

온라인 채널 제품 판매량 예측 AI 오프라인 해커톤

주혁이

강민규, 김권호, 장준보, 전주혁

Contents

-
- 01** **Intro**
 - 02** Analysis
 - 03** Model Algorithm
 - 04** Validation
 - 05** Application
-

대회 배경 및 목표



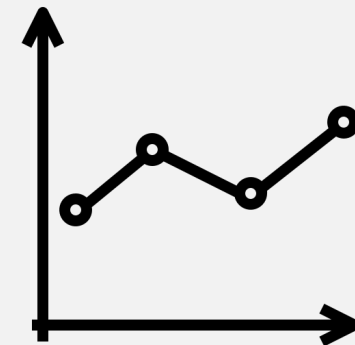
온라인 판매 채널 확대



효율적 재고 관리



타겟 마케팅 전략 수립



온라인 채널 제품 판매량 예측 AI 모델 개발

제공데이터

File name	Columns	Remarks
train	ID, 제품, 대분류, 중분류, 소분류, 브랜드, 쇼핑물	2022-01-01~2023-04-24 기간 실제 일별 판매량
sales	ID, 제품, 대분류, 중분류, 소분류, 브랜드, 쇼핑물	2022-01-01~2023-04-24 기간 실제 일별 판매 금액
brand_keyword_cnt	브랜드	브랜드의 연관키워드 언급량을 정규화한 일별 데이터
product_info	제품, 제품 특성	제품 특성 데이터

2023-04-25~2023-05-15까지의 ID별 판매량 예측 task

평가 지표

$$PSFA_m = 1 - \frac{1}{n} \sum_{day=1}^n \sum_i^N \left(\overbrace{\left(\frac{|y_i^{day} - p_i^{day}|}{\max(y_i^{day}, p_i^{day})} \right)}^{\text{예측 오차}} \times \overbrace{\left(\frac{y_i^{day}}{\sum_{i=1}^N y_i^{day}} \right)}^{\text{일별 판매 비중}} \right)$$

$$PSFA = \frac{1}{M} \sum_{m=1}^M PSFA_m$$

- m : 대분류 index
- i : 대분류 내 제품 index
- y_i^{day} : i 번째 제품의 day 일 판매량
- p_i^{day} : i 번째 제품의 day 일 예측 판매량

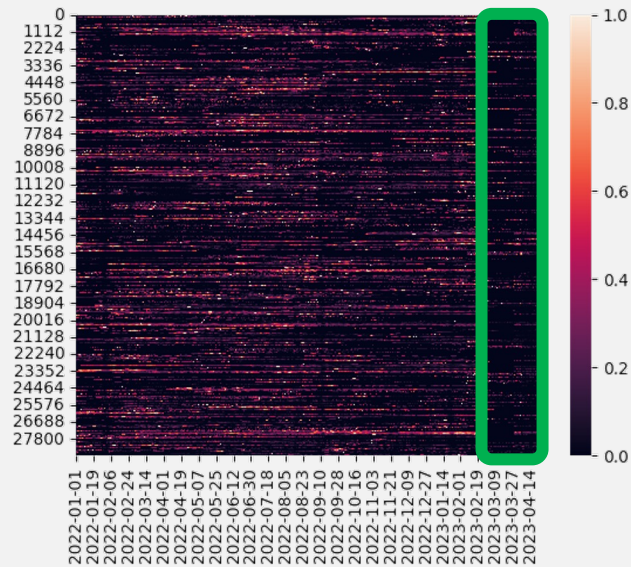
Our Approach

- Time Series
- Proper Inductive Bias
- Causal Feature Selection
- Consider Global & Local Property
- Robust Forecast

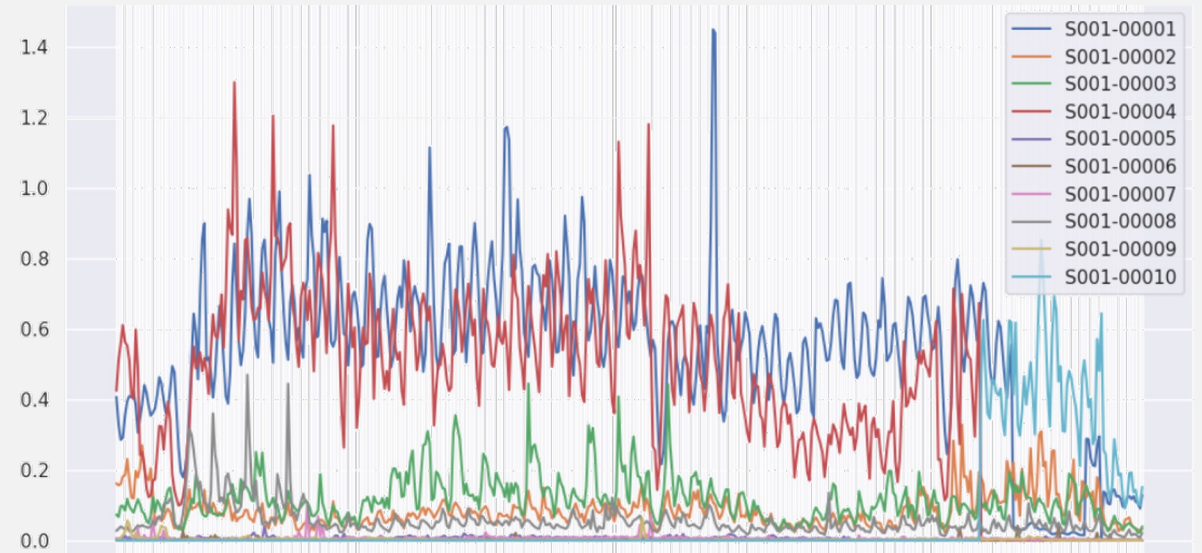
Contents

-
- 01 Intro
 - 02 Analysis**
 - 03 Model Algorithm
 - 04 Validation
 - 05 Application
-

EDA



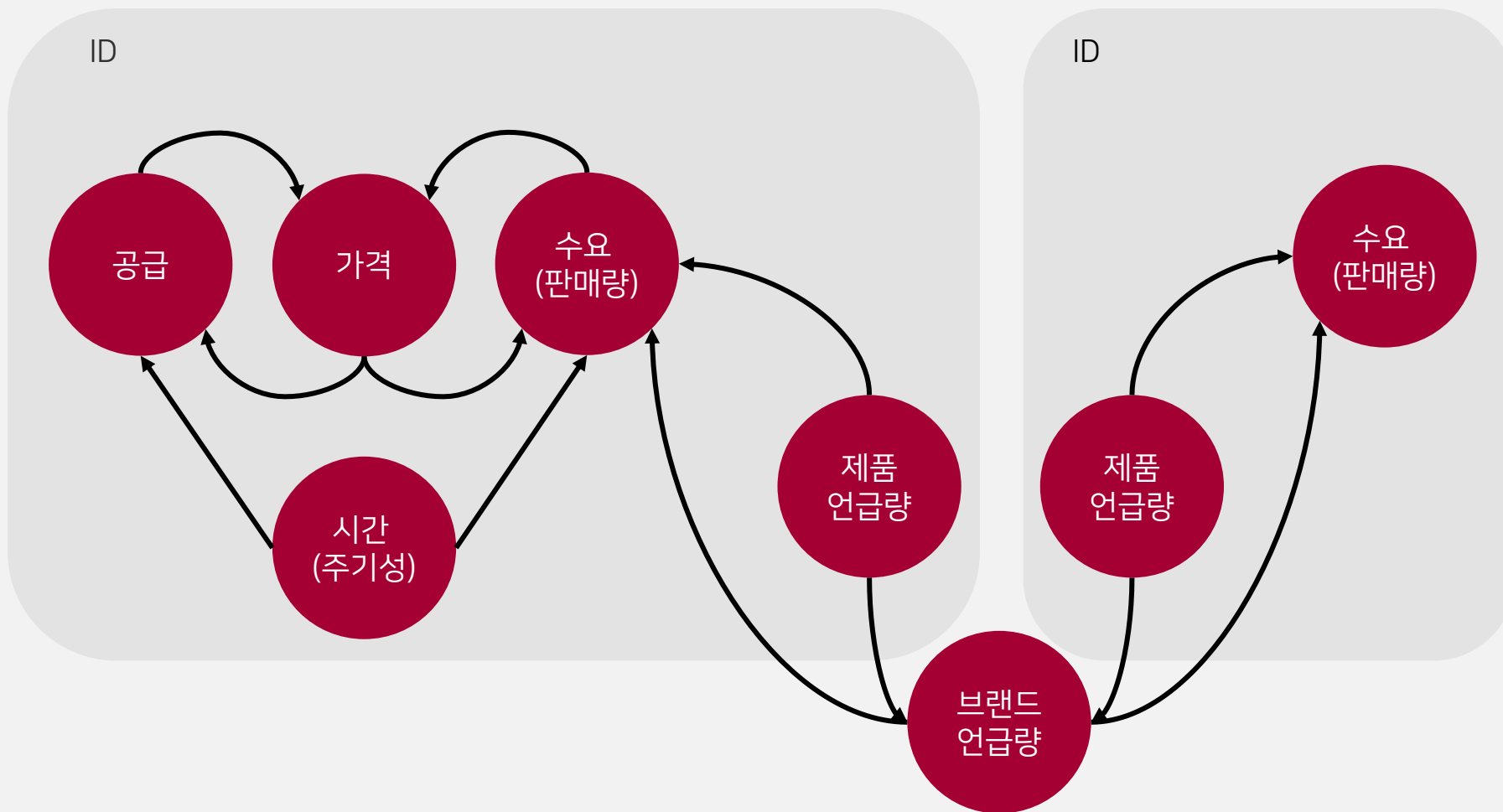
- 전체 data 중 0값이 약 65.4%를 차지함



- 쇼핑몰 별 다른 판매 양상 (추세, 주기성)
- 쇼핑몰 cluster내 분산/쇼핑몰 cluster간 분산 = 0.676

과도한 0 값 + 쇼핑몰간 이분산성
→ 학습시 방해요소로 작용

Causal graph (assumption)



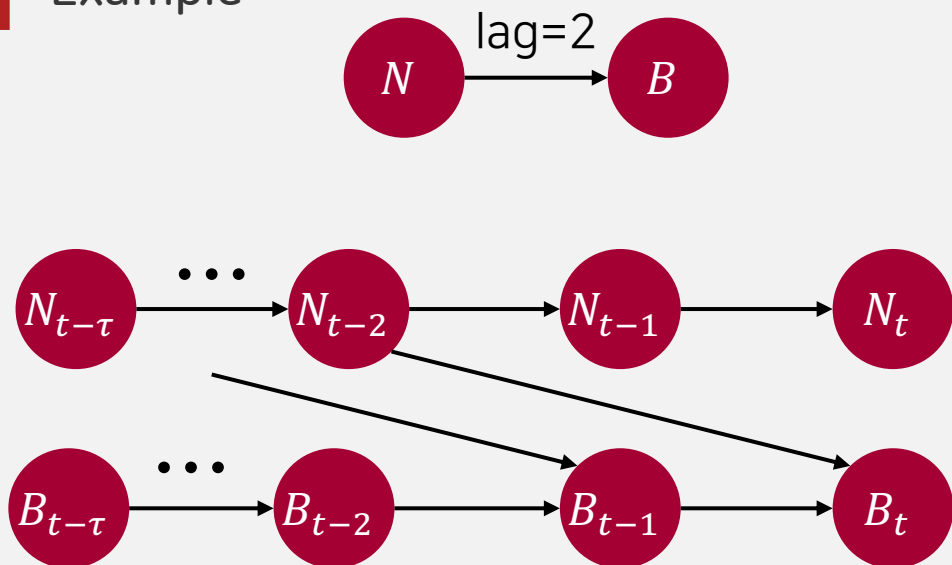
- ① 공급과 수요사이, 가격을 통한 cycle에 의해 수요 자체의 시계열적 특성 unidentifiable
- ② 브랜드 언급량으로 모델링 시 제품 언급량이 confounder로 작용해 정확히 브랜드 언급량이 수요에 미치는 영향 unidentifiable

Granger causality test

The series X is Granger non-causal for the series Y with respect to $\mathbf{V} = \{X, Y\} \cup \mathbf{Z}$ if $Y_{t+1} \perp X^t | Y^t, \mathbf{Z}^t$ for all $t \in \mathbb{Z}$

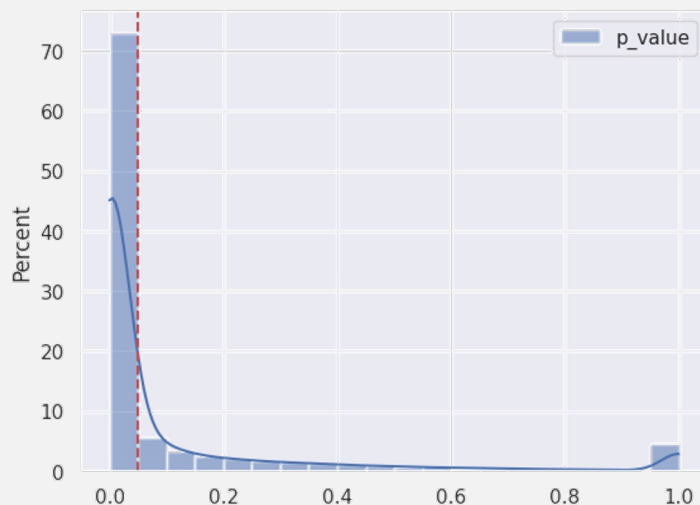
한 시계열이 다른 시계열에 대해 Granger causal하다는 의미는 해당 시계열의 과거 값이 다른 시계열의 미래 값을 예측할 때 도움이 되는 정보를 준다는 의미 (correlation 기반 test)

Example

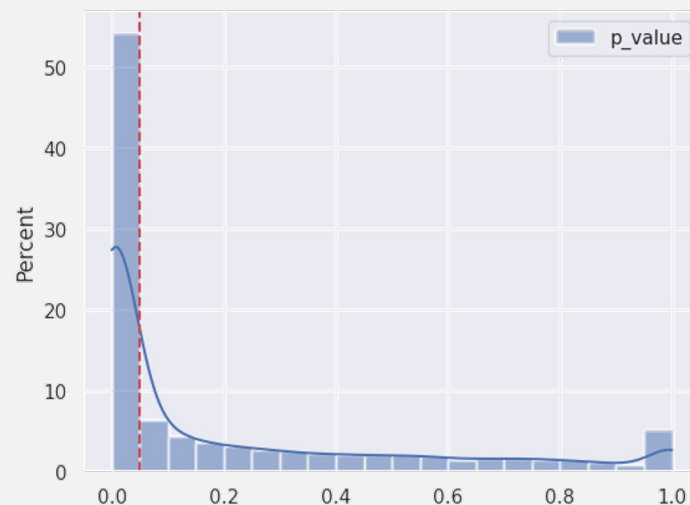


- Test $B_t \perp N_{t-\tau} | B^{t-1}, N^{t-1} \setminus \{N_{t-\tau}\}$ for different τ 's
- 이 경우 lag 즉, $\tau = 2$
- Our method : 14 lag까지 test하여 유의한 lag가 존재한다면, Granger cause 관계라고 판별.

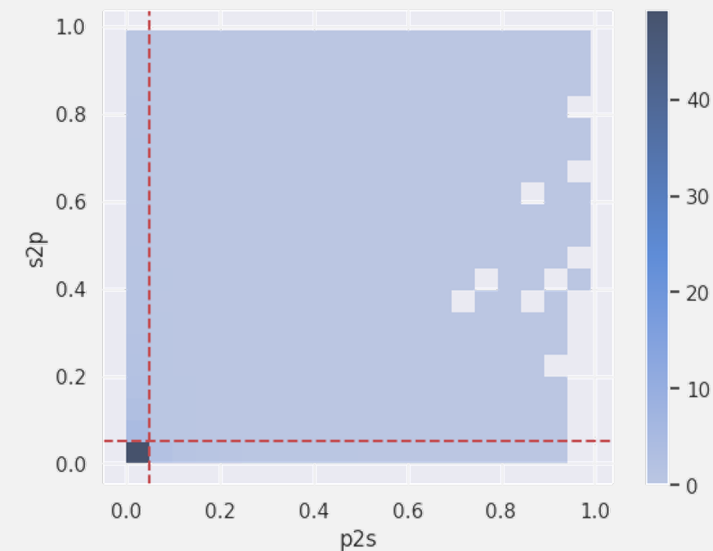
가격



price → sales test



sales → price

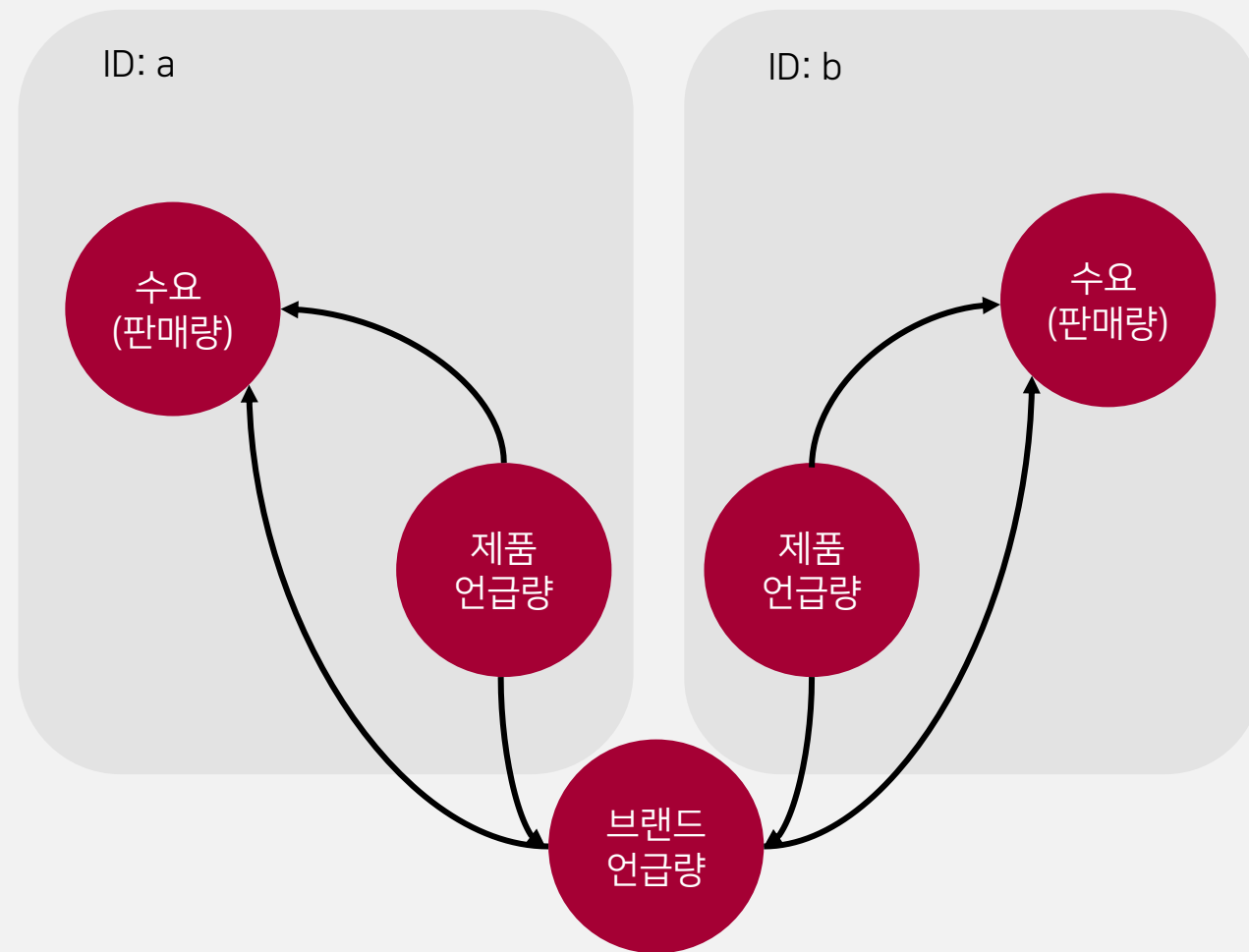


total

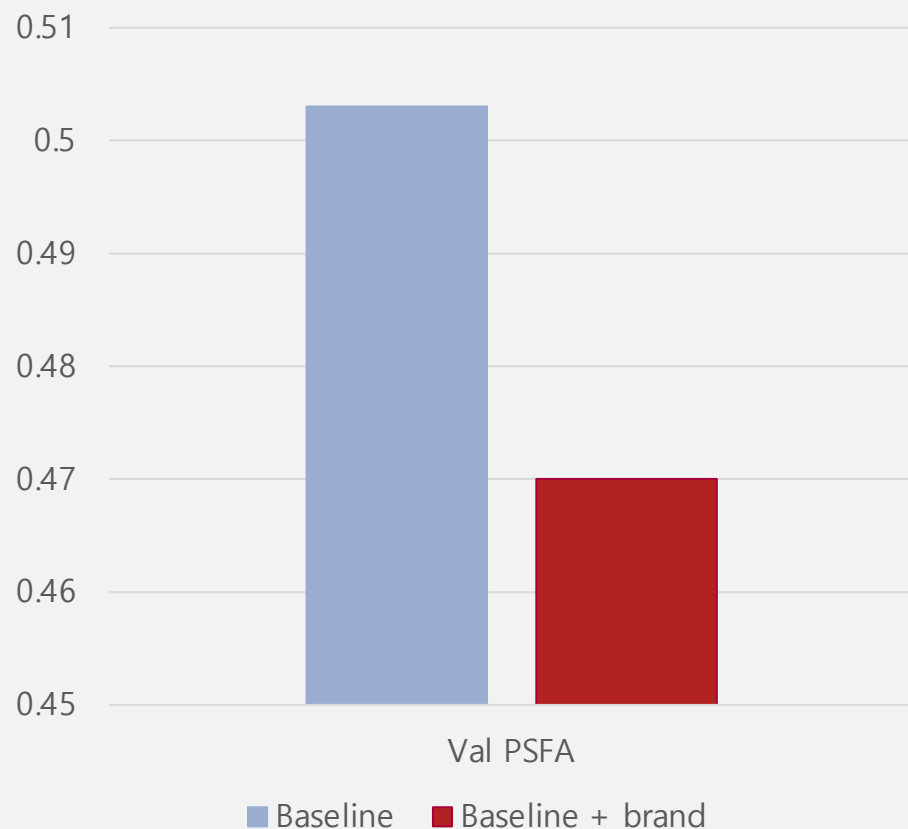
- 양방향 Granger causality test 결과 대부분의 제품에서 cycle이 확인되었음.
- price를 사용하여 모델링 한다면 수요 자체의 시계열적 특성에 noise로 작용할 것.
- price 제외 후 모델링

브랜드 언급량

- 브랜드 언급량이 모델에 포함될 경우, 브랜드 언급량에서 수요로 back door path를 통해 d-connect
- 브랜드 A의 상품 a의 언급량이 늘어 판매량이 늘어난 경우, 같은 브랜드 A의 다른 제품 b는 변화가 없음에도 영향받은 것으로 예측할 것
- 제품 언급량이 confounder로 작용하여 generalization에 악영향
- baseline LSTM에 기반하여 generalization 성능 평가



브랜드 언급량



- baseline code에서 기존 feature (제품 분류 코드 및 브랜드)에 브랜드 언급량을 쌓아 돌린 결과
- Baseline 성능이 0.50으로 feature를 쌓은 모델의 성능인 0.47보다 높았음
- Generalization에 브랜드 언급량 feature가 악영향을 미친다고 판단하고 제외 후 모델링

Contents

01 Intro

02 Analysis

03 **Model Algorithm**

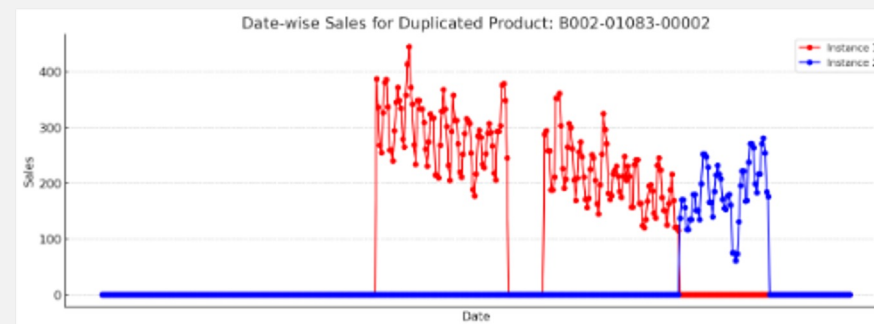
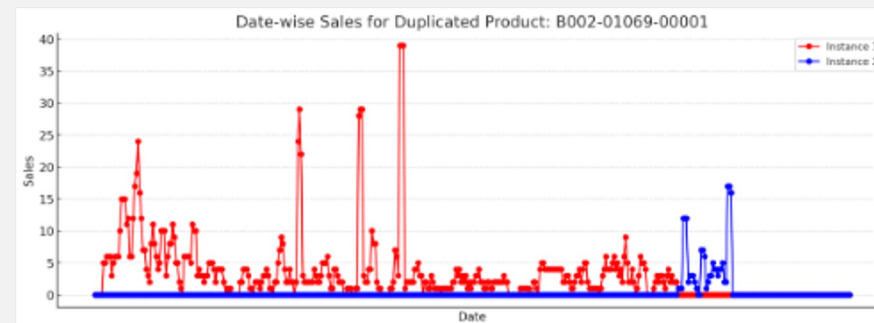
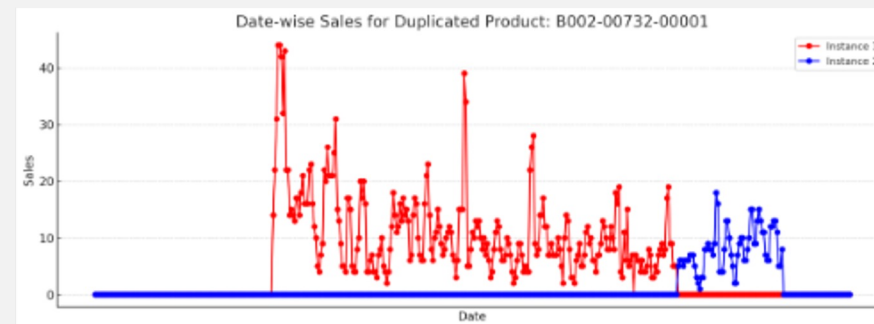
04 Validation

05 Application

Custom imputation v1

Hypothesis

- 동일 제품이라면, 채널에 관계없이 사람들의 수요는 유사할 것
- 동일 제품군끼리 판매량을 공유하는 method
- Model이 이러한 공통 패턴을 학습할 수 있도록 의도함



Inductive bias

① 수요 측면

$$PSFA_m = 1 - \frac{1}{n} \sum_{day=1}^n \sum_i^N \left(\underbrace{\left(\frac{|y_i^{day} - p_i^{day}|}{\max(y_i^{day}, p_i^{day})} \right)}_{\text{예측 오차}} \times \underbrace{\frac{y_i^{day}}{\sum_{i=1}^N y_i^{day}}}_{\text{일별 판매 비중}} \right)$$

- 실제 판매량이 0이라면, 판매 비중이 0이되어 예측오차와 상관없이 objective에 반영되지 않음.

→ PSFA 지표가 판매량(수요)의 존재를 implicit하게 보장

② 공급 측면

판매량이 0이 되는 이유

- 공급부족
- 해당 시기에 제품이 존재하지 않음
- 실제 판매가 되지 않은 경우

} 외부 요인
(noise)

∴ 공급과 수요는 항상 stationary함을 가정
즉, 어느정도 일정하게 유지됨을 가정

Custom imputation v2

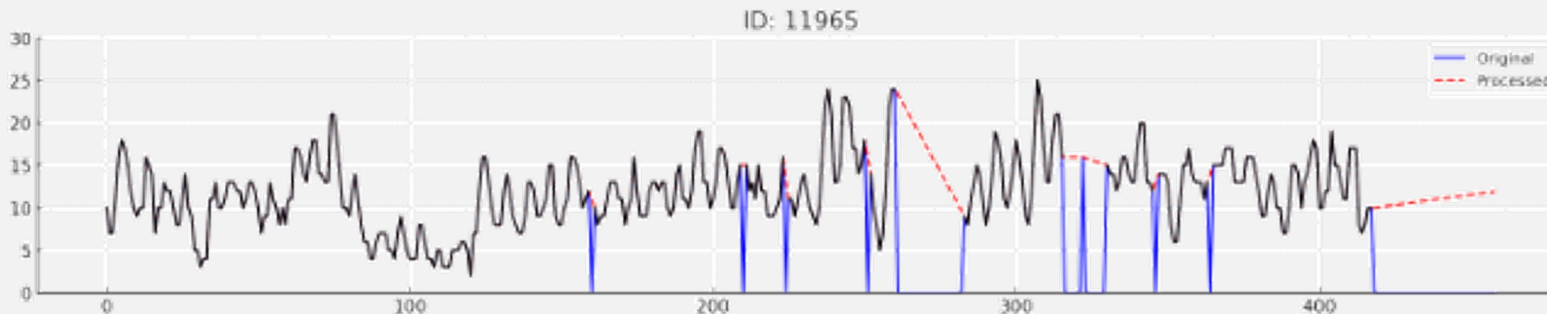
- ① 공급부족 —————> 0이 아닌 값들 사이에 갑자기 0인 경우
- ② 해당 시기에 제품이 존재하지 않음 —————> data의 시작과 끝 부분이 연속으로 0인 경우
- ※ 모든 data가 0인 경우는 그대로 사용

Imputation ①

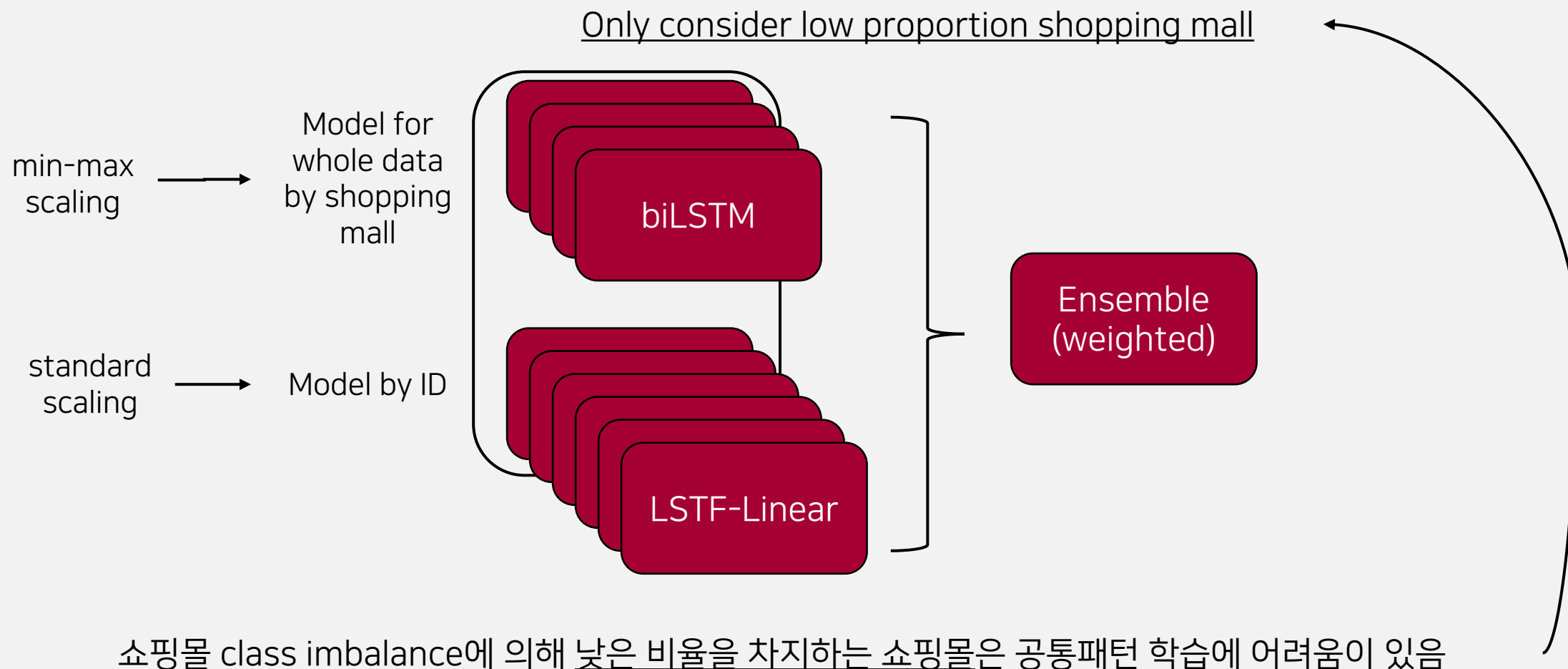
- 공급이 항상 존재한다는 가정 하에서 선형 보간

Imputation ②

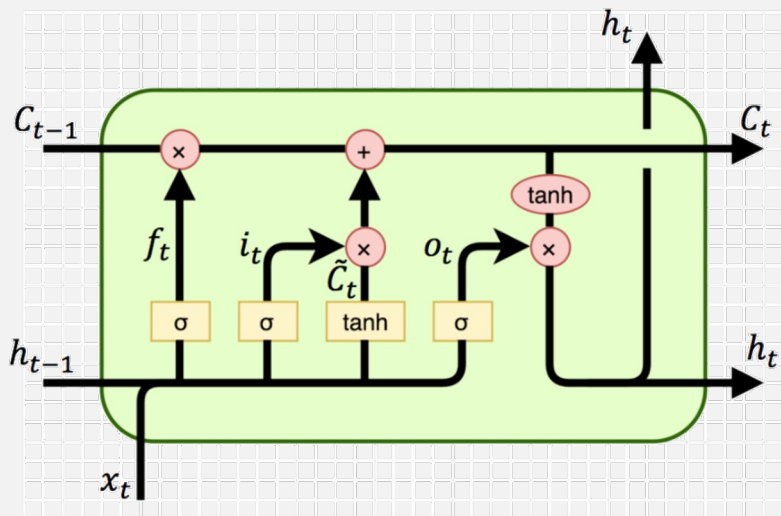
- 제품이 없는 구간도 수요가 존재한다는 가정하에서, 첫 번째로 0이 아닌값 혹은 마지막으로 0이 아닌 값과 0을 제외하고 계산한 중앙값사이를 선형 보간



Forecast process



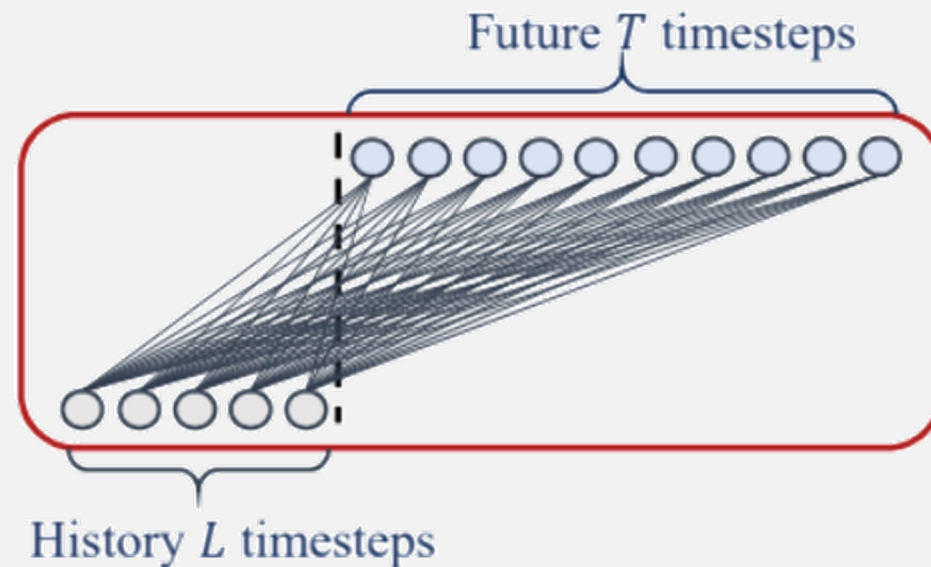
LSTM



(a) Long Short-Term Memory

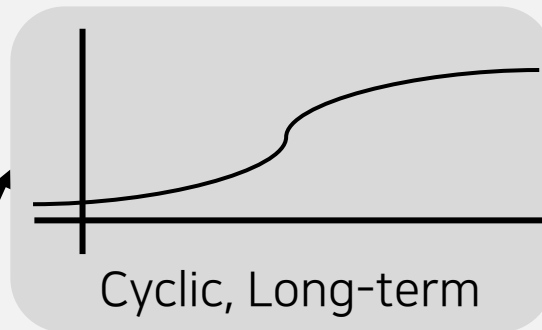
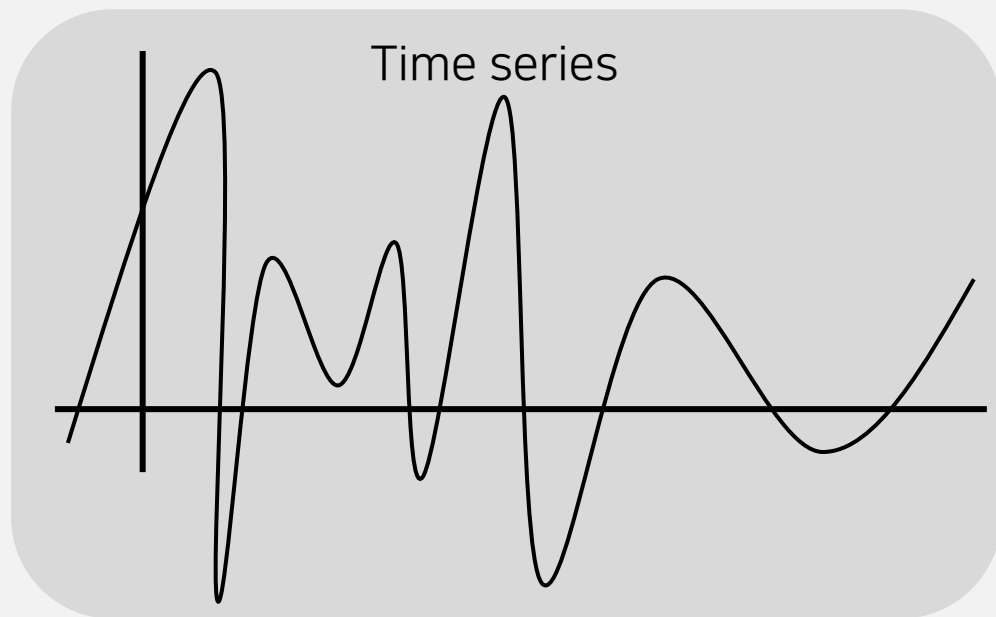
- 이전 정보를 장기간 기억할 수 있는 메모리 셀을 통해 긴 시퀀스 데이터를 처리할 수 있음.
- 장기 메모리와 단기 메모리에 들어갈 정보를 나누어 학습을 진행

LSTF-Linear

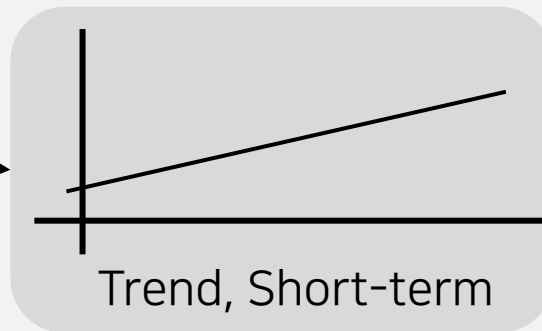


- 시간 선형 레이어를 사용하여 과거 시계열을 직접 회귀하여 미래 예측을 수행
- 다른 변수들 간에 가중치를 공유하고 공간적 상관관계를 모델링하지 않음

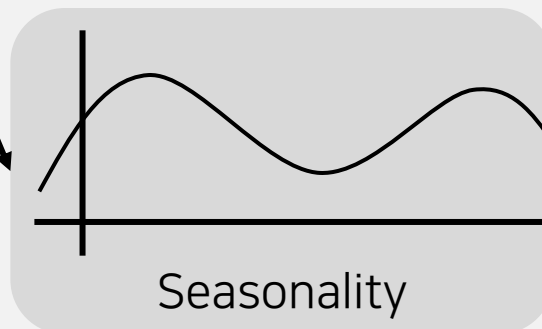
Modeling Strategy



- 주기적으로 발생하는 공통 패턴
- 데이터가 공유하는 장기 패턴

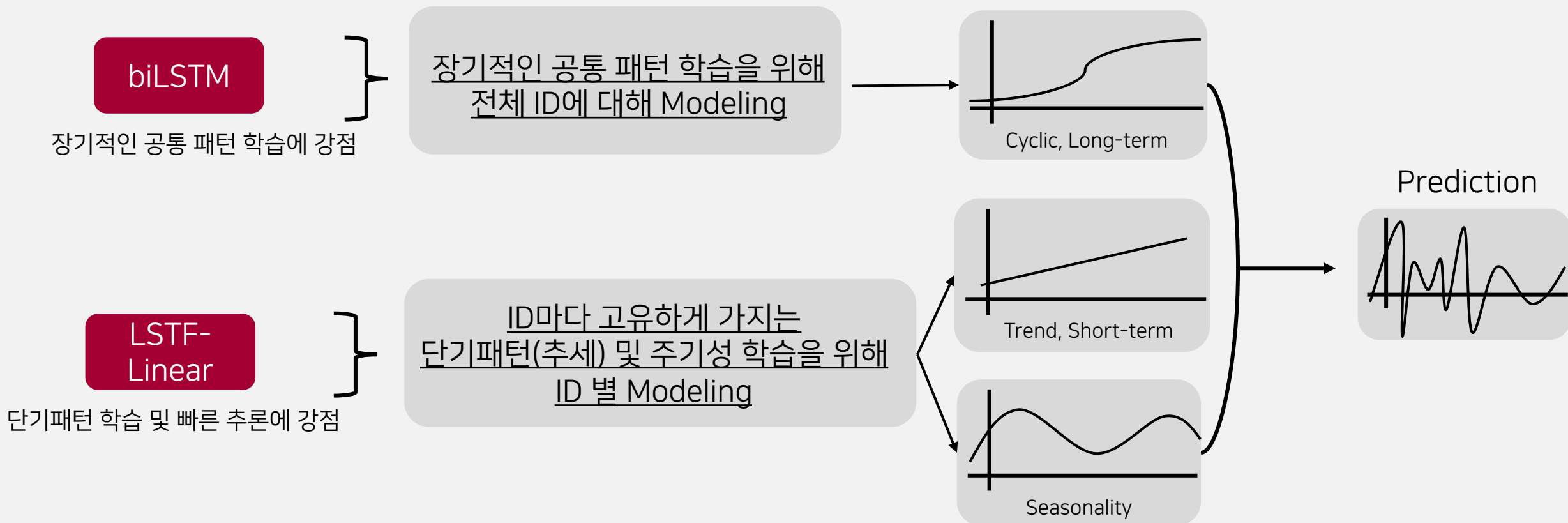


- 단기적 이벤트에 의한 추세
- ID마다 고유하게 가지는 단기 패턴



- 연간, 월간, 주간 등 반복 패턴
- ID마다 고유하게 가지는 계절성

Modeling Strategy

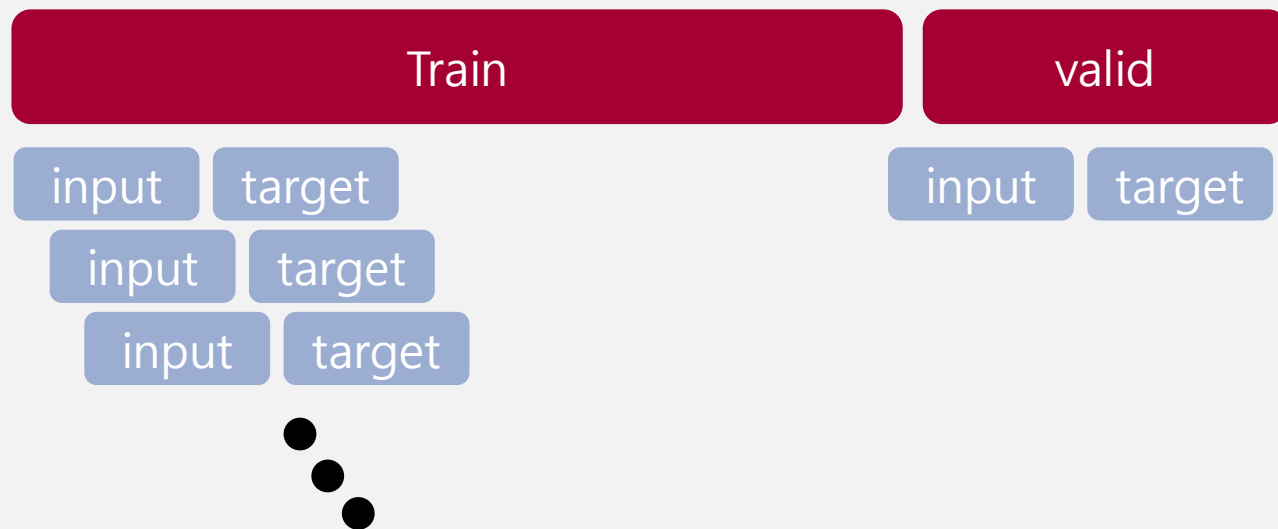


각 모델의 장점을 데이터의 다양한 특성에 맞게 사용하여 예측력 극대화

Contents

-
- 01 Intro
 - 02 Analysis
 - 03 Model Algorithm
 - 04 Validation**
 - 05 Application
-

Validation Strategy



- 마지막 21일에 해당하는 data에 대한 구간을 validation set으로 지정
- PSFA를 validation score로 사용
- Imputation이 실제 data에는 없음을 고려, validation set에 imputation을 적용하지 않음
- 모델 imputation 방법 평가 및 모델 성능 평가에 활용

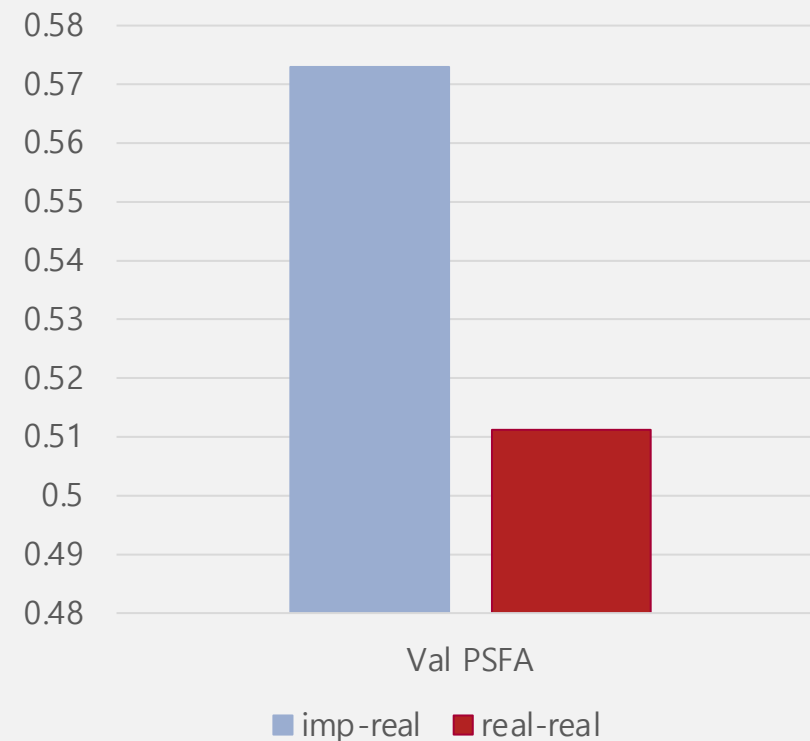
Model Evaluation

- (Train : imputation, Valid : real data)

Epoch : [6] Train Loss : [0.01634] Val PSFA : [0.57293]

- (Train : real data, Valid : real data)

Epoch : [6] Train Loss : [0.01563] Val PSFA : [0.51123]



- (Train : real data, Valid : real data) vs (Train : imputed, Valid : real data) 성능 비교 (imputation 평가)
- Baseline에 대해 동일환경에서 실험 결과 imputation의 성능이 PSFA score에 대해 강점을 보임

Contents

-
- 01 Intro
 - 02 Analysis
 - 03 Model Algorithm
 - 04 Validation
 - 05 Application**
-

Computing Power

GPU - NVIDIA Tesla A100

CPU - Intel(R) Xeon(R) CPU @ 2.20GHz

시간당 학습 cost
\$16.3
(AWS 기준)

학습시간 - Bi-LSTM

GPU 학습 시 약 1시간 40분 소요

“장기적인
공통 패턴
학습에 강점”

학습시간 - LSTFLinear

276개 학습 시 1분

모델 학습 시 샘플당 0.2초 소요

“단기패턴 학습 및
빠른 추론에
강점”

현업 적용 가능성

- 쇼핑몰간 전체적인 흐름과 쇼핑몰별 개별적인 흐름을 결과에 적용함으로써 예측에 대한 정확성을 높일 수 있음.
- 다양한 모델 사용으로 인해서 예측이 불가능한 패턴이 일어나는 경우 Robust함
- 학습 및 추론 속도가 빨라 다음날의 예측을 미리 파악하여 재고 확보 및 발주량을 정확하게 할 수 있음

감사합니다

Outro