 사건의 발생은 시계열 값의 상당한 증가와 연관되어 있다. 이러한 두 데이터 흐름 사이의 양, 혹은 음의 영향을 파악하는 것은 인과적 추론에서 중요한 역할을 한다. 예를 들어, 우리의 도메인 지식 상에서 CPU usage의 증가는 quality SLA를 위반할 수 있으나 감소는 그렇지 않다.

사건 시퀀스 E에 대하여 사건의 타임 스탬프(timestamp)를 $T_E = (t_1, t_2, \cdots, t_n)$라고 명시한다. 시계열 $S = (s_1, s_2, \cdots, s_m)$에 대하여 타임 스탬프는 $T_S = (t(s_1), t(s_2), \cdots, t(s_m))$으로 명시하며 $t(s_i) = t(s_{i-1}) + \tau$의 관계를 갖는다. 이때, τ는 샘플링 간격이다. 여기서 샘플링 간격은 거의 일정하다고 가정하며 추가적으로 특정 사건이 시계열에 미치는 영향은 특정한 짧은 시간 간격 동안만 지속된다고 가정한다. 이러한 가정은 실제 상황에서 일반적으로 유효하다.

만약 사건 E와 시계열 S가 상관관계를 갖는다면 사건 E가 일어날 때마다 시계열 S에 상응하는 변화가 발생한다. 이때 각 변화를 S의 부분열(sub-series)이라 한다. $\ell_k^{rear}(S, e_i)$를 $e_i \in E$가 k 길이로 일어난 후의 부분열이라고 하고, $\ell_k^{front}(S, e_i)$를 e_i가 일어나기 전의 부분열이라고 한다. 직관적으로 보았을 때, 만약 E가 S과 상관성이 없다면 위 두 부분열과 e_i 사이의 관계성은 존재하지 않는다. 즉, $\Gamma^{front} = \{\ell_k^{front}(S, e_i), i = 1, \cdots, n\}$와 $\Gamma^{rear} = \{\ell_k^{rear}(S, e_i), i = 1, \cdots, n\}$는 랜덤하게 샘플링한 부분열 Θ와 통계적으로 다르지 않다. 위 정보를 기반으로 아래의 정의를 나타낸다.

Definition 1. 사건 시퀀스 E와 시계열 S가 상관관계를 가지고 E가 S의 변화 이후에 주로 나타난다.($S \to E$로 표기한다) $\leftrightarrow \{\ell_k^{front}(S, e_i), i = 1, \cdots, n\}$의 확률 분포는 랜덤하게 샘플링한 부분열 Θ와 통계적으로 다르다.

Definition 2. 사건 시퀀스 E와 시계열 S가 상관관계를 가지고 E가 S의 변화 이전에 주로 나타난다.($E \to S$로 표기한다) $\leftrightarrow \{\ell_k^{rear}(S, e_i), i = 1, \cdots, n\}$의 확률 분포는 랜덤하게 샘플링한 부분열 Θ와 통계적으로 다르며, $\{\ell_k^{front}(S, e_i), i = 1, \cdots, n\}$의 확률 분포는 랜덤하게 샘플링한 부분열 Θ와 통계적으로 다르지 않다.

Definition 3. E와 S의 관계가 $S \to E$ 또는 $E \to S$이면 E와 S는 상관되어 있다. ($E \sim S$)

Definition 4.
만약 $E \to S$이고 E의 발생이 S의 상당한 증가와 연관되어 있다면 $E \xrightarrow{+} S$라고 표기한다.
만약 $E \to S$이고 E의 발생이 S의 상당한 감소와 연관되어 있다면 $E \xrightarrow{-} S$라고 표기한다.

2.2 2샘플 가설 검정 문제로의 모델링

위 정의에 기반하여 상관 분석을 다변량 2샘플 가설 검정 문제(multivariate two-sample hypothesis-testing problem)로 생각할 수 있다. 이 논문의 배경에서 하나의 샘플은 부분열 Γ^{front} (또는 Γ^{rear})이고, 다른 샘플은 S에서 랜덤하게 추출된 부분열 집합 Θ이다. 이때, 각 데이터 포인트는 k차원 벡터로 나타난다. 따라서, 원래의 문제는 다변량 2샘플 가설 검정으로 환원된다. Γ^{front}와 Θ가 각각 알려지지 않은 분포 F와 G에서 랜덤하게 생성된 독립적인 샘플이라고 하면 아래와 같은 2샘플 가설을 설정할 수 있다. 이때, 분포들은 르베그 측도(Lebesgue measure)에 대하여 절대 연속(absolutely continuous)*인 것으로 가정한다.

$$\begin{cases} H_0 & : F = G \\ H_1 & : F \neq G \end{cases} \tag{1}$$

만약 귀무가설 H_0를 기각한다면 (H_1을 채택) Γ^{front}와 Θ은 통계적으로 다르며, $Definition1$에 따라 S와 E가 상관관계를 갖는다. 반대로 H_0를 채택한다면 S와 E가 상관관계를 갖지 않는다.

*분포가 르베그 측도에 대하여 절대 연속이면 분포는 derivative의 르베그 측도에 대한 적분으로 나타낼 수 있다. (즉, 분포는 확률 밀도함수를 갖는다.)

3 접근 방식

3.1 최근접 이웃 알고리즘

$\Gamma^{front} = \{\ell_k^{front}(S, e_i), i = 1, \cdots, n\}$와 $\Gamma^{rear} = \{\ell_k^{rear}(S, e_i), i = 1, \cdots, n\}$와 $\Theta = \{\theta_0, \theta_0, \cdots, \theta_{n'}\}$에 대하여 $Z = \Gamma^{front} \cup \Theta$로 정의한다. 그리고 다음과 같이 라벨링한다.

$$Z_i = \begin{cases} \ell_k^{front}(S, e_i) & \text{if } i = 1, \cdots n \\ \theta_{i-n} & \text{if } i = n+1, \cdots p(= n + n') \end{cases} \tag{2}$$

유한 부분열 집합 A와 부분열 $x \in A$에 대하여 $NN_r(x, A)$를 집합 $\{A \backslash x\}$에서 x와 r번째로 가까운 이웃(r-th nearest neighbor)이라고 한다. 상호 배재 부분집합(mutually exclusive subset) A_1과 A_2과 $x \in A$에 대하여 다음의 지시 함수를 정의한다.

$$I_r(x, A_1, A_2) = \begin{cases} 1 & \text{if } x \in A_i \text{ and } NN_r(x, A) \in A_i \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

본 문제에서 최근접 기반 검정은 다음의 정량적 수치에 의존한다.

$$T_{r,p} = \frac{1}{pr} \sum_{i=1}^{p} \sum_{j=1}^{r} I_j(x, A_1, A_2) \tag{4}$$

여기서 $p = |n + n'|$는 샘플의 크기이다. 검정 통계량 $T_{r,p}$는 한 부분열과 전체 샘플 영역 Z에서의 최근접 원소 중 하나로 구성되는 모든 쌍 중에서 같은 샘플로부터의 두 부분열로 구성되는 쌍의 비율이다. 직관적으로 $T_{r,p}$는 두 샘플이 잘 섞인 귀무가설 하에서는 작고 두 분포가 다를수록 크다. p가 충분히 크다면 $(pr)^{1/2}(T_{r,p} - \mu_r)/\sigma_r$는 표준 정규분포를 따른다. 따라서, $(pr)^{1/2}(T_{r,p} - \mu_r)/\sigma_r > \alpha$, (where $\alpha = 1.96$ for $P = 0.0025$)로 귀무가설을 기각할 수 있다.

3.2 상관관계의 존재와 순서 파악

Section2에 기반하면 E와 S가 상관되어 있을 경우 Γ^{front}와 Θ 혹은 Γ^{rear}와 Θ가 통계적으로 다르다. 만약 전부분열(front sub-series) Γ^{front}가 랜덤하게 샘플링된 부분열 Θ와 통계적으로 다르다면 $S \rightarrow E$이다. 반대로 후부분열(rear sub-series) Γ^{front}가 Θ와 통계적으로 다르다면 $E \rightarrow S$이다. Figure 2에서 CPU intensive program $\rightarrow$ CPU usage and CPU usage $\rightarrow$ query alert를 나타내고 있다.

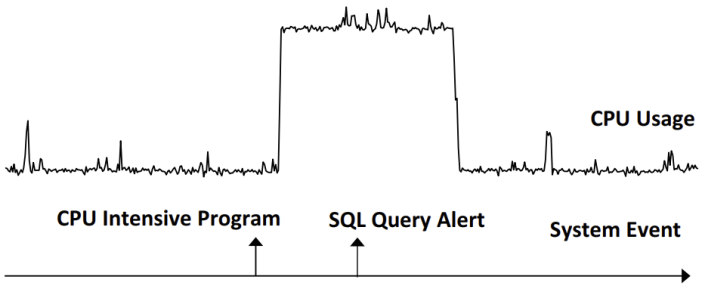

Figure 2: Example of temporal order, CPU intensive program → CPU usage and CPU usage → query alert.

3.3 효과 형태의 판별

Definition 4에 따르면 $E \xrightarrow{+} S$과 같은 단조 효과를 결정하기 위해서는 사건 E가 일어난 후에 시계열 S의 상당한 증가가 나타나는지를 확인해야 한다. 이 문제 역시 유사하게 통계적 가설 검정 문제로 수식화할 수 있다. 이 논문에서는 Γ^{front}와 Γ^{rear} 사이에 상당한 증가가 발생하였는지 확인하기 위하여 t-검정을 사용한다. t_{score}는 다음과 같이 나타난다.

$$t_{score} = \frac{\mu_{\Gamma^{front}} - \mu_{\Gamma^{rear}}}{\sqrt{\frac{(n_1-1)\sigma^2_{\Gamma^{front}} + (n_2-1)\sigma^2_{\Gamma^{rear}}}{n_1+n_2-2}\left(\frac{1}{n_1} - \frac{1}{n_2}\right)}} \tag{5}$$

본 논문에서는 $n_1 = n_2 = n$이므로 Equation 5는 다음과 같다.

$$t_{score} = \frac{\mu_{\Gamma^{front}} - \mu_{\Gamma^{rear}}}{\sqrt{\frac{\sigma^2_{\Gamma^{front}} + \sigma^2_{\Gamma^{rear}}}{n}}} \tag{6}$$

만약 $t_{score} > \alpha$면 음의 단조 영향이 있는 것이고 $t_{score} < \alpha$이면 양의 단조 영향이 있는 것으로 판정한다. $|t_{score}| < \alpha$인 경우에는 단조 영향을 높은 신뢰도(confidence)로 결정할 수는 없으나 실제 사고 진단 시나리오에서 이러한 경우는 적다.

3.4 전체 알고리즘

Figure 3에서는 CETS의 전체 알고리즘을 나타내고 있다.

3.5 파라메터 설정

부분열의 길이 k는 알고리즘의 성능에 크게 영향을 미친다. Figure 4에서는 두 그래프를 도시하고 있는데 각 경우에서의 부분열의 길이 k에 대한 $(Nk)^{1/2}(T_{k,N} - \mu_k)/\sigma_k$를 나타내고 있다. α가 주어질 때, 이 값은 대립가설 H_1의 신뢰도와 직접적으로 연관된다. 높은 $(Nk)^{1/2}(T_{k,N} - \mu_k)/\sigma_k$ 값은 대립가설 H_1의 높은 신뢰도를 의미한다. 이를 "신뢰 계수(confidence)"라고 표기한다. Figure 4의 검은 선은 "CPU Extensive Program"과 "CPU Usage" 사이의 상관관계를 평가하여 얻은 신뢰 계수이다. 도메인 지식으로부터 "CPU Extensive Program"과 "CPU Usage" 사이에 상관관계가 존재한다는 것을 알고 있으므로 신뢰 계수가 높을수록 좋다는 것을 알 수 있다. k가 너무 작으면 제한된 정보량으로 인하여 낮은 신뢰 계수를 보이며 반대로 k가 너무 크면 사건의 영향이 감소하고 Θ에 근사하게 되므로 역시 신뢰 계수가 감소한다.

Figure 4의 파란 선은 "Disk Extensive Program"과 "CPU Usage" 사이의 상관관계를 평가하여 얻은 신뢰 계수이다. 신뢰 계수는 거의 0에 가까우며 k의 영향을 받지 않는다. 이는 상관성이 존재하지 않는 경우 대립 가설 H_1의 기각이 k의 영향을 받지 않음을 의미한다. 특정한 경우에는 k를 도메인 지식에 기반하여 선택할 수도 있으나 수많은 시계열 데이터 및 사건들이 존재하는 실제 환경에서 부분열 길이를 미리 선택하는 것은 어렵다. 따라서, 자기상관 함수(autocorrelation function)에 기반하여 k를 선택하는 방식을 사용한다. 시계열 $S = (s_1, s_2, \cdots, s_n)$에 대하여 자기상관은 다음과 같이 나타난다.

$$R(l) = E(s_i * s_{i-1}) \tag{7}$$

Algorithm 1: The Overall Algorithm

Input: Event $E = (e_1, e_2, ..., e_n)$, and Time Series $S = (s_1, s_2, ..., s_m)$, and the sub-series length k.

Output: The correlation flag C, the direction D, and the effect type T

Initialize Γ^{front} and Γ^{rear};
Initialize Θ;
Initialize $R = false$, $D = NULL$, $T = NULL$;
Normalize each $\ell_k^{front}(S, e_i)$ and $\ell_k^{rear}(S, e_i)$.;
Test Γ^{front} and Θ using Nearest Neighbors Method. The result is denoted as D_f.;
Test Γ^{rear} and Θ using Nearest Neighbors Method. The result is denoted as D_r.;
if $(D_r == true \&\& D_f == false)$ **then**
    $R = true$.;
    Calculate t_{score} using Equation (8).;
    if $(t_{score} > \alpha)$ **then**
        $T = E \xrightarrow{-} S$.;
    else if $(t_{score} < -\alpha)$ **then**
        $T = E \xrightarrow{+} S$.;
else if $(D_r == false \&\& D_f == true)$ || $(D_r == true \&\& D_f == true)$ **then**
    $R = true$.;
    Calculate t_{score} using Equation (8).;
    if $(t_{score} > \alpha)$ **then**
        $T = S \xrightarrow{-} E$.;
    else if $(t_{score} < -\alpha)$ **then**
        $T = S \xrightarrow{+} E$.;
Out put R, D and T;
Algorithm End.

Figure 3: CETS overall algorithm

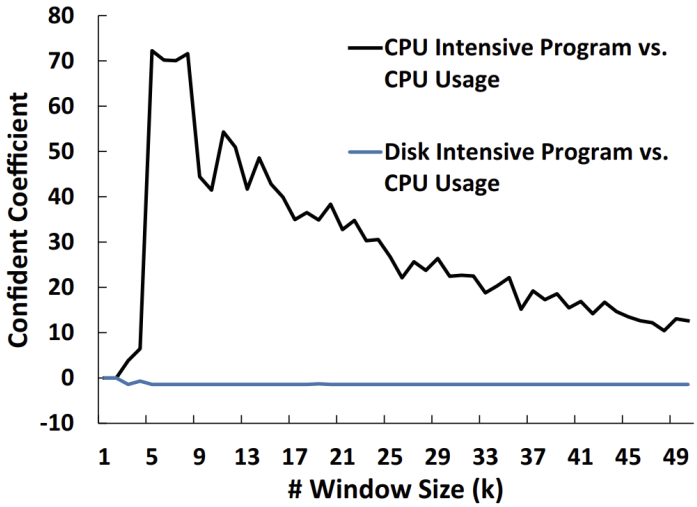

Figure 4: Confidence vs. sub-series length k

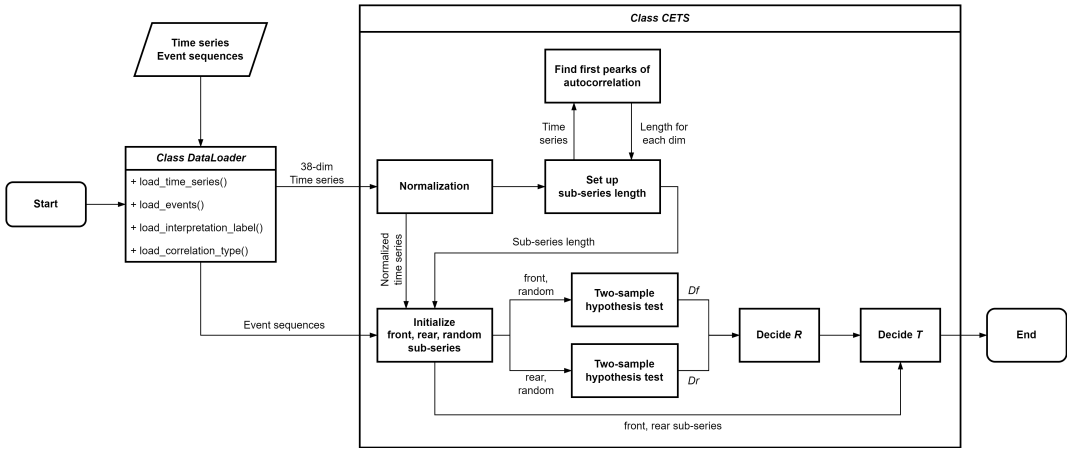

Figure 5: CETS flow chart.

이때, l은 상관관계의 지연(lag)을 의미한다. 자기상관 함수는 시계열의 신호 에너지를 나타내기 위해 사용될 수 있다. 그러므로 k는 시계열의 주요한 신호를 포함하기 위하여 자기상관 함수의 첫 번째 극대점(peak)으로 설정될 수 있다. k에 더하여 이웃의 수 r 역시 주요한 파라메터이다. 이때, $r = ln(p)$이 적절한 선택임이 증명되었다. 다만, p값이 작은 경우에는 $\frac{p}{2}$이하의 적절한 값을 취한다.

3.6 구현 세부사항

Figure 5에서는 CETS의 전체적인 흐름도를 나타내고 있다. CETS는 시계열과 사건 시퀀스를 입력으로 받는다. 이를 위해서 DataLoader 클래스를 구현하였다. DataLoader에서는 입력을 받아 다중 차원 시계열을 넘파이(numpy) 배열과 사건 시퀀스 배열을 만든다. 추가적으로 성능 확인을 위해서는 실제 사실(ground truth)에 해당하는 interpretation label과 correlation type을 불러와 배열을 형성한다. 그리고, 이렇게 형성한 시계열과 사건 시퀀스 배열을 입력으로 하여 CETS 클래스를 생성한다.

3.6.1 CETS 초기화

CETS에서는 초기화 과정에서 시계열을 0과 1사이로 정규화(normalization)하고 시계열의 각 차원에 대하여 부분열 길이를 설정한다. 정규화 과정은 다음과 같이 나타낼 수 있다.

$$T_{norm} = \frac{T - T_{min}}{T_{max} - T_{min}} \tag{8}$$

각 차원의 부분열 길이를 구하는 방식은 3.5에서 기술한 방식을 사용하나, 실제 구현에서는 추가적인 고려 사항이 존재한다. 첫 번째로 자기상관 함수의 첫 극대점을 바로 취하게 되면 노이즈에 영향을 받을 가능성이 있으며 부분열 길이를 과소평가하게 될 수 있다. 따라서, 극대점을 취할 때 적절한 너비를 설정해줄 필요가 있다. 본 연구에서는 선험적으로 ADF(argumented Dickey-Fuller test) [6]의 usedlag의 2배를 너비로 설정하였는데, 이 부분에 대해서는 추가적인 확인이 필요할 것으로 판단된다. 또한, 이렇게 구한 부분열 길이를 그대로 사용하지 않고 최소 및 최대 길이를 설정하였다. 최소 부분열 길이를 설정한 이유는 정상성(stationarity) 경향이 낮은 시계열의 경우 부분열 길이가 과소평가되어 이상 현상의 전파 지연시간을 고려하지 못하고 짧은 구간 내 일시적 변화에 의존하게 되기 때문이다. 더불어 최대 부분열 길이를 설정한 이유는 부분열 길이 k의 증가에 따라 CETS 알고리즘의 실행 시간이 급격하게 증가하기 때문으로 이에 대해서는 4절에서 추가적으로 논의하기로 한다.

3.6.2 인접 샘플 동일시 (Near Sample Equation)

최근접 이웃 알고리즘에서 전부분열(혹은 후부분열) $\Gamma^{front} = \{\ell_k^{front}(S, e_i), i = 1, \cdots, n\}$이 상당히 균일한 경우 (즉, 각 부분열의 유사성이 상당히 큰 경우) 랜덤 부분열 $\Theta = \{\theta_0, \theta_0, \cdots, \theta_n\}$과의 차이가 미미함에도 r개의 인접 부분열을 모두 전부분열로 선택하는 문제가 발생할 수 있다. 이는 특히 전체 샘플의 수 p가 작을 때 발생할 수 있으며 2샘플 가설 검정에서 귀무가설을 기각하게 하여 상관관계 결정의 오류를 유발할 수 있다.

따라서, 본 연구에서는 전부분열과 랜덤 부분열를 합한 샘플 집합 Z 내의 모든 샘플 간 거리(DTW distance [7])를 구했을 때, 최대 거리와 최소 거리의 차가 일정 미만일 경우 전부분열 샘플 집합과 랜덤 부분열 샘플 집합이 충분히 인접했다고 보고 두 샘플 집단을 동일시하는 방식을 사용하였다. 이러한 인접 샘플 동일시(Near Sample Equation, NSE)는 최근접 이웃 알고리즘의 특성으로 나타나는 두 샘플의 통계적 거리의 과대평가를 막아 상관관계 예측 성능을 높일 수 있다. 그러나, NSE의 임계값을 지나치게 높게 설정할 경우 대부분의 검정은 NSE의 임계값에만 의존하게 되어 최근접 이웃 알고리즘을 무의미하게 만든다. 본 연구에서는 임계값을

$$D_{th} = 2k\alpha_{th} \tag{9}$$

로 설정하였으며 α_{th}를 선험적으로 0.0025로 설정하여 NSE를 사용하지 않았을 때보다 높은 성능을 확인하였다. 이 결과에 대해서는 4절에서 기술하고 있다.

4 실험적 평가

4.1 Baseline

비교를 위한 baseline으로는 시계열에서의 상관관계를 찾아내는 알고리즘인 Pearson correlation을 사용하였다. Pearson correlation은 두 시계열 사이의 상관관계를 측정하는 데에 가장 널리 사용되는 방식이다. Pearson correlation cofficient ρ는 다음과 같이 계산된다.

$$\rho_{X,Y} = \frac{cov(X,Y)}{\sigma_X \sigma_Y} = \frac{E[(X-\mu_X)(Y-\mu_Y)]}{\sigma_X \sigma_Y} \tag{10}$$

Pearson correlation은 시계열과 사건 데이터 사이의 상관관계를 파악하기 위하여 직접적으로 사용될 수 없다. 따라서 사건 E를 시계열 $S^E = (s_1^E, s_2^E, \cdots, s_m^E)$을 다음과 같이 계산한다.

$$s_i^E = \begin{cases} 1 & \text{if } t(s_i) \in T_E \\ 0 & \text{otherwise.} \end{cases} \tag{11}$$

그리고 S^E와 S 사이의 Pearson correlation cofficient ρ를 직접적으로 계산한다. 만약 $\rho > p_\alpha$이면 $E \overset{+}{\sim} S$이고, $\rho < -p_\alpha$이면 $E \overset{-}{\sim} S$이다. $|\rho| < p_\alpha$인 경우에는 E와 S가 상관되어 있지 않다. p_α는 pearson 검정을 위한 파라메터로 CETS 연구에서는 0.1을, 본 연구에서는 보다 작은 0.005를 사용하였다.

4.2 Dataset

시계열과 사건 시퀀스 간 상관관계 파악에 대한 공유 데이터셋은 확인하기 어려웠으므로, 본 연구에서는 이상현상 감지 연구에서 사용된 SMD 데이터셋 [5]을 사용하고자 한다. SMD 데이터셋은 인터넷 회사에서 5주간 측정된 데이터셋으로 28개의 서로 다른 장치로 구성된다. 각 장치에서는 38차원의 다변량 시계열

$$S^t = \begin{bmatrix} s_{11}^t & s_{12}^t & \cdots & s_{1k}^t \\ s_{21}^t & s_{22}^t & \cdots & s_{2k}^t \\ \vdots & \vdots & \ddots & \vdots \\ s_{m1}^t & s_{m2}^t & \cdots & s_{mk}^t \end{bmatrix}$$

이 관측되었으며, 본 연구에서는 그 중 8개의 장치의 데이터를 사용하였다. 즉 본 연구에서의 데이터셋은 다변량 시계열 S^t의 집합 $\mathcal{S} = \{S^1, S^2, \cdots, S^8\}$와 이상 현상에 대한 라벨링 $L = (l_1, l_2, \cdots, l_m)$의 집합 $\mathcal{L} = \{L^1, L^2, \cdots, L^8\}$으로 구성된다. 이때, 라벨링은 이상 현상 발생동안 1, 그렇지 않을 때 0으로 표시되며 해당 이상현상에 관여하는 차원 정보도 추가적으로 주어진다. 이때, 라벨링 L이 0에서 1로 바뀌는 지점을 이상현상 발생 시점으로 생각할 수 있으므로 시계열 S^v의 j번째 사건 e_j의 타임 스탬프를 $T_{e_j}^v = (t_1^{vj}, t_2^{vj}, \cdots, t_{n_j}^{vj})$, (where t_i^{vj} = 사건 e_j의 i번째 발생 시간, n_j= 사건 e_j의 전체 발생 횟수)로 간주하여 시계열과 사건 시퀀스 간 상관관계 파악에 대한 데이터셋으로 사용하였다. 이때, 이상현상 감지 연구는 상관관계의 방향과 단조 효과에 대한 정보를 포함하고 있지 않기 때문에 본 연구에서는 이를 수동적으로 라벨링하여 사용하였다.

Methods	Existence	Temporal Order	Order and Effect Type	Run Time
	F_1 Score	F_1 Score	F_1 Score	
CETS (with NSE)	**0.5039**	**0.4511**	**0.4195**	2931s
CETS (without NSE)	0.4853	0.4188	0.3894	2922s
Pearson correlation	0.3568	-	0.2595 (Only Effect Type)	**17s**

Table 1: Result in SMD dataset.

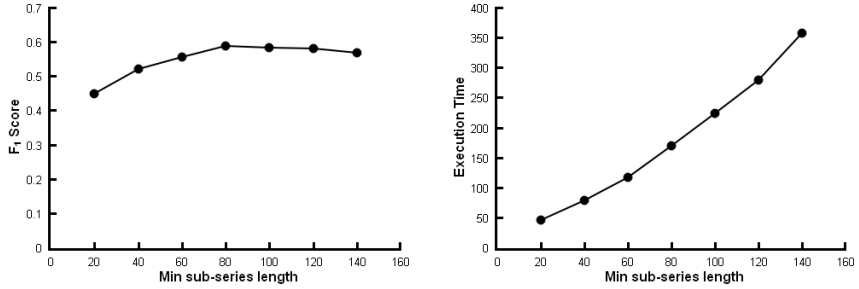

Figure 6: (a) F_1 score and (b) execution time versus min sub-series length L_{min}. ($L_{max} = 2L_{min}$)

4.3 결과 및 분석

Table 1에서는 CETS($L_{max} = 200$, $L_{min} = 80$, $\alpha = 2.58$)과 Pearson correlation의 성능 측정 결과를 도시하고 있다. CETS 알고리즘은 가설 검정에 기반을 두고 있으므로 F_1 score [8]를 성능 척도로 사용하였다. F_1 score는 이하와 같이 계산된다.

$$F_1 = \frac{2 * TruePositive}{2 * TruePositive + FalsePositive + FalseNegative} \tag{12}$$

이때, TruePositive는 상관관계를 정확히 예측한 경우, FalsePositive는 상관관계를 예측하였으나 실제로는 상관관계가 존재하지 않는 경우, 마지막으로 FalseNegative는 실제로 상관관계가 존재하나 이를 예측하지 못한 경우를 의미한다.

Baseline인 Pearson 방식과 비교하였을 때, CETS는 모든 영역에서 Pearson보다 우수한 수치를 보였으며 시간적 순서까지 추정하는 모습을 확인할 수 있다. 또한, NSE를 적용한 경우에 NSE를 적용하지 않은 경우보다 다소 높은 F_1 score를 보임을 확인할 수 있다. 그러나, CETS는 높은 F_1 score를 보이는 반면 상당한 실행 시간을 나타내고 있다. 특히 SMD 데이터셋은 5주간 관측되었으며 그 길이가 $25000 \sim 30000$에 달하는 상당히 긴 시계열로 구성되어 있다. 또한, 이상 현상 이후 그 영향이 나타날 때까지의 지연 시간과 이상 현상의 영향이 긴 특성을 가져 정확한 판정을 위해서 높은 부분열 길이를 요구하고 이에 따라 실행 시간의 상당한 증가를 유발한다. 보다 자세하게는 CETS는 사건 데이터 크기 n과 부분열 길이 k에 대하여 $O(n^2k)$의 시간복잡도를 가진다. 이전 CETS 연구에서는 최적 부분열 길이 k가 20 미만인 데이터셋에 대하여 이루어졌으므로 시간적인 부분이 강조되지 않았으나, n과 k이 모두 큰 데이터셋에 대해서 CETS는 상당한 실행 시간의 증가를 겪을 것으로 보인다. 다만, 본 구현에서는 최근접 이웃 알고리즘을 구현할 때 각 샘플에 대하여 모든 샘플에 대한 거리를 반복적으로 구하였는데, 중복된 거리 계산을 줄이도록 구현하면 시간복잡도를 절반정도로 줄일 수 있다.

Figure 6의 (a)에서는 최소 부분열 길이 L_{min}에 따른 F_1 score를 나타내고 있다. 이때, $L_{max} = 2L_{min}$로 설정되었다. L_{min}가 증가함에 따라 F_1 score는 다소 증가하다가 L_{min}가 일정 이상으로 증가할 경우 큰 차이를 보이지 않으며 이후 감소하기 시작한다. Figure 6의 (b)에서는 L_{min}에 따른 실행 시간을 나타내고 있다. 그래프는 완전한 선형은 아니나 선형에 가까운 모습을 보이고 있다. 이러한 결과에서 적절한 최소 부분열 길이의 설정은 검정 결과가 지나치게 짧은 구간 내 일시적 변화에 의존하지 않게 하여 성능을 개선함을 알 수 있으며 L_{max}는 과도한 실행 시간의 증가를 방지하는 효과가 있음을 확인할 수 있다.

5 결론

CETS 연구에서는 시계열과 사건 시퀀스(event sequence) 사이의 상관성 분석을 다변량 2샘플 가설 검정 문제로 생각하고 최근접 이웃 알고리즘에 기반한 검정 통계량으로 가설을 검정하여 상관관계를 파악

하고자 시도하였다. 본 연구에서는 CETS 알고리즘을 구현하고 이상현상 감지 연구에서 사용된 SMD 데이터셋에 적용하여 그 성능을 Pearson correlation 방식과 비교하여 확인하였으며, 그 결과 F_1 score의 측면에서 CETS가 Pearson correlation 방식보다 훨씬 우수한 성능을 보임을 확인하였다. 그러나, 실행 시간의 측면에서 CETS는 Pearson 방식보다 현저히 느린 모습을 보였다. 이러한 문제는 CETS가 상관관계의 시간적 순서까지 검정하므로 추가적인 정보 제공에 따른 trade-off로 생각할 수도 있지만 이를 고려하여도 과도한 실행 시간을 보이는 것으로 생각된다.

이러한 높은 실행시간은 CETS의 실행 시간이 부분열 길이 k가 증가함에 따라 선형적으로, 혹은 그 이상으로 증가함에 따라 발생하는 것으로 보이며 부분열 길이는 데이터셋의 특성에 크게 의존한다. SMD 데이터셋은 길이가 $25000 \sim 30000$에 달하는 상당히 긴 시계열로 구성되어 있다. 또한, 이상 현상 이후 그 영향이 나타날 때까지의 지연 시간과 이상 현상의 영향이 긴 특성을 가져 정확한 판정을 위해서 높은 부분열 길이를 요구하고 이에 따라 실행 시간의 상당한 증가를 유발하는 것으로 보인다. 이에 따라, 기존의 CETS 알고리즘에 더하여 최대 부분열 길이 L_{max}를 추가적으로 설정하여 실행 시간의 과도한 증가를 막을 수 있게 설정하였다. 또한, SMD 데이터셋의 일부 데이터는 정상성(stationarity) 경향이 낮은 시계열을 포함하는데 이 경우 부분열 길이가 과소평가되어 짧은 구간 내 일시적 변화에 의존하게 되는 문제를 발생시킨다. 따라서, 이를 방지하기 위해 최소 부분열 길이 L_{min}를 추가적으로 설정하였다. 마지막으로, CETS는 부분열 샘플이 상당히 균일한 경우 부분열 샘플이 랜덤 샘플과 상당히 유사함에도 이를 통계적으로 다르다고 판정하는 문제가 존재함을 확인하였다. 이에 따라, 모든 샘플 간 거리의 최대와 최소의 차가 일정 미만일 경우 두 샘플 집단을 동일시하는 인접 샘플 동일시(NSE)를 적용하였고 SMD 데이터셋에 대하여 F_1 score의 개선을 확인하였다.

그러나, 이러한 개선에도 CETS는 여전히 $O(n^2k)$라는 높은 시간 복잡도를 보이며 사건의 수가 증가할 경우 실행 시간의 급격한 증가가 예상된다. 이를 해결하기 위해서는 사건 군집화(clustering)을 통한 사건 데이터 크기 n의 감소 등 추가적인 연구가 필요할 것으로 보인다. 또한, 이상 현상 발생에 따른 시계열의 변화가 긴 지연 시간 이후에 발생하는 경우에 CETS는 상관관계를 잘 추정하지 못하는 것으로 보인다. 특히 지연 시간의 파악을 위해서는 최소한 지연 시간 이상의 부분열 길이가 보장되어야 하는데, 이는 또다시 실행 시간의 증가로 이어질 수 있다. 따라서, 지연 시간 문제를 해결하기 위한 추후 연구 역시 필요하다고 사료된다.

References

[1] Tie-Yan Liu et al. Learning to rank for information retrieval. *Foundations and Trends® in Information Retrieval*, 3(3):225–331, 2009.

[2] Ping Liu, Yu Chen, Xiaohui Nie, Jing Zhu, Shenglin Zhang, Kaixin Sui, Ming Zhang, and Dan Pei. Fluxrank: A widely-deployable framework to automatically localizing root cause machines for software service failure mitigation. In *2019 IEEE 30th International Symposium on Software Reliability Engineering (ISSRE)*, pages 35–46. IEEE, 2019.

[3] Huasong Shan, Yuan Chen, Haifeng Liu, Yunpeng Zhang, Xiao Xiao, Xiaofeng He, Min Li, and Wei Ding. ϵ-diagnosis: Unsupervised and real-time diagnosis of small-window long-tail latency in large-scale microservice platforms. In *The World Wide Web Conference*, pages 3215–3222, 2019.

[4] Ya Su, Youjian Zhao, Chenhao Niu, Rong Liu, Wei Sun, and Dan Pei. Robust anomaly detection for multivariate time series through stochastic recurrent neural network. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2828–2837, 2019.

[5] TsingHuasuya. Omnianomaly. https://github.com/NetManAIOps/OmniAnomaly, 2021.

[6] David A Dickey and Wayne A Fuller. Distribution of the estimators for autoregressive time series with a unit root. *Journal of the American statistical association*, 74(366a):427–431, 1979.

[7] Toni Giorgino. Computing and visualizing dynamic time warping alignments in r: the dtw package. *Journal of statistical Software*, 31:1–24, 2009.

[8] David MW Powers. Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. *arXiv preprint arXiv:2010.16061*, 2020.