

모바일 스마트 장치 배터리의 잔여 시간 예측 향상을 위한 데이터 전처리 프레임워크

탁성우*

A Data Preprocessing Framework for Improving Estimation Accuracy of Battery Remaining Time in Mobile Smart Devices

Sungwoo Tak*

*Professor, School of Electrical and Computer Engineering, Pusan National University, Busan 46241, Korea

요 약

모바일 스마트 장치 배터리의 잔여 시간을 예측하기 위해 범용 통계적 회귀 기법을 적용한 경우, 배터리 잔량별 배터리 사용 시간의 편차가 커질수록 범용 통계적 회귀 기법의 예측 정확도가 낮아진다. 따라서 범용 통계적 회귀 기법의 예측 정확도를 향상시키기 위해서는 배터리 잔량별 배터리 사용 시간의 편차가 큰 원 측정 데이터를 가공 처리하여 정제된 데이터로 변환시키는 작업이 필요하다. 이에 본 논문에서는 원 측정 데이터를 정제된 데이터로 가공 처리하는 데이터 전처리 프레임워크를 제안하였다. 제안한 프레임워크를 통해 가공 처리하여 정제된 데이터를 범용 통계적 회귀 기법에 적용한 결과, 범용 통계적 회귀 기법의 예측 정확도가 향상됨을 확인하였다.

ABSTRACT

When general statistical regression methods are applied to predict the battery remaining time of a mobile smart device, they yielded the poor accuracy of estimating battery remaining time as the deviations of battery usage time per battery level became larger. In order to improve the estimation accuracy of general statistical regression methods, a preprocessing task is required to refine the measured raw data with large deviations of battery usage time per battery level. In this paper, we propose a data preprocessing framework that preprocesses raw measured battery consumption data and converts them into refined battery consumption data. The numerical results obtained by experimenting the proposed data preprocessing framework confirmed that it yielded good performance in terms of accuracy of estimating battery remaining time under general statistical regression methods for given refined battery consumption data.

키워드 : 스마트 장치, 데이터 전처리, 통계적 예측, 배터리 남은 시간, 예측 정확도

Keywords : Smart Devices, Data Preprocessing, Statistical Estimation, Battery Remaining Time, Estimation Accuracy

Received 4 January 2020, Revised 11 February 2020, Accepted 25 February 2020

* Corresponding Author Sungwoo Tak(E-mail:swtak@pusan.ac.kr, Tel:+82-51-510-2387)

Professor, School of Electrical and Computer Engineering, Pusan National University, Busan 46241, Korea

Open Access <http://doi.org/10.6109/jkiice.2020.24.4.536>

print ISSN: 2234-4772 online ISSN: 2288-4165

©This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.
Copyright © The Korea Institute of Information and Communication Engineering.

I. 서 론

시간에 따라 일정을 세우는 사용자는 스마트 장치의 남은 배터리 용량 정보뿐만 아니라 현재 남은 배터리 잔량에서 스마트 장치를 사용할 수 있는 남은 배터리의 잔여 시간 정보도 필요하다. 그러나 대부분의 기존 연구들은 모바일 스마트 장치의 남은 사용 시간 예측보다는 주로 배터리 자체의 남은 충전 용량 및 사이클에 대한 예측 연구를 수행하였다. 이러한 기존 연구들이 사용한 예측 기법을 살펴보면 다음과 같다.

[1]에서는 지수 및 다항식 회귀 모형을 사용하여 충전 횟수에 따라 감소되는 배터리 충전 가능 용량을 예측하였다. [2]와 [3]에서는 무향 칼만 필터 (Unscented Kalman Filter)와 확장 칼만 필터 (Extended Kalman Filter) 기반 배터리 충전 상태 예측 기법을 제안하였다. [4]에서는 순환 신경망을 사용하여 배터리의 충전 사이클을 예측하였다. [5]에서는 베이지안 추론 기반 예측 모형을 제안한 후 배터리의 남은 수명을 예측하였다. [6]에서는 배터리 상태 예측을 위해 입자 군집 최적화 (Particle Swarm Optimization) 기반 메타 휴리스틱 기법을 사용하였다. [7]에서는 몬테카를로 시뮬레이션기반 배터리의 남은 수명 예측 기법을 제안하였다.

지금까지 살펴 본 기존 연구들은 불규칙한 데이터 변화를 고려하기 위하여 복잡한 형태의 수식 혹은 데이터 모델을 제안하여 예측 성능을 향상시키고자 하였다. 또한 이로 인해 알고리즘의 계산 및 공간 복잡도가 증가한다. 또한 개별 문제에 적합한 예측 기법 자체에 대한 연구를 주로 수행하여, 다른 문제에 대해 적용하거나 변경하는 것이 어렵다. 본 논문에서는 이와 다르게, 기존 범용 통계적 회귀 기법을 그대로 활용하면서 예측 정확도를 향상시키고자 한다. 그러나 회귀 모형 예측 기법이 규칙적인 변화 유형을 가지는 시계열 데이터를 사용하는 경우, 비교적 정확도가 우수한 예측 결과를 생성한다. 불규칙적인 변화 유형을 가지는 시계열 데이터인 경우에는 과적합 문제로 인해 정확도가 낮은 예측 결과를 생성한다. 과적합은 데이터의 일시적인 큰 변화에 과도하게 편향되는 예측 결과를 생성하는 현상이다 [8]. 이에 본 논문에서는 불규칙적인 데이터 변화량을 가진 원 데이터를 정제된 데이터로 변환하는 데이터 전처리 프레임워크를 제안하였다.

II. 배터리 사용 시간 편차에 따른 통계적 회귀 기법의 성능 분석

Table. 1 Procedures of generating data

1.	procedure generatePDF ($\min_{BT}, \max_{BT}, \mu_{BT}, \sigma_{BT}$)
1.1	$w_{BT} \leftarrow \max_{BT} - \min_{BT}$
1.2	$\mu_{w_{BT}} \leftarrow (\mu_{BT} - \min_{BT}) / w_{BT}$
1.3	$\sigma'_{BT} \leftarrow (\sigma_{BT} / w_{BT})^2, \mu'_{BT} \leftarrow (\mu_{w_{BT}} / (1 - \mu_{w_{BT}}))$
1.4	$\beta' \leftarrow ((\mu'_{BT} / \sigma'_{BT}) - \mu'_{BT}^2 - 2\mu'_{BT} - 1) / (\mu'_{BT}^3 + 3\mu'_{BT}^2 + 3\mu'_{BT} + 1)$
1.5	$\alpha' \leftarrow \mu'_{BT} \times \beta'$
1.6	$y \leftarrow (x - \min_{BT}) / w_{BT}, \text{ where } \min_{BT} \leq x \leq \max_{BT}$
1.7	$BT_Y(y) \leftarrow \text{Beta}_Y(y, \alpha', \beta') / w_{BT}$
2.	procedure rejectionSamplingData ($BT_Y(y), C, N_{BT}$)
2.1	$Data_{BT} \leftarrow \emptyset, n \leftarrow 0, C \leftarrow 1, Z \leftarrow \max_{BT_Y(y)}$
2.2	$q(Y) \leftarrow \text{uniform}(0, Z), 0 \leq q(Y) \leq Z \leq 1$
2.3	while $i < N_{BT}$ {
2.4	$y \leftarrow q(Y), u \leftarrow \text{random}(0, Cq(y)),$ $\text{where } BT_Y(y) \leq Cq(y)$
2.5	if $u < BT_Y(y) / Cq(y)$ {
2.6	$BT_i \leftarrow (y/Z) \times w_{BT} + \min_{BT}$
2.7	$Data_{BT} \leftarrow Data_{BT} \cup \{BT_i\}$
2.8	$i \leftarrow i + 1$
2.9	}
2.10	}
3.	procedure validateSamplingData ($Data_{BT}, \epsilon_\mu, \epsilon_\sigma, \mu_{BT}, \sigma_{BT}, N_{BT}$)
3.1	$\bar{\mu}_{BT} \leftarrow \frac{1}{N_{BT}} \sum_{i=1}^{N_{BT}} BT_i, \text{ where } \forall_{BT_i} \in Data_{BT}$
3.2	$\bar{\sigma}_{BT} \leftarrow \sqrt{\frac{1}{N_{BT}} \sum_{i=1}^{N_{BT}} (BT_i - \bar{\mu}_{BT})^2}, \text{ where } \forall_{BT_i} \in Data_{BT}$
3.3	$\overline{\Delta\mu}_{BT} \leftarrow \mu_{BT} - \bar{\mu}_{BT} $
3.4	$\overline{\Delta\sigma}_{BT} \leftarrow \sigma_{BT} - \bar{\sigma}_{BT} $
3.5	if ($\overline{\Delta\mu}_{BT} < \epsilon_\mu$ and $\overline{\Delta\sigma}_{BT} < \epsilon_\sigma$) accept $Data_{BT}$
3.6	else reject $Data_{BT}$

이 장에서는 배터리 사용 시간 편차를 임의 생성하는 모델을 기술한 후, 다양한 편차에 따른 통계적 회귀 기법의 성능을 분석하였다. 표 1은 배터리 잔량별 배터리

사용 시간 편차를 임의로 생성하는 절차이다. 단계 1에서 입력 값은 배터리 잔량별 최소 사용 시간(min_{BT})과 최대 사용 시간(max_{BT}), 배터리 잔량별 사용 시간의 평균 (μ_{BT}) 및 표준편차 (σ_{BT})이다. μ_{BT} 와 σ_{BT} 를 만족하는 확률 밀도 함수 ($BT_Y(y)$)를 생성하기 위해 베타 분포를 활용하였다.

$$B_Y(y, \alpha, \beta) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} y^{\alpha-1} (1-y)^{\beta-1} \quad (1)$$

$$\mu_Y = \frac{\alpha}{\alpha + \beta}, \sigma_Y = \frac{\alpha\beta}{(\alpha + \beta)^2(\alpha + \beta + 1)} \quad (2)$$

where $0 \leq y \leq 1, \Gamma(n) = (n-1)!$

수식 (1)과 (2)는 확률 변수 Y와 모수 α 및 β 를 가지는 베타 분포 밀도 함수 ($Beta_Y(y, \alpha, \beta)$)와 평균 (μ_Y) 및 표준편차 (σ_Y)를 보여준다. 수식 (1)의 베타 분포는 구간 $[0, 1]$ 범위 내에서 확률 분포를 생성한다. 따라서 구간 $[min_{BT}, max_{BT}]$ 내에서 확률 분포를 생성하기 위한 과정은 다음과 같다. 첫째, min_{BT} 를 원점으로 두고 $max_{BT} - min_{BT}$ 를 단위 측도 (동작 과정 1.1의 w_{BT})로 설정한 후, 구간 $[0, 1]$ 로 변환한다. 둘째, μ_{BT} 를 w_{BT} 단위로 변환한다. 마지막으로 μ_{BT} 와 σ_{BT} 도 구간 $[0, 1]$ 구간의 평균 (동작 과정 1.3의 μ'_{BT})과 표준 편차 (동작 과정 1.3의 σ'_{BT})로 변환한다. 수식 (2)의 μ_Y 와 σ_Y 에 μ'_{BT} 와 σ'_{BT} 를 대입하여 α' 와 β' 값을 유도한다 (동작 과정 1.4와 1.5). 마지막으로 배터리 잔량별 사용 시간 y 를 w_{BT} 단위로 변환한 후, 변환된 값에 대한 $BT_Y(y)$ 를 생성한다.

단계 2는 기각 표집 [9]을 사용하여 N_{BT} 개를 가진 배터리 사용 시간 데이터 ($Data_{BT}$)를 생성한다. 변수 C는 기각 표집의 실행 속도를 결정하는 인자이다. IV장의 성능 분석에서 C 값을 1로 설정하여 (동작 과정 2.1) 기각 표집의 실행 속도를 빠르게 하였다. 표집된 데이터는 $Data_{BT}$ 에 저장된다. Z는 $BT_Y(y)$ 의 최대값이다. i 는 표집된 데이터 개수를 나타낸다. $Data_{BT}$ 의 채택 유무 기준에 사용되는 함수는 $q(Y)$ 이다. $q(Y)$ 는 구간 $[0, Z]$ 에서 정의되는 연속형 균일 분포 (Uniform Distribution)이다 (동작 과정 2.2). 구간 $[0, Cq(Y)]$ 사이의 난수 (u)를 생성하고, $q(Y)$ 로부터 임의의 값 y 를 생성한다 (동작 과정 2.4). 생성 값 y 에 대한 채택 영역은 $[0, BT_Y(y)$

$/Cq(y)]$ 이며, 그 외의 지역은 기각역이 된다 (동작 과정 2.5). u 값이 채택 영역에 있는 경우 (동작 과정 2.5), y 를 i 번째 배터리 사용 시간 데이터 (BT_i)로 변환한다 (동작 과정 2.6). 변환 과정은 Z 대비 y 의 비율을 w_{BT} 에 곱한 후, 해당 결과 값을 min_{BT} 에 더하는 과정을 수행한다. 이와 같이 수행된 결과 값은 i 가 N_{BT} 에 도달할 때까지 $Data_{BT}$ 에 포함된다 (동작 과정 2.7).

단계 3은 단계 2에서 생성한 $Data_{BT}$ 의 평균 (동작 과정 3.1의 $\bar{\mu}_{BT}$) 및 편차 (동작 과정 3.2의 $\bar{\sigma}_{BT}$)와 단계 1의 μ_{BT} 및 σ_{BT} 간의 유사성을 확인한다 (동작 과정 3.5). $\bar{\mu}_{BT}$ 와 μ_{BT} 의 오차와 $\bar{\sigma}_{BT}$ 와 σ_{BT} 의 오차가 각각 허용 오차 ϵ_μ 와 ϵ_σ 보다 작으면, $Data_{BT}$ 를 채택한다 (동작 과정 3.5). IV장의 성능 분석에서 ϵ_μ 와 ϵ_σ 를 0.01로 설정하였다.

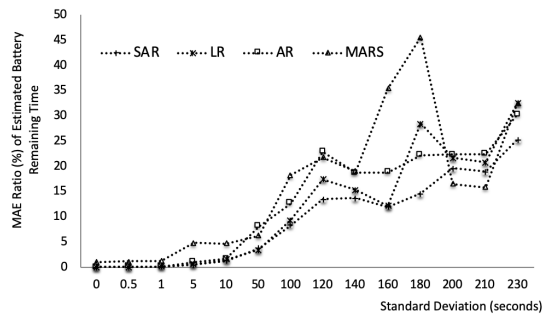


Fig. 1 MAE (Mean Absolute Error) ratio of estimated battery remaining time

그림 1은 통계적 회귀 기법의 예측 오차율을 보여준다. 사용한 데이터는 표 1의 동작 과정을 통해 생성하였다. II장과 IV장의 성능 평가에서 사용한 스마트 장치는 배터리 측정이 용이한 구글 넥서스 스마트 폰을 사용하였다 [10]. 사용한 회귀 기법은 단순 평균 회귀 (SAR: Simple Average Regression), 선형 회귀 (LR: Linear Regression), 자기 회귀 (AR: Auto Regression), 그리고 다변수 적응 회귀 (MARS: Multivariate Adaptive Regression Splines) 기법이다 [11]. 그림 1에서 스마트 폰의 이용률 (Utilization)은 100%이고, 실행되는 서비스와 장치 사용 시간이 임의로 설정한 환경에서 측정된 μ_{BT} 는 91.5 초이다. 총 배터리 사용 시간은 2시간 29분 38초이다. 다양한 σ_{BT} 값에 따른 14개의 배터리 사용 데이터를 생성

하였다. 그림 1의 가로축은 14개의 σ_{BT} (0초부터 230초까지)를 나타낸다. 세로축은 배터리 잔량별 예측 배터리 잔여 시간에 대한 평균 절대 오차율 MAE_{RBT} (MAE: Mean Absolute Error)를 나타낸다.

$$MAE_{RBT} = \sum_{i=J}^k \left| \widehat{RBT}_i - RBT_i \right| / (J - k + 1) \quad (3)$$

$$RBT_i = BT_{J,1} - BT_{J,i+1}, \widehat{RBT}_i = \widehat{BT}_{J,1} - BT_{J,i+1}$$

수식 (3)에서 RBT_i 와 \widehat{RBT}_i 는 각각 배터리 잔량 i 에서 측정 및 예측한 배터리 잔여 시간을 나타낸다. J 는 배터리 초기 잔량 100을 나타낸다. RBT_i 는 배터리 잔량 J 부터 1까지의 사용 시간 ($BT_{J,1}$)에서 배터리 잔량 J 부터 $i+1$ 까지의 사용 시간 ($BT_{J,i+1}$)을 차감한 값이다. \widehat{RBT}_i 는 배터리 잔량 J 부터 1까지의 예측 사용 시간 ($\widehat{BT}_{J,1}$)에서 $BT_{J,i+1}$ 을 차감한 값이다. 배터리 잔량 J 부터 $i+1$ 까지의 측정 사용 시간 ($BT_J \cdots BT_{i+1}$)을 통계적 회귀 기법의 입력 값으로 사용하여 $\widehat{BT}_{J,1}$ 를 계산한다. 그림 1에서 배터리 사용 시간이 동일하더라도 σ_{BT} 가 큰 배터리 데이터에서는 MAE_{RBT} 가 증가하였다.

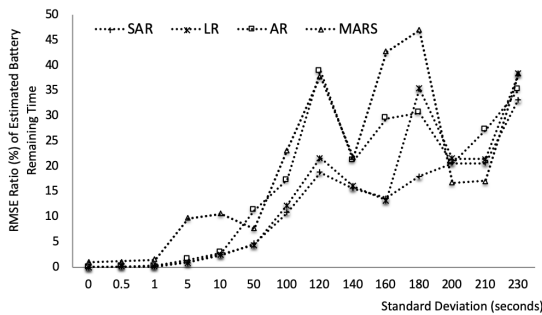


Fig. 2 RMSE (Root Mean Square Error) ratio of estimated battery remaining time

그림 2는 그림 1의 결과에 대한 평균 제곱근 오차 (RMSE: Root Mean Square Error)를 보여준다. 그림 2에서 σ_{BT} 가 120초와 180초인 경우, 평균 제곱근 오차가 높게 나왔다. 이러한 현상을 분석하기 위해, 그림 3에서는 σ_{BT} 가 50, 120, 160, 180, 그리고 210일 때, 배터리 잔량별 사용시간 변화를 분석하였다.

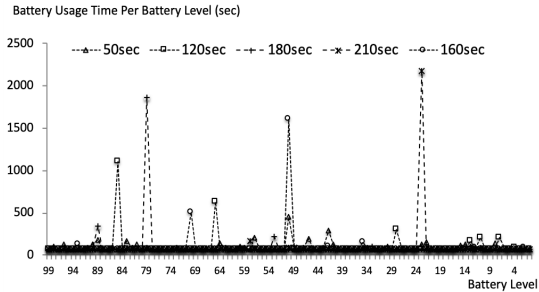


Fig. 3 Battery usage time per battery level

σ_{BT} 가 120초 및 180초인 경우, 잔량 84 및 79에서 배터리 사용시간이 최대이다. 배터리 사용 시간의 큰 편차가 초기에 발생하는 경우, 통계적 회귀 기법의 예측 오차율이 누적된다. 이로 인해 σ_{BT} 가 180초인 경우, 210초인 경우보다 높은 예측 오차율을 보여준다.

III. 데이터 전처리 프레임워크의 설계

이 장에서는 배터리 잔량별 배터리 사용 시간의 편차가 불균등한 데이터를 정제된 데이터로 가공 처리하는 데이터 전처리 프레임워크를 기술하였다. 프레임워크는 필터링 기반 데이터 전처리 결정 정책과 전처리 작업 절차로 구성된다.

3.1. 필터링 기반 데이터 전처리 결정 정책

필터링 기반 데이터 전처리 결정 정책은 전체 데이터에 대한 Coarse-Grained (큰 가공) 처리 결정 (표 2의 동작 과정)과 개별 데이터에 대한 Fine-Grained (미세 가공) 처리 결정 (표 3의 동작 과정)으로 구성된다. 표 2는 Coarse-Grained 처리 결정의 동작 과정을 보여준다. 통계적 회귀 기법 (동작 과정 1.1)의 SRM: Statistical Regression Method)은 $BT_J \cdots BT_{i+1}$ 을 사용하여 배터리 잔량 J 부터 i 까지의 예측 사용 시간 ($\widehat{BT}_J \cdots \widehat{BT}_i$)을 계산한다 (동작 과정 1.1). 그리고 이 값들의 합을 계산하여 배터리 잔량 J 부터 i 까지의 전체예측 사용 시간을 계산한다 (동작 과정 1.2의 $\widehat{BT}_{J,i}$).

Table. 2 Coarse-grained processing decision

1. procedure Coarse-grained processing decision():

1.1 $\widehat{BT}_J, \widehat{BT}_i, \dots, \widehat{BT}_i \leftarrow SRM[\{BT_J, \dots, BT_{i+1}\}]$

1.2 $\widehat{BT}_{J,i} \leftarrow \sum_{k=i}^J \widehat{BT}_k$

1.3 if $(|\widehat{BT}_{J,i} - BT_{J,i}| > T_{hreshold})$
 $BT_{Filter} \leftarrow BT_{Filter} \cup \{BT_J, \dots, BT_{i+1}\}$

1.4 return BT_{Filter}

해당 데이터의 예측값과 측정값의 차이가 발생하는 경우, 측정값의 가공처리를 실행한다. $T_{hreshold}$ 는 예측 사용 시간과 측정 사용 시간의 임계 허용 시간 차이이다 (동작 과정 1.3). IV장의 성능 분석에서 $T_{hreshold}$ 를 $BT_{J,i}$ 값의 1%로 설정하였다. $BT_{J,i}$ 와 $\widehat{BT}_{J,i}$ 간의 절대 오차가 $T_{hreshold}$ 보다 큰 경우 (동작 과정 1.3), $BT_J \dots BT_{i+1}$ 을 데이터 전처리 저장소 (BT_{Filter})에 포함시킨다 (동작 과정 1.3). BT_{Filter} 에 포함된 데이터는 3.2절에서 기술한 데이터 전처리 작업을 통해 가공 처리된다.

표 3은 개별 데이터에 대한 Fine-Grained 처리 결정 과정을 보여준다. 먼저 표 2의 동작 과정 1.1 및 1.2가 실행된다 (동작 과정 1.1). $BT_J \dots BT_{i+1}$ 의 평균 (μ)과 표준편차 (δ)를 계산한다 (동작 과정 1.2). 그리고 결정 규칙에 사용할 결정 함수를 설정한다 (동작 과정 1.3). 결정 함수는 정적 결정 (단계 2의 StaticDecision()), 로지스틱 결정 (단계 3의 LogisticDecision()), 오차 결정 (단계 4의 erfDecision()), 그리고 하이퍼볼릭 탄젠트 결정 (단계 5의 tanhDecision()) 함수들 중에 1개가 선택된다. 이러한 결정 함수들은 입력 데이터를 2개의 집단으로 분류하는데 많이 사용되어져 왔다 [12]. 배터리 잔량을 나타내는 변수 k 를 배터리 초기 잔량 J 로 설정하고, 저장소 (BT_{Filter})를 초기화한다 (동작 과정 1.4). 배터리 잔량 k 는 Δ_k 만큼 감소시킨다. IV장의 성능 분석에서 Δ_k 를 0.1로 설정하였다 (동작 과정 1.10). BT_k 및 \widehat{BT}_k 와 μ 간의 편차 (동작 과정 1.6과 1.7의 δ_k 와 $\widehat{\delta}_k$)를 각각 계산한다.

Table. 3 Fine-grained processing decision

1. procedure Find-grained processing decision():

1.1 Execute the same steps as 1.1 and 1.2 steps of Table 2

1.2 $\mu \leftarrow E[X], \delta \leftarrow \sqrt{Var[X]}$, where $X = BT_J, \dots, BT_{i+1}$

1.3 $Decision() \leftarrow staticDecision() \vee logisticDecision() \vee tanhDecision() \vee erfDecision()$

1.4 $k \leftarrow J, BT_{Filter} \leftarrow \{ \}$

1.5 while $(|\widehat{BT}_{J,i} - BT_{J,i}| > T_{hreshold}) \wedge (k \geq (i+1)) \{$

1.6 $\delta_k \leftarrow BT_k - \mu$

1.7 $\widehat{\delta}_k \leftarrow \widehat{BT}_k - \mu$

1.8 $flag \leftarrow Decision(\mu, \delta, \delta_k, \widehat{\delta}_k, T_{hreshold})$

1.9 if $(flag == true)$
 $BT_{Filter} \leftarrow BT_{Filter} \cup \{BT_k\}$

1.10 $k \leftarrow k - \Delta_k$

1.11 }

1.12 return BT_{Filter}

2. function StaticDecision($\mu, \delta, \delta_k, T_{hreshold}$):

2.1 if $(|\delta_k - (\theta_k \cdot \delta + \varepsilon_k)| > T_{hreshold})$ return true
else return false

3. function LogisticDecision($\mu, \delta_k, \widehat{\delta}_k, T_{hreshold}$):

3.1 $\omega \leftarrow (\widehat{\delta}_k - \delta_k) / T_{hreshold}$

3.2 $P \leftarrow -A \cdot \left(B - \frac{1}{1 + e^{(-2C \cdot \omega)}} \right)$

3.3 $u \rightarrow uniform(0, 1)$

3.4 if $(|P| > u)$ then return true
else return false

4. function erfDecision($\mu, \delta_k, \widehat{\delta}_k, T_{hreshold}$):

4.1 Execute the same steps as step 3.1

4.2 $P \leftarrow \frac{2}{\sqrt{\pi}} \int_0^{C \cdot \omega} e^{-t^2} dt$

4.3 Execute the same steps as described in steps 3.3 and 3.4

5. function tanhDecision($\mu, \delta_k, \widehat{\delta}_k, T_{hreshold}$):

5.1 Execute the same steps as step 3.1

5.2 $P \leftarrow \frac{e^{C \cdot \omega} - e^{-C \cdot \omega}}{e^{C \cdot \omega} + e^{-C \cdot \omega}}$

5.3 Execute the same steps as described in steps 3.3 and 3.4

$T_{hreshold}, \mu, \delta, \delta_k$, 그리고 $\widehat{\delta}_k$ 를 사용하는 결정 함수가 호출된다 (동작 과정 1.8). 결정 함수는 BT_k 를 데이

터 전처리 저장소 (BT_{Filter})에 포함시킬지를 결정한다. 단계 2에서 정적 결정 함수는 δ 와 δ_k 간의 차이가 $T_{hrshold}$ 보다 크면 필터링 작업이 필요함을 명시한다 (동작 과정 2.1: return true). θ_k 와 ε_k 는 δ 에 대한 가중치 및 BT_k 값의 허용 오차를 나타낸다 (동작 과정 2.1). IV장의 성능 분석에서 θ_k 와 ε_k 를 1과 0으로 설정하였다.

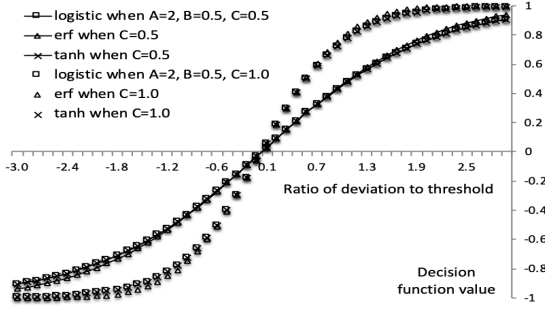


Fig. 4 Acceptance probability of BT_k inclusion in BT_{Filter} over varying ω values

그림 4는 ω 값 (가로축)에 따른 결정 함수 P 값 (세로 축)을 보여준다. ω 값은 $T_{hrshold}$ 대비 δ_k (동작 과정 1.6의 BT_k 와 μ 간의 편차)와 $\hat{\delta}_k$ (동작 과정 1.7의 \widehat{BT}_k 와 μ 간의 편차) 간의 차이에 대한 비율을 나타낸다. 결정 함수 P 값은 원 측정 데이터 BT_k 를 가공 처리가 필요한 저장소 (BT_{Filter})에 포함시킬 확률로 사용된다 (동작 과정 3.2, 4.2, 그리고 5.2). 음 확률 P 에 대해서는 절대값을 사용한다 (동작 과정 3.4). 결정 함수의 기울기 변화율을 크게 하기 위해서는 C 값을 증가시킨다. 결정 함수의 변화율이 증가할수록 높은 P 값을 생성할 경우도 증가하여 BT_k 가 BT_{Filter} 에 포함될 확률은 높아진다 (동작 과정 3.4, 4.3, 그리고 5.3). IV장의 성능 분석에서 C 값이 0.5인 로지스틱 결정 함수를 사용하였다. 마지막으로 확률 P 가 균등 분포를 통해 생성된 u 보다 크면, 해당 BT_k 는 데이터 전처리 작업 절차가 필요함을 명시한다 (동작 과정 3.4: return true).

3.2. 필터링 기반 데이터 전처리 작업 절차

제한한 데이터 전처리 작업 절차를 요약하면 다음과 같다. 첫째, 측정 데이터의 평균과 분산을 모수로 사용하는 분포 함수를 통해 배터리 잔량별 사용 가능 시간을

추정하는 모수적 추정 수식 모델을 유도한다. 이 모델을 통해 생성한 모수적 추정 사용 시간과 측정 사용 시간 간의 편차 (수식 (7)의 α_{BT_i})를 계산한다. 둘째, 측정 데이터로부터 배터리 잔량별 사용 시간에 대한 확률 밀도 추정을 생성하여 배터리 잔량별 사용 가능 시간을 추정하는 비모수적 추정 모델을 유도한다. 이 모델을 통해 생성한 비모수적 추정 사용 시간과 측정 사용 시간 간의 편차 (수식 (8)의 β_{BT_i})를 계산한다. 마지막으로, 모수적 및 비모수적 편차 값을 통해 유도한 신뢰도를 (수식 (13)) 사용하여 정제된 데이터를 생성한다.

$$BT_{J,i}^- = \widehat{BT}_{J,i+1}^+ + \mu_{BT_{J,i+1}} + \pi_i, \quad (4)$$

$$\text{where } \pi_i = \rho_i \cdot \delta_{BT_{J,i+1}}, \widehat{BT}_{J,i+1}^+ = \sum_{k=i+1}^J \widehat{BT}_k^+$$

모수적 추정 모델 (수식(4)부터 (7)까지)에서는 배터리 잔량 J 부터 i 까지 사용 가능한 모수적 추정 시간 (수식 (4)의 $BT_{J,i}^-$)을 유도한다. 모수적 추정 모델에서는 $BT_{J,i}^-$ 가 측정 데이터의 평균과 표준 편차에 가중치 (수식 (5)의 ρ_i)를 곱한 범위 내에서 변화할 것이라고 추정한다. 수식 (4)는 배터리 잔량 J 부터 $i+1$ 까지 필터링한 개별 사용 시간 (수식 (4)의 \widehat{BT}_k^+ 이며, 수식 (13)에서 유도한 \widehat{BT}_i^+ 를 나타낸다. 수식 (4)에서 합 ($\sum_{k=i+1}^J \widehat{BT}_k^+$)을 표현하기 위해 첨자 i 대신 k 를 사용한다)을 합한 값 ($\widehat{BT}_{J,i+1}^+$)과 배터리 잔량 J 부터 $i+1$ 까지의 평균 배터리 사용 시간 ($\mu_{BT_{J,i+1}}$) 및 편차 ($\delta_{BT_{J,i+1}}$)를 이용하여 $BT_{J,i}^-$ 를 계산한다. ρ_i 는 $\delta_{BT_{J,i+1}}$ 에 대한 가중치를 나타낸다. ρ_i 와 $\delta_{BT_{J,i+1}}$ 를 곱한 값은 π_i 로 설정된다.

$$\rho_i = \text{random}(Z_{pdf}(\mu_{BT_{J,i+1}}, \delta_{BT_{J,i+1}})), \quad (5)$$

$$\text{where } Z_{pdf}(\mu_{BT_{J,i+1}}, \delta_{BT_{J,i+1}}) \leftarrow \left\{ \frac{BT_J - \mu_{BT_{J,i+1}}}{\delta_{BT_{J,i+1}}}, \dots, \frac{BT_{i+1} - \mu_{BT_{J,i+1}}}{\delta_{BT_{J,i+1}}} \right\}$$

수식 (5)에서는 $\delta_{BT_{J,i+1}}$ 대비 측정 데이터 (BT_J, \dots, BT_{i+1})와 $\mu_{BT_{J,i+1}}$ 간 차이에 대한 확률 밀도 함수 (Z_{pdf})를 사용한다. $\mu_{BT_{J,i+1}}$ 과 편차 $\delta_{BT_{J,i+1}}$ 는 Z_{pdf} 의 모수로 사용된다. Z_{pdf} 로부터 확률 밀도 값을 임의 추출

하여 가중치 (ρ_i)로 설정한다. IV장의 성능 분석에서 Z_{pdf} 를 표준 정규 분포로 설정하였다. $\delta_{BT_{J,i+1}}$ 가 매우 작으면, 측정값들이 $\mu_{BT_{J,i+1}}$ 근처에 있을 가능성이 높다. 임의 생성한 ρ_i 값이 큰 경우이라도 $\delta_{BT_{J,i+1}}$ 값이 매우 작기 때문에, π_i 는 비교적 작은 값을 가질 것이다. 따라서 $\delta_{BT_{J,i+1}}$ 가 매우 작으면, 수식 (4)의 $BT_{J,i}^-$ 는 $BT_{J,i+1}^+$ 에 $\mu_{BT_{J,i+1}}$ 근처에 더한 값으로 수렴할 가능성이 크다. 이와 반대로 $\delta_{BT_{J,i+1}}$ 가 매우 크면, 균등 분포에 근접하게 되어 N_{BT} 개의 측정 데이터가 거의 유사한 확률 밀도 값을 가진다. 따라서 수식 (4)의 ρ_i 는 $1/N_{BT}$ 에 가까운 값으로 설정된다. 따라서 $BT_{J,i}^-$ 는 $BT_{J,i+1}^+$ 에 $\mu_{BT_{J,i+1}}$ 및 편차의 평균이 더해진 값으로 수렴한다.

$$\frac{\min BT_{J,i+1} - \mu_{BT_{J,i+1}}}{\delta_{BT_{J,i+1}}} \leq \rho_i \wedge \rho_i \leq \frac{\max BT_{J,i+1} - \mu_{BT_{J,i+1}}}{\delta_{BT_{J,i+1}}},$$

where $\min BT_{J,i+1} = \min\{BT_J, \dots, BT_{i+1}\}$
 $\max BT_{J,i+1} = \max\{BT_J, \dots, BT_{i+1}\}$

수식 (6)은 ρ_i 의 제약 조건을 기술한다. 배터리 잔량 i 에서 실행될 스마트 장치의 서비스가 이전 배터리 잔량 J 부터 $i+1$ 까지 실행된 스마트 장치 서비스 중에 하나가 사용될 것이라고 기대할 수 있다. 이에 배터리 잔량 i 의 추정 사용 시간 (BT_i^-) 하한 및 상한 값은 이전 측정값들 중에서 최소값 ($\min BT_{J,i+1}$)과 최대값 ($\max BT_{J,i+1}$)으로 설정하였다. 배터리 잔량 i 에서 사용한 시간 (BT_i)은 수식 (4)에서 유도한 $BT_{J,i}^-$ 를 이용하여 수식 (7)과 같이 유도될 수 있다. 수식 (7)은 BT_i^- 가 BT_i 로부터 α_{BT_i} 만큼의 편차가 있음을 나타낸다.

$$BT_i = BT_i^- + \alpha_{BT_i}, \text{ where } BT_i^- = BT_{J,i}^- - BT_{J,i+1}^- \quad (7)$$

비모수적 추정 모델 (수식 (8)부터 (13)까지)에서는 배터리 잔량 J 부터 i 까지 사용 가능한 비모수적 추정 시간 (수식 (8)의 $\widehat{BT}_{J,i}$)을 유도한다. 배터리 잔량 J 부터 $i+1$ 까지 사용한 시간 ($BT_{J,i+1}$)과 평균 배터리 사용 시간 ($\mu_{BT_{J,i+1}}$), 그리고 수식 (12)에서 유도한 값 (β_{BT_i})를

사용한다. β_{BT_i} 는 $\mu_{BT_{J,i+1}}$ 로부터 얼마나 떨어져 있는지를 나타내는 예측 편차 값이다.

$$\widehat{BT}_{J,i} = BT_{J,i+1} + \mu_{BT_{J,i+1}} + \beta_{BT_i}, \quad (8)$$

$$\text{where } \widehat{BT}_i = \widehat{BT}_{J,i} - \widehat{BT}_{J,i+1}$$

$$BT_k' \leftarrow BT_k - \mu_{BT_{J,i+1}}, \forall k = J, J-1, \dots, i+1 \quad (9)$$

$$\widehat{f}_h(x_{BT_m}) = \frac{1}{Mh} \sum_{k=J}^{i+1} K\left(\frac{x_{BT_m} - BT_k'}{h}\right) \quad (10)$$

, where $M = J - i, K(u) \geq 0, \forall u$

수식 (9)는 $\mu_{BT_{J,i+1}}$ 과 BT_k 간의 차이 (BT_k')를 계산한다. 수식 (10)은 수식 (11)에서 기술한 x_{BT_m} 값에 대한 확률 밀도 함수 ($\widehat{f}_h(x_{BT_m})$)를 유도하여, BT_k' 에 대한 확률 밀도 값을 추정한다. 이를 위해 배터리 잔량 J 부터 $i+1$ 까지의 커널 함수 값들을 더한 후, 전체 데이터 개수 M 으로 나눈다. 수식 (10)에서 $K(u)$ 는 커널 밀도 함수를 나타낸다. V장의 성능 분석에서는 가우시안 분포를 $K(u)$ 로 사용하였다. 수식 (10)에서 변수 h 는 커널의 첨도 (Kurtosis) 크기를 결정한다 [13]. IV장의 성능 분석에서 1부터 10까지의 h 값 중에서 예측 오차율이 가장 작은 3을 사용하였다.

$$\chi_{BT_m} \leftarrow \widehat{f}_h(x_{BT_m}), \forall m = J, J-1, \dots, i+1 \quad (11)$$

$$\text{where } (\min BT_{J,i+1} - \mu_{BT_{J,i+1}}) \leq x_{BT_m} \leq (\max BT_{J,i+1} - \mu_{BT_{J,i+1}})$$

$$\beta_{BT_i} = \text{random}(BT_J', \dots, BT_{i+1}', \text{weight} = \{\chi_{BT_J}, \dots, \chi_{BT_{i+1}}\}) \quad (12)$$

수식 (11)은 ($\widehat{f}_h(x_{BT_m})$)를 사용하여 확률 밀도 값 (χ_{BT_m})을 생성한다. x_{BT_m} 는 $\min BT_{J,i+1}$ 및 $\max BT_{J,i+1}$ 와 $\mu_{BT_{J,i+1}}$ 간 차이 구간 내에서 있으며, 총 M 개 데이터 중에서 1개이다. 수식 (12)에서는 수식 (11)에서 생성한 $\{\chi_{BT_J}, \dots, \chi_{BT_{i+1}}\}$ 을 가중치로 사용하여 BT_J' 부터 BT_{i+1}' 까지 중에서 1개의 값을 임의 생성하여 수식 (8)의 β_{BT_i} 로 설정한다.

$$\begin{aligned} \widetilde{BT}_i^+ &= \left(1 - \frac{|\beta_{BT_i}^k|}{|\alpha_{BT_i}^k| + |\beta_{BT_i}^k|}\right) \cdot \widehat{BT}_i + \left(1 - \frac{|\alpha_{BT_i}^k|}{|\alpha_{BT_i}^k| + |\beta_{BT_i}^k|}\right) \cdot BT_i \\ &= \frac{|\alpha_{BT_i}^k|}{|\alpha_{BT_i}^k| + |\beta_{BT_i}^k|} \cdot \widehat{BT}_i + \frac{|\beta_{BT_i}^k|}{|\alpha_{BT_i}^k| + |\beta_{BT_i}^k|} \cdot BT_i \\ &= \widehat{BT}_i + \frac{|\beta_{BT_i}^k|}{|\alpha_{BT_i}^k| + |\beta_{BT_i}^k|} \cdot (BT_i - \widehat{BT}_i) \end{aligned} \quad (13)$$

수식 (13)에서는 모수적 추정 모델에서 유도한 편차(수식 (7)의 α_{BT_i})와 비모수적 추정 모델에서 유도한 편차(수식 (8)의 β_{BT_i})를 이용한다. 그리고 추정 (\widehat{BT}_i) 값에 대한 비신뢰율 ($|\beta_{BT_i}^k| / (|\alpha_{BT_i}^k| + |\beta_{BT_i}^k|)$) 및 측정 (BT_i) 값에 대한 비신뢰율 ($|\alpha_{BT_i}^k| / (|\alpha_{BT_i}^k| + |\beta_{BT_i}^k|)$)을 계산한다. 1에서 비신뢰율을 차감하여 유도한 신뢰율을 추정 및 측정값의 가중치로 곱해진 후, 정제된 데이터 \widetilde{BT}_i^+ 를 계산한다. IV장의 성능 분석에서는 k 를 1로 선택하여 선형적으로 계산하였다.

IV. 성능 분석

성능 평가 요소는 배터리 잔여 시간의 예측 오차율과 오차 시간이다. 제안한 기법 외에 기존 평균 필터(SAF: Simple Average Filter), 이동 평균 필터(MAF: Moving Average Filter), 그리고 지수 필터(ESF: Exponential Smoothing Filter) 기법에 대한 실험 및 성능 분석을 수행하였다 [11]. 성능 분석에서 사용된 약어들은 다음과 같다. NF (NF: No Filter)는 필터 기법 없이 측정 데이터를 그대로 사용하는 무필터 기법, CGF (Coarse-Grained Filtering)는 표 2에서 기술한 Coarse-Grained 처리 결정 규칙을 적용한 필터링 기법, FGSF (Fine Grained Static Filtering)와 FGLF (Fine Grained Logistic Filtering)는 표 3에서 기술한 로지스틱 결정 함수를 적용한 필터링 기법을 나타낸다. 실험 환경은 배터리 잔량별 서비스 사용 시간의 편차별 개별 배터리 사용량 정보를 각각 20개씩 생성한 후, 실험한 결과의 평균값을 사용하였다.

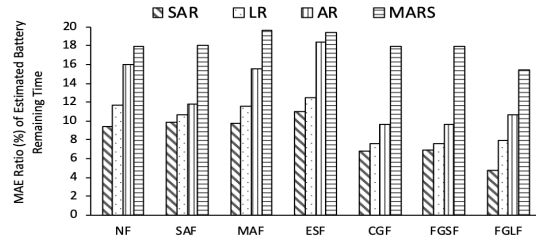


Fig. 5 MAE (Mean Absolute Error) ratio of estimated battery usage time

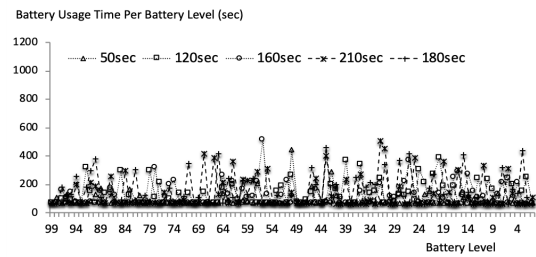


Fig. 6 Smoothed battery usage time per battery level

그림 5는 그림 1과 동일한 데이터를 적용한 후, 예측한 배터리 잔여 시간에 대한 평균 절대 오차율을 보여준다. 스마트 폰의 이용률은 100%이다. CGF, FGSF, 그리고 FGLF의 평균 절대 예측 오차율은 SAF, MAF, 그리고 EAF에 비해 낮았다. FGLF인 경우, 단순 평균 회귀(SAR)기법에서 매우 낮은 예측 오차율을 보여준다. 이는 필터링을 통해 정제된 데이터에 대해서는 단순한 평균 회귀 기법만으로도 우수한 예측이 가능함을 보여준다.

그림 6은 그림 3에서 보여준 결과 대비 FGLF에서 필터링한 배터리 잔량별 사용시간이 비교적 균등하게 분포되어 있음을 보여준다. 측정된 배터리 잔량별 배터리 사용 시간의 편차 (σ_{BT})가 50, 120, 160, 180, 그리고 210초인 경우, 정제된 데이터의 σ_{BT} 는 0.1%, 33%, 51%, 41%, 47% 만큼 감소하였다.

Table. 4 Mean absolute error ratio and battery remaining time estimation in four different utilizations

	Utilization: 10%			Utilization: 30%		
	SAR	LR	AR	SAR	LR	AR
NF	11.99	13.50	18.71	16.20	16.39	43.68
SAF	17.69	18.52	20.72	29.30	29.70	40.44
MAF	12.48	13.81	18.50	17.03	17.10	23.77
ESF	12.36	13.76	17.32	16.45	16.70	23.63

	Utilization: 10%			Utilization: 30%		
	SAR	LR	AR	SAR	LR	AR
CGF	6.51	7.91	13.54	12.56	11.26	19.81
FGSF	5.59	8.456	12.16	6.21	13.72	15.19
FGLF	5.60	7.66	12.0	5.49	10.26	13.56
	Utilization: 50%			Utilization: 70%		
	SAR	LR	AR	SAR	LR	AR
NF	8.14	9.84	43.70	10.44	12.09	25.69
SAF	9.80	11.02	49.73	12.94	14.02	16.55
MAF	7.75	9.47	42.68	10.27	11.92	18.52
ESF	8.98	10.46	64.61	10.88	12.41	17.60
CGF	7.55	8.69	12.26	13.97	13.01	17.87
FGSF	3.97	8.02	9.78	4.41	7.62	11.94
FGLF	3.10	6.26	7.19	3.95	7.09	9.62

표 4는 스마트 폰의 이용률이 다른 실험 환경에서 개별 배터리 잔량마다 예측한 배터리 잔여 시간에 대한 평균 절대 오차율에 대한 측정을 수행하였다. 이용률 10%, 30%, 50%, 그리고 70%에서 평균 배터리 사용 시간은 25시간 9분 58초, 8시간 23분 14초, 5시간 1분 56초, 그리고 3시간 35분 40초이었다. 전반적으로 모든 이용률에서 SAF, MAF, 그리고 ESF를 적용한 결과가 NF보다 우수하지 않았다. 제한한 FGSF와 FGLF는 CG보다 낮은 예측 오차율을 보여 주었다. FGSF와 FGLF를 적용한 경우, SAR만으로도 매우 낮은 예측 오차율을 보여 주었다. 이는 데이터 전처리 프레임워크를 통해 배터리 잔량별 사용 시간 편차가 큰 데이터를 평활시킨 경우, 범용 통계적 회귀 기법만으로도 우수한 성능을 생성할 수 있음을 보여준다.

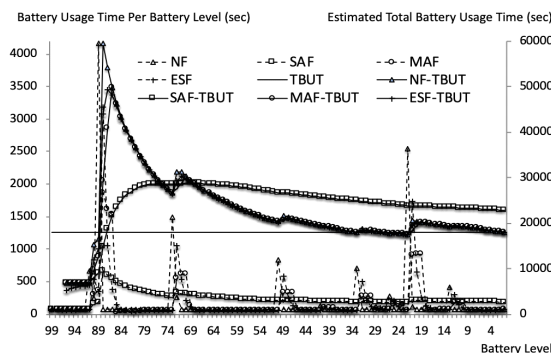


Fig. 7 Battery usage time per battery level and total battery usage time by NF, SAF, MAF, and ESF

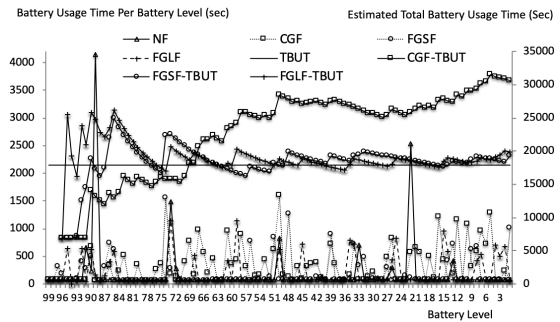


Fig. 8 Battery usage time per battery level and total battery usage time by NF, CGF, FGSF, and FGLF

그림 7과 8에서는 예측 오차율이 비교적 높은 데이터에 대한 기존 필터링 기법 및 제안한 기법의 성능을 비교하였다. 사용한 데이터의 스마트 폰 이용률은 50%, 배터리 잔량별 사용 시간의 평균 및 편차가 181초 및 500이다. 배터리 잔량 96부터 배터리 사용 시간을 예측하였다. 전체 배터리 사용시간 (TBUT: Total Battery Usage Time)은 5시간 1분 56초이다. 사용한 통계적 회귀 기법은 SAR이다. 그림 7과 8에서 배터리 잔량(가로축: Battery Level)별 배터리 사용 시간(왼쪽 세로축)과 필터링된 데이터를 적용하여 예측한 배터리 사용 시간(오른쪽 세로축: NF-TBUT, SAF-TBUT 등)을 보여준다.

그림 7에서 NF인 경우, 높은 침도를 가지는 잔량 89, 73, 50, 33, 그리고 22에서 사용 시간은 4156초, 1494초, 832초, 705초, 그리고 2533초이다. SAF인 경우, 잔량 89에서 최대 사용 시간 668초(왼쪽 세로축)를 보여준다. 잔량 89 이후, 사용 시간 값은 감소한다. MAF는 3개의 최근 측정 데이터에 대한 평균값으로 필터링한다. ESF에서 최근 및 이전 측정값에 각각 가중치 0.7 및 0.3을 부여하였다. 이에 MAF와 ESF로 필터링된 데이터들은 최근 측정값 근처에 존재한다. 따라서 NF와 유사하게 잔량 89, 73, 50, 33, 그리고 22에서 뾰족한 침도를 나타낸다. 이와 같이 정제된 데이터를 사용한 SAR도 잔량 89, 73, 50, 33, 그리고 22에서 뾰족한 침도 형태를 보여준다. NF, SAF, MAF, 그리고 ESF인 경우, 예측 오차율은 각각 35.53%, 44.24%, 32.87%, 그리고 34.14% 이었다.

그림 8에서 CGF인 경우, 약 10%정도가 다른 데이터에 비해 높은 배터리 사용시간 데이터들임에도 불구하고, 전체 데이터에 대하여 정제 작업을 수행한다. 이로 인해 오히려 예측 오차율이 증가하는 현상을 보여준다.

FGSF와 FGLF은 표 3의 $T_{threshold}$ 보다 큰 측정 데이터만 필터링 작업을 수행한다. 따라서 FGSF와 FGLF은 CGF보다 비교적 낮은 값을 가지는 데이터를 생성한다. CGF, FGSF, 그리고 FGLF인 경우, 예측 오차율은 각각 41.3%, 44.24%, 9.68%, 그리고 8.61% 이었다.

V. 결론

제안한 데이터 전처리 프레임워크를 요약하면 다음과 같다. 첫째, 배터리 잔량별 배터리 사용 시간의 편차가 작은 경우, 범용 통계적 회귀 기법의 예측 정확도는 매우 우수하였다. 둘째, 배터리 잔량별 배터리 사용시간의 편차가 불균등한 데이터를 정제된 데이터로 가공 처리한 경우, 단순 평균 회귀 기법을 사용하더라도 예측 오차율이 향상됨을 확인하였다. 이는 불균등한 측정 데이터가 정제된 데이터로 가공 처리되어진다면, 새로운 예측 기법 없이 기존 통계적 회귀 기법을 적용하여도 우수한 배터리 잔여 시간 예측 성능을 생성할 수 있다.

ACKNOWLEDGEMENT

This work was supported by a 2-Year Research Grant of Pusan National University

References

- [1] Y. Xing, E. Ma, K-L. Tsui, and M. Pecht, "An ensemble for predicting the remaining useful performance of lithium-ion batteries," *Microelectronics Reliability*, vol. 53, no. 6, pp.811-820, June 2013.
- [2] W. Wang, X. Wang, C. Xiang, C. Wei, and Y. Zhao, "Unscented kalman filter-based battery SOC estimation and peak power prediction method for power distribution of hybrid electric vehicles," *IEEE Access*, vol. 6, pp. 35957-35965, June 2018.
- [3] W. Yan, B. Zhang, G. Zhao, S. Tang, G. Niu, and X. Wang, "A battery management system with a lebesgue-sampling-based extended kalman filter," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 4, pp. 3227-3236, April 2019.
- [4] Y. Zhang, R. Xiong, H. He, and Z. Liu, "A LSTM-RNN method for the lithium-ion battery remaining useful life prediction," *Proceeding of 2017 Prognostics and System Health Management Conference*, China: Harbin, pp.811-820, July 2017.
- [5] R. Richardson, M. Osborne, and D. Howey, "Gaussian process regression for forecasting battery state of health," *Journal of Power Sources*, vol. 357, pp. 209-219, July 2017.
- [6] T. Mesbahi, F. Khenfri, N. Rizoug, K. Chaaban, P. Bartholomeus, and P. Moigne, "Dynamical modeling of Li-ion batteries for electric vehicle applications based on hybrid Particle Swarm - Nelder - Mead (PSO - NM) optimization algorithm," *Electric Power Systems Research*, vol. 131, pp. 195-204, Feb. 2016.
- [7] Y. Zhang, R. Xiong, H. He, and M. Pecht, "Lithium-ion battery remaining useful life prediction with box-cox transformation and Monte Carlo simulation," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 2, pp.1585-1597, Feb. 2019.
- [8] J. Wiley, *R Deep learning essentials*, 1th ed. Birmingham, Packt Publishing, 2016.
- [9] L. Martino, D. Luengo, J. Miguez, *Independent random sampling methods*, 1th ed. Switzerland, Springer, 2018.
- [10] A. Saksonov, "Method to derive energy profiles for android platform," Master thesis, University of Oldenburg, Oldenburg, Germany, 2014.
- [11] M. Kuhn and K. Johnson, *Applied predictive modeling*, 1st ed. New York, Springer, 2013.
- [12] K. Nikolay and M. Svetoslav, *Sigmoid functions: some approximation and modelling aspects*, 1th ed. Riga: Latvia, Lambert Academic Publishing, 2015.
- [13] A. Gramacki, *Nonparametric kernel density estimation and its computational aspects*, 1th ed. New York, Springer, 2018.



탁성우(Sungwoo Tak)

2003년 2월 미국미주리주립대학교
Computer Science 박사
2004년~현재 부산대학교 정보컴퓨터공학부 교수
(부산대 컴퓨터및정보통신연구소
겸임 연구원)
※ 관심분야: 유무선 네트워크, 위치인식,
통계적 예측 기법