DLINEAR 논문 리뷰

Abstract

- transformer 기반의 모델은 non-transformer 모델에 비해 성능이 좋은 것처럼 보였지만, aggressive한 방법으로 실험이 진행되어 error accumulation effect로 인해 long-term 예측에 성능이 안 좋을 수 밖에 없었음
- `DLinear`
  - 입력을 trend와 reminder로 분해
  - 2개의 1-layer network로 구성된 모델
  - transformer 기반의 모델들보다 성능이 좋음

Transformer-Based LSTF Solutions

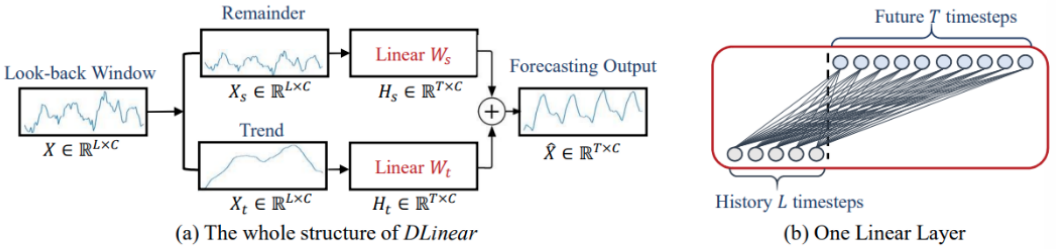
- Vanilla Transformer을 LTSF 문제에 적용할 때 `높은 계산, 메모리 복잡도` , `Error accumulation effect` 발생
- transformer 기반 모델에서의 해결 방법론
  - **Time Series Decomposition**
    - `Moving Average Kernel` 을 적용해 trend와 seasonal 요소로 분해 → Autoformer, FEDformer
  - **Input Embedding Strategies**
    - transformer의 self-attention layer이 시계열의 positional information을 보존하지 못함
    - 이로 인해 시계열 데이터의 temporal context를 유지하기 위한 임베딩이 매우 중요함
    - `hierarichical timestamps` 나 `agnostic timestamps` 를 담고 있는 temporal information을 담기 위해 다양한 임베딩 방법 도입

    - **Self-Attention Schemes**
        - paired element 사이에 존재하는 semantic dependency를 추출하기 위해 사용
        - 이때 연산이 n^2로 매우 큼
        - Informer에서 sparse attention을 사용하거나 pyraformer에서 pyramidal attention을 사용하는 방법 제시
    - **Decoder**
        - vanilla 모델의 경우 decoder에서 autoregressive한 상태로 시퀀스가 나오기 때문에 LSTF에 비해 연산 속도도 느리고, error accumulation effect 발생
        - decoder 대신 FC layer 이용 or 개별적인 decomposition scheme을 사용
- Transformer의 전제가 `semantic correlation between paired element` 이지만, 시계열 데이터에서는 포인트 간 semantic correlation이 거의 존재하지 않음

  → 오히려 element 사이의 순서가 더 중요한 것으로 판단됨
- 따라서, transformer에서 positional encoding을 하거나 토큰을 사용해 임베딩하는 식으로 순서 정보를 보존한다고 하더라도, permutation-invariant self attention mechanism 자체가 본질적으로 temporal information loss를 초래함
- LTSF의 해결책은 Transformer을 사용하지 않는 것이다

DLinear

- `1-layer linear network`는 미래를 예측하기 위해 과거의 정보를 압축할 수 있는 가장 간단한 모델
- `시계열 분해`가 TSF 문제에서 transformer 기반 모델들의 성능을 높일 수 있는 방법임을 확인할 수 있음 → model-agonstic하고 다른 모델에도 쉽게 적용할 수 있다는 점에서 선형 모델에 적용 가능
- DLinear 구성 요소
    - 시계열 분해
        - trend component와 remainder component로 분해
    - 선형 네트워크
        - 2개의 1-layer linear network
    - 이미지



(a) The whole structure of *DLinear* (b) One Linear Layer

Figure 2: Illustration of the Decomposition Linear Model.

- 수식

$$(1)\ \hat{X} = H_s + H_t$$

$$(2)\ \text{Remainder}\ H_s = W_s X_s$$

$$(3)\ \text{Trend}\ H_t = W_t X_t$$

- 장점
  - `O(1) Traversing Path Length`

    short range와 long range의 temporal relation을 낮은 연산량으로 확인할 수 있음
  - `High Efficiency`

    transformer에 비해 사용 메모리와 파라미터가 적기 때문에, 연산 속도가 더 빠름
  - `Interpretability`

    seasonability와 trend에 대한 weight가 분해된 input에 곱해진 선형 모델

    → weight에 대한 분석으로 직관적인 해석 가능
  - `Easy to use`

    transformer와 달리 하이퍼파라미터 튜닝 없이 사용 가능