# TaBERT : Pretraining for Joint Understanding of Textual and Tabular Data

https://arxiv.org/abs/2005.08314

## 0. Introduction

- 긴 시계열(long-term time series)을 정확히 예측하는 것은 어려운 문제였음
- Transformer 기반 모델들이 등장했지만, 긴 horizon에서 성능 급락 문제가 지속됨
- 주요 원인은 다음 두 가지였음
  - self-attention이 시계열의 주기성*을 제대로 반영하지 못함
  - autoregressive 구조로 인해 horizon이 길어질수록 오차 누적이 심해짐
- Auto-Correlation 메커니즘과 Series Decomposition 구조를 도입함
- 기존 Transformer 기반 LTSF(Long-Term Series Forecasting) 모델보다 장기 구간에서 큰 폭의 성능 향상을 달성함

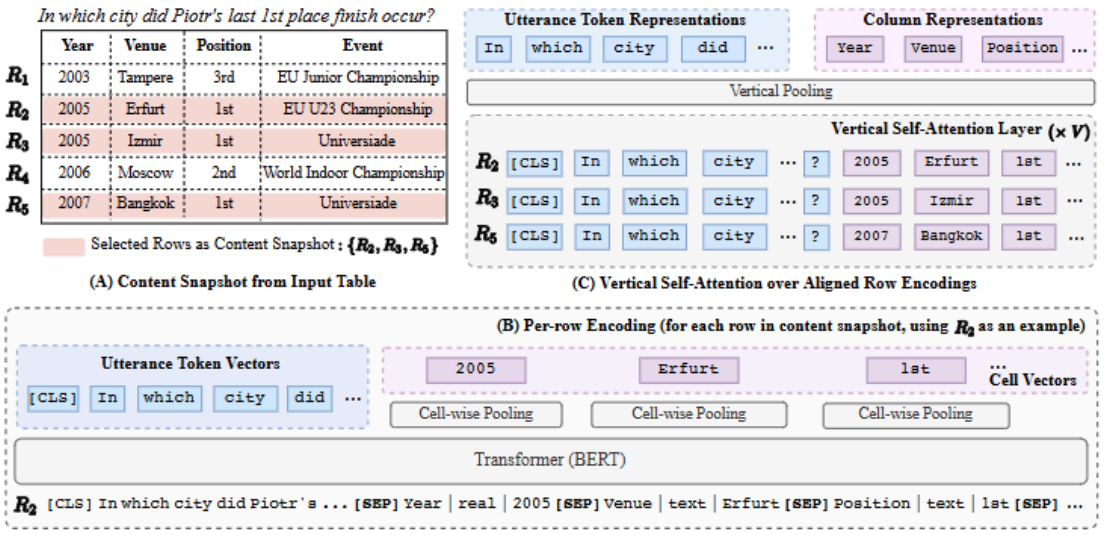
## 1. Overview

- Autoformer는 Encoder-Decoder 구조를 유지하면서 내부 블록을 Decomposition + Auto-Correlation으로 재설계함
- Self-attention 대신 Auto-Correlation을 사용해 반복적 패턴 주기를 직접 찾아냄
- 각 블록마다 시계열을 trend와 seasonal로 분해해 noise를 줄이고 학습 안정성을 확보함
- Decoder는 trend는 linear interpolation, seasonal은 Auto-Correlation 기반 보간으로 예측

## 2. Challenges

- Transformer의 self-attention은 긴 시계열에서 다음 문제를 가짐
  - attention이 패턴 간 유사성을 직접적으로 계산하지 못함
  - 길이가 늘어날수록 복잡도 증가
  - autoregressive 방식의 누적 오류
- 실무에서 사용되는 시계열은 계절성, 갑작스러운 변화, 장기 패턴 등이 혼재해 모델링이 어려움
- 기존 LTSF 연구들은 주기적 패턴을 직접적으로 다루지 못해 장기 성능이 떨어지는 경향 있었음
- 따라서 긴 horizon에서 안정적 성능을 내기 위한 구조가 필요했음

# 3. Method




- 모델은 **Series Decomposition**과 **Auto-Correlation Mechanism** 두 축으로 설계됨
- Encoder와 Decoder 모두 decomposition을 반복적으로 적용하는 구조로 구성됨
- Self-attention을 완전히 Auto-Correlation으로 대체함

## Series Decomposition

- 입력 시계열을 **trend + seasonal**로 분해함
- moving average 기반 방식 사용함
- decomposition은 각 블록에서 residual connection 대체 방식으로 반복 적용됨
- 분해를 통해 noise가 줄고 장기 예측 안정성이 높아짐
- encoder와 decoder 모두 동일한 decomposition 구조 사용함

## Auto-Correlation Mechanism

- self-attention을 대체하는 핵심 모듈임
- 시계열을 shift 시킨 여러 복사본과의 **Auto-Correlation(자기상관)**을 계산함
- correlation score가 높은 shift들 중 **top-k delay**를 선택함
- 선택된 delay의 seasonal 패턴을 재조합하여 새로운 representation 생성함
- 반복 패턴이 강한 시계열의 seasonality를 명시적으로 학습 가능함
- self-attention 대비 복잡도가 더 낮고 장기 패턴 포착 능력이 뛰어남

## Encoder 구조

- 입력을 decomposition으로 trend와 seasonal로 분리함
- seasonal만 Auto-Correlation을 통과함
- trend는 별도로 분리되어 다음 블록으로 전달됨
- 블록마다 decomposition이 반복되어 안정적 feature 추출이 가능해짐

## Decoder 구조

- 디코더 입력 역시 decomposition으로 trend/seasonal 분리됨
- seasonal 입력은 Auto-Correlation으로 과거 주기 패턴을 불러와 예측에 활용함
- trend는 linear extrapolation 방식으로 예측됨
- autoregressive 방식 없이 전체 horizon을 한 번에 출력하도록 설계됨

# 4. Experiments

- 사용 데이터셋

    - ETTh1, ETTh2
    - ETTm1, ETTm2
    - Electricity
    - Exchange-Rate
    - Traffic
- 예측 길이(horizon)는 96, 192, 336, 720
- 비교 모델
    - Transformer 계열: Informer, LogTrans, Reformer
    - 딥러닝 계열: LSTM, N-BEATS
    - 기존 시계열 기반 모델 등
- MSE, MAE로 평가
- 모델은 모든 horizon에서 안정적으로 성능 측정됨

# 5. Results

*Previous Systems on WikiTableQuestions*

| Model | DEV | TEST |
|---|---|---|
| Pasupat and Liang (2015) | 37.0 | 37.1 |
| Neelakantan et al. (2016) | 34.1 | 34.2 |
| Ensemble 15 Models | 37.5 | 37.7 |
| Zhang et al. (2017) | 40.6 | 43.7 |
| Dasigi et al. (2019) | 43.1 | 44.3 |
| Agarwal et al. (2019) | 43.2 | 44.1 |
| Ensemble 10 Models | – | 46.9 |
| Wang et al. (2019b) | 43.7 | 44.5 |

*Our System based on MAPO (Liang et al., 2018)*

| | DEV | Best | TEST | Best |
|---|---|---|---|---|
| Base Parser† | 42.3 ±0.3 | 42.7 | 43.1 ±0.5 | 43.8 |
| *w/* $\text{BERT}_{\text{Base}}$ (K = 1) | 49.6 ±0.5 | 50.4 | 49.4 ±0.5 | 49.2 |
| – content snapshot | 49.1 ±0.6 | 50.0 | 48.8 ±0.9 | 50.2 |
| *w/* $\text{TaBERT}_{\text{Base}}$ (K = 1) | 51.2 ±0.5 | 51.6 | 50.4 ±0.5 | 51.2 |
| – content snapshot | 49.9 ±0.4 | 50.3 | 49.4 ±0.4 | 50.0 |
| *w/* $\text{TaBERT}_{\text{Base}}$ (K = 3) | 51.6 ±0.5 | 52.4 | 51.4 ±0.3 | 51.3 |
| *w/* $\text{BERT}_{\text{Large}}$ (K = 1) | 50.3 ±0.4 | 50.8 | 49.6 ±0.5 | 50.1 |
| *w/* $\text{TaBERT}_{\text{Large}}$ (K = 1) | 51.6 ±1.1 | 52.7 | 51.2 ±0.9 | 51.5 |
| *w/* $\text{TaBERT}_{\text{Large}}$ (K = 3) | **52.2** ±0.7 | **53.0** | **51.8** ±0.6 | **52.3** |

*Top-ranked Systems on Spider Leaderboard*

| Model | DEV. ACC. |
|---|---|
| Global–GNN (Bogin et al., 2019a) | 52.7 |
| EditSQL + BERT (Zhang et al., 2019a) | 57.6 |
| RatSQL (Wang et al., 2019a) | 60.9 |
| IRNet + BERT (Guo et al., 2019) | 60.3 |
| + Memory + Coarse-to-Fine | 61.9 |
| IRNet V2 + BERT | 63.9 |
| RyanSQL + BERT (Choi et al., 2020) | **66.6** |

*Our System based on TranX (Yin and Neubig, 2018)*

| | Mean | Best |
|---|---|---|
| *w/* $\text{BERT}_{\text{Base}}$ (K = 1) | 61.8 ±0.8 | 62.4 |
| – content snapshot | 59.6 ±0.7 | 60.3 |
| *w/* $\text{TaBERT}_{\text{Base}}$ (K = 1) | 63.3 ±0.6 | 64.2 |
| – content snapshot | 60.4 ±1.3 | 61.8 |
| *w/* $\text{TaBERT}_{\text{Base}}$ (K = 3) | 63.3 ±0.7 | 64.1 |
| *w/* $\text{BERT}_{\text{Large}}$ (K = 1) | 61.3 ±1.2 | 62.9 |
| *w/* $\text{TaBERT}_{\text{Large}}$ (K = 1) | 64.0 ±0.4 | 64.4 |
| *w/* $\text{TaBERT}_{\text{Large}}$ (K = 3) | **64.5** ±0.6 | **65.2** |

*u*: *How many years before was the film Bacchae out before the Watermelon?*

Input to TABERT$_{Large}$ (K = 3) ▷ *Content Snapshot with Three Rows*

| Film | Year | Function | Notes |
|---|---|---|---|
| The Bacchae | 2002 | Producer | Screen adaptation of... |
| The Trojan Women | 2004 | Producer/Actress | Documutary film... |
| The Watermelon | 2008 | Producer | Oddball romantic comedy... |

Input to TABERT$_{Large}$ (K = 1) ▷ *Content Snapshot with One Synthetic Row*

| Film | Year | Function | Notes |
|---|---|---|---|
| The Watermelon | 2013 | Producer | Screen adaptation of... |

| Cell Linearization Template | WIKIQ. | SPIDER |
|---|---|---|
| Pretrained TABERT$_{Base}$ Models (K = 1) | | |
| `Column Name` | 49.6 +0.4 | 60.0 +1.1 |
| `Column Name` \| `Type`† (–content snap.) | 49.9 +0.4 | 60.4 +1.3 |
| `Column Name` \| `Type` \| `Cell Value`† | 51.2 +0.5 | 63.3 +0.6 |
| BERT$_{Base}$ Models | | |
| `Column Name` (Hwang et al., 2019) | 49.0 +0.4 | 58.6 +0.3 |
| `Column Name` `is` `Cell Value` (Chen19) | 50.2 +0.4 | 63.1 +0.7 |

- Autoformer는 모든 데이터셋에서 long-horizon 기준 SOTA 기록
- 특히 예측 길이가 336, 720처럼 매우 길어질수록 성능 우위가 두드러짐
- Ablation 결과
  - Auto-Correlation 제거 시 성능 급락
  - Decomposition 제거 시 장기 구간에서 특히 불안정
- 계산량(FLOPs)은 Informer보다 낮으면서 성능은 더 좋음
- long-term forecasting에서 Transformer 기반 모델 중 가장 안정적인 모델로 평가됨

# 6. Insight

- 시계열에서 seasonality는 매우 중요한 구조인데, 기존 Transformer는 이를 명시적으로 다루지 못했음
- Auto-Correlation은 반복 패턴을 직접 modeling하므로 시계열 특성에 더 적합함
- autoregressive 제거가 horizon 증가에 따른 누적 오류를 근본적으로 줄여줌
- decomposition은 trend를 분리하여 seasonal만 모델링하게 함 → 예측 난이도 감소
- 실무적으로 전력 수요, 교통량 같은 반복 패턴이 강한 데이터에 특히 유리
- 비주기적 시계열에서는 개선 폭이 작을 가능성 있음
- moving average 기반 decomposition이 모든 데이터에 최적은 아닐 수 있음