MAMBAFORMER-MOE: MAMBA-BASED MIXTURE-OF-EXPERTS FOR MULTIVARIATE TIME SERIES PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

We propose the MambaFormer-MOE, a mamba-based mixture-of-experts (MOEs) model for multivariate time series prediction. There are three major features of our model. 1. We propose a temporal modeling module based-on the Mamba architecture to model temporal correlations with linear complexity. 2. We propose a cross-variate correlation modeling mechanism based-on self-attention to equip Mamba with multivariate time series prediction capability. 3. We propose a MOE mechanism that has experts that specialize in mixing the variates in different ways. It makes the model generalizable to multivariate time series from different domains. Our empirical results demonstrate that our model has SOTA prediction performance on various multivariate time series datasets.

1 INTRODUCTION

Transformer has become successful in various domains including natural language processing, computer vision, etc. Its attention mechanism shows tremendous generalization ability and scalability. In the time series prediction domain, Transformer-based models are gaining tremendous popularity and achieving state-of-the-art prediction performance.

However, the quadratic computation complexity of the attention mechanism in the Transformer architecture has long been a major issue hindering its scalability in time series prediction tasks. Regarding such issue, many researchers try developing more computationally friendly architectures for sequence modeling. A recent achievement is the Mamba architecture that has a linear computation complexity while still has a sequence modeling performance on par with the self-attention mechanism. It is a good alternative method to model temporal correlations in time series sequence. However, the only downside of Mamba is that it lacks proper mechanism to capture cross-variate correlations in multivariate time series.

On the other hand, recent work discovers that in time series prediction tasks, the self-attention mechanism, the core of the Transformer architecture, is more suitable of modeling cross-variate correlations across time series of variates rather than modeling temporal correlations along the time series of each variate.

Based on the aforementioned observations, we propose a new multivariate time series prediction neural architecture, MambaFormer-MOE. In this model, we intend to merge the advantages from both the Mamba architecture and the self-attention mechanism. Regarding temporal correlation modeling, we propose a Mamba-based module to utilize its sequence modeling capability and its linear complexity so that the model complexity can linearly scale with the length of the multivariate time series. To model the cross-variate correlations, we utilize the self-attention mechanism that calculates the self-attention between different varieties of the time series. By doing this, we can leverage its ability of capturing cross-varaite correlations while controlling its quadratic complexity as a big-O of constant with respect to the number of variates of the time series. Our major contributions are as follows.

- We propose a state-of-the-art multivariate time series prediction model with linear computation complexity.
- We demonstrate the temporal modeling capability of the Mamba architecture

- We propose an MOE mechanism to increase the scalability and the generalizability of a time series prediction model.

We organize the rest of the paper in the following way. Section 2 is the related work. Section 3 is the introduction of our methodology. Section 4 is the experiment section. Section 5 is the discussion and conlusion.

2 Related Work

There are multiple lines of deep learning research regarding multivariate time series prediction despite the difference of model architectures. The first line is about being able to model a time series input with long history. The second line is to model time series as a combination of trend and seasonality. The third line is to model the multi-resolution information of a time series. The fourth line is to model time series in a generic way such that one model can apply on time series from various domains without or with little adaptations. Last but not least, the fifth line tries to model time series together with heterogenous covariates including static features and time-relevant features.

Long History Modeling. The Transformer architecture is a successor to solve the inefficiency and ineffectiveness of recurrent neural networks modeling long sequence data (Vaswani et al., 2017). Transformers have gained prominence in time series prediction (Beltagy et al., 2020; Salinas et al., 2020; Lai et al., 2018), inspired by their success in Natural Language Processing and Computer Vision. One challenge in using transformers in time series prediction is managing their' quadratic complexity due to the typically long sequences. To address this, many researchers have not only optimized for prediction performance but also sought to reduce memory and computational complexity. LogTrans (Li et al., 2019) proposes a transformer-based neural network for time series prediction. They propose a convolution layer over the vanilla transformer to better capture local context information and a sparse attention mechanism to reduce memory complexity. In their version of sparse attention, the query only attends to a subset time steps spaced apart following a fixed log-based stride pattern. Similarly, Informer (Zhou et al., 2021) proposes convolutional layers in between attention blocks to distill the dominating attention and a sparse attention mechanism where the keys only attend to a subset of most dominant queries according to some dominance measurement. Reformer (Kitaev et al., 2020) replaces the dot-product self-attention in the vanilla transformer with a hashing-based attention mechanism to reduce the complexity.

Decomposition-based Modeling. Besides directly feeding the raw time series inputs to the model, a different line of research is to treat a time series as a combination a trend series and a seasonality series and model them in a decomposition way. Among transformer-based models, Autoformer (Wu et al., 2021) introduces a series decomposition module to its transformer-based model to separately model the seasonal component and the trend-cyclical of the time series. FEDformer (Zhou et al., 2022) also models the decomposed time series and they introduce a block to extract signals by transforming the time series to the frequency domain.

Multi-resolution-based Modeling Some works intend to model the multi-resolution or multi-scale signals in the time series with a dedicated network design. Pyraformer (Liu et al., 2021) designs a pyramidal attention module to extract the multi-scale signals from the raw time series. Crossformer (Zhang & Yan, 2022) proposes a multi-scale encoder-decoder architecture to hierarchically extract signals of different resolutions from the time series. Besides, STanHop (Wu et al., 2023a) is a multi-resolution model based on modern hopfield. Its novelty is that it also includes an associative memory module enabling quick reactions to sudden anomaly in the time series.

Foundation-model-based Modeling. PatchTST (Nie et al., 2023) is a transformer-based model following a channel-independence assumption. According to this assumption, a univariate time series model is capable of modeling a multivariate time series by independently feeding each channel of the multivariate time series as a univariate series into the model during training and testing. With channel-independence, the same model architecture is immediately transferable to multivariate time series of different domains. Besides channel-independence, PatchTST also proposes a patch encoding mechanism by tokenizing the time series by segments of consecutive time steps. It is based on the assumption that local patterns spanning a small segment of consecutive time steps are more suitable to be the fundamental meaningful tokens compared to the fine-grained and noisy time steps themselves. PatchTST has been a state-of-the-art multivariate time series prediction model and is still having top-tier performance in comparison to some latest models. Recently, transformer-based large pretrained models show their potential of being the perfect paradigm for constructing a foundation model. They demonstrate their capability in natural language processing (Devlin et al., 2018; Liu et al., 2019; Lagler et al., 2013; Floridi & Chiriatti, 2020; Achiam et al., 2023; OpenAI, 2023; Touvron et al., 2023a;b; Anil et al., 2023; Chowdhery et al., 2023; Zhang et al., 2022a), computer vision (Dosovitskiy et al., 2020; Radford et al., 2021; Yuan et al., 2021; Zhang et al., 2021; Zhai et al., 2022; Yu et al., 2022; Li et al., 2022; 2023; Liu et al., 2024), and genomics modeling (Ji et al., 2021; Zhou et al., 2023b; Dalla-Torre et al., 2023). They inspire time series researchers to adapt the same paradigm to time series prediction. This adaptation is natural by adopting the channel-independence assumption. LLMTIME (Gruver et al., 2024) is a zero-shot method to equip a large language model (LLM), such as Llama and GPT-3, with a special tokenization method on time series. By doing that, the LLM can directly generate the predicted future values of the time series. Their results are comparable to domain-specific transformer-based model such as FEDFormer and Autoformer. Rather than assuming the input time series is in the natural language domain like LLMTIME does, Time-LLM (Jin et al., 2023) tries to utilize a pretrained LLM while add additional components to bridge the time series domain and the natural language domain. They added an alignment module that uses cross-attentions to align the patch-encoded input time series and the tokenized input natural language prompts. Then, a concatenated embedding including the information from both the prompt and the time series becomes the input to the frozen pretrained Llama model. Rather than keeping the LLM frozen, GPT4TS (Zhou et al., 2023a) proposes to fine-tune some components of a pretrained LLM on the time series data. They feed as an input the patch-encoded time series to a GPT-2 model and only fine-tune the residual connections (He et al., 2016) and the LayerNorm layers (Ba et al., 2016) in the transformer block (Vaswani et al., 2017). LLM4TS (Chang et al., 2023) follows the same path. The difference is that in addition to the residual connections and LayerNorm layers, they also fine-tune the self-attention module. More specifically, they use LoRA (Hu et al., 2021) to fine-tune the three weight matrices of the self-attention module. Both LLMTIME, Time-LLM, GPT4TS and LLM4TS adopt the channel-independence assumption. Both Time-LLM, GPT4TS and LLM4TS outperform PatchTST. Instead of using a pretrained LLM, some researchers also try taking an untrained LLM architecture and pretrain it exclusively on time series data. Lag-Llama (Rasul et al., 2023) is a univariate time series foundation model that pretrains a Llama architecture on a time series corpus with time series from various domains and of different lengths. Lag-Llama does not make point-wise predictions. It predicts the parameters of a t-distribution (Student, 1908) for each time step in the prediction horizon.

Time Series Modeling with Heterogenous Covariates. Most time series prediction models assume the input includes only time series. Some researchers focus on a time series model that can simultaneously model various kinds of covariates that include additional information helping the time series prediction. Temporal Fusion Transformer (Lim et al., 2021) proposes to first use a residual block based on fully connected layers to encode static covariates, such as the location of the store, into a context vector. Then, an LSTM module takes the context vector as a hidden states to help the time series prediction. TiDE (Das et al., 2023) adopts a similar strategy. It also uses residual blocks of fully connected layers to encode covariates. Its novelty is that it concatenates both the time series, static covariates and time-relevant covariates into a flat vector and uses an architecture based on the same type of residual blocks to do the final prediction.

Time Series Model Architectures. Transformer-based architectures are dominant in multivariate time series prediction. However, some recent models based on linear layers (Zeng et al., 2023; Chen et al.; Wang et al., 2024; Oreshkin et al., 2020; Challu et al., 2023; Zhang et al., 2022b) are showing performance on par with transformer-based models. There are also time series models based on convolutional neural networks (O'shea & Nash, 2015; Wu et al., 2023b; Franceschi et al., 2019) and recurrent networks (Hochreiter & Schmidhuber, 1997; Lai et al., 2018; Franceschi et al., 2019). But, their performance is not on par with transformers and linear-based architectures.

3 Methodology

Problem Definition. We focus on the multivariate time series prediction problem. Given a multivariate time series input $X = \{X_1, X_2, ..., X_T\} \in \mathbb{N}^{T\times N}$ of the history, we want to predict the future time steps of the same multivaraite time series, $Y = \{X_{T+1}, X_{T+2}, ..., X_{T+H}\} \in \mathbb{N}^{H\times N}$. T is the look back window, the number of time steps of the input, H is the horizon, the number of time steps of the predicted output, and N is the number variates in the multivariate time series.

3.1 Structure Overview

State Space Models. Satet space models represent any recurrent modeling process that includes latent variables as states. We use this term specifically for the structured state space model (S4). The S4 model is to model a continuous system by mapping an input sequence, $\{x_1, x_2, ..., x_t\}, x_t \in \mathbb{R}$, to an output sequence, $\{y_1, y_2, ..., y_t\}, y_t \in \mathbb{R}$. S4 includes four learnable parameters, Δ, A, B, C and maps the input sequence (function) to the output sequence (function) through a hidden state, $h(t)$:

$$h'(t) = Ah(t) + Bx(t),\ A \in \mathbb{R}^{N\times N},\ B \in \mathbb{R}^{N\times D} \tag{1}$$

$$y(t) = Ch(t),\ C \in \mathbb{R}^{N\times D} \tag{2}$$

The above continuous system is discretized using a step size, Δ:

$$h_t = \bar{A}h_{t-1} + \bar{B}x_t, \bar{A} = exp(\Delta A), \bar{B} = (\Delta A)^{-1}(exp(\Delta A) - I) \cdot \Delta B \tag{3}$$

$$y_t = Ch_t \tag{4}$$

After discretization, the calculation of the model is efficient using a linear recursive approach Gu et al. (2021). In addition, S4 is so named because the initialization process uses HiPPO to impose structure on the state matrix A to improve the modeling of long-range dependency.

Mamba. Mamba proposes a data-dependent selection mechanism on top of S4 to better compress the information in the sequence into the hidden state and a hardware-aware computation algorithm to improve the computation efficiency.

3.2 Model Architecture

4 Main Results

5 Ablation Study

6 Conclusion and Future Work

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*, 2023.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

Iz Beltagy, Matthew E Peters, and Arman Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020.

Cristian Challu, Kin G Olivares, Boris N Oreshkin, Federico Garza Ramirez, Max Mergenthaler Canseco, and Artur Dubrawski. Nhits: neural hierarchical interpolation for time series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 6989–6997, 2023.

Ching Chang, Wen-Chih Peng, and Tien-Fu Chen. Llm4ts: Two-stage fine-tuning for time-series forecasting with pre-trained llms. *arXiv preprint arXiv:2308.08469*, 2023.

SA Chen, CL Li, N Yoder, SO Arik, and T Pfister. Tsmixer: An all-mlp architecture for time series forecasting. arxiv 2023. *arXiv preprint arXiv:2303.06053*.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240): 1–113, 2023.

Hugo Dalla-Torre, Liam Gonzalez, Javier Mendoza-Revilla, Nicolas Lopez Carranza, Adam Henryk Grzywaczewski, Francesco Oteri, Christian Dallago, Evan Trop, Bernardo P de Almeida, Hassan Sirelkhatim, et al. The nucleotide transformer: Building and evaluating robust foundation models for human genomics. *bioRxiv*, pp. 2023–01, 2023.

Abhimanyu Das, Weihao Kong, Andrew Leach, Shaan Mathur, Rajat Sen, and Rose Yu. Long-term forecasting with tide: Time-series dense encoder. *arXiv preprint arXiv:2304.08424*, 2023.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2018.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Luciano Floridi and Massimo Chiriatti. Gpt-3: Its nature, scope, limits, and consequences. *Minds and Machines*, 30:681–694, 2020.

Jean-Yves Franceschi, Aymeric Dieuleveut, and Martin Jaggi. Unsupervised scalable representation learning for multivariate time series. *Advances in neural information processing systems*, 32, 2019.

Nate Gruver, Marc Finzi, Shikai Qiu, and Andrew G Wilson. Large language models are zero-shot time series forecasters. *Advances in Neural Information Processing Systems*, 36, 2024.

Albert Gu, Isys Johnson, Karan Goel, Khaled Saab, Tri Dao, Atri Rudra, and Christopher Ré. Combining recurrent, convolutional, and continuous-time models with linear state space layers. *Advances in neural information processing systems*, 34:572–585, 2021.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Yanrong Ji, Zhihan Zhou, Han Liu, and Ramana V Davuluri. Dnabert: pre-trained bidirectional encoder representations from transformers model for dna-language in genome. *Bioinformatics*, 37(15):2112–2120, 2021.

Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, et al. Time-llm: Time series forecasting by reprogramming large language models. *arXiv preprint arXiv:2310.01728*, 2023.

Nikita Kitaev, Łukasz Kaiser, and Anselm Levskaya. Reformer: The efficient transformer. In *International Conference on Learning Representations*, 2020.

Klemens Lagler, Michael Schindelegger, Johannes Böhm, Hana Krásná, and Tobias Nilsson. Gpt2: Empirical slant delay model for radio space geodetic techniques. *Geophysical research letters*, 40(6):1069–1073, 2013.

Guokun Lai, Wei-Cheng Chang, Yiming Yang, and Hanxiao Liu. Modeling long-and short-term temporal patterns with deep neural networks. In *The 41st international ACM SIGIR conference on research & development in information retrieval*, pp. 95–104, 2018.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pp. 12888–12900. PMLR, 2022.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pp. 19730–19742. PMLR, 2023.

Shiyang Li, Xiaoyong Jin, Yao Xuan, Xiyou Zhou, Wenhu Chen, Yu-Xiang Wang, and Xifeng Yan. Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting. *Advances in Neural Information Processing Systems*, 32, 2019.

Bryan Lim, Sercan Ö Arık, Nicolas Loeff, and Tomas Pfister. Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting*, 37(4): 1748–1764, 2021.

Shizhan Liu, Hang Yu, Cong Liao, Jianguo Li, Weiyao Lin, Alex X Liu, and Schahram Dustdar. Pyraformer: Low-complexity pyramidal attention for long-range time series modeling and forecasting. In *International conference on learning representations*, 2021.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Yixin Liu, Kai Zhang, Yuan Li, Zhiling Yan, Chujie Gao, Ruoxi Chen, Zhengqing Yuan, Yue Huang, Hanchi Sun, Jianfeng Gao, et al. Sora: A review on background, technology, limitations, and opportunities of large vision models. *arXiv preprint arXiv:2402.17177*, 2024.

Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *International Conference on Learning Representations*, 2023.

OpenAI. Gpt-4 technical report, 2023.

Boris N Oreshkin, Dmitri Carpov, Nicolas Chapados, and Yoshua Bengio. N-beats: Neural basis expansion analysis for interpretable time series forecasting. In *International Conference on Learning Representations*, 2020.

Keiron O'shea and Ryan Nash. An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458*, 2015.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PMLR, 2021.

Kashif Rasul, Arjun Ashok, Andrew Robert Williams, Arian Khorasani, George Adamopoulos, Rishika Bhagwatkar, Marin Biloš, Hena Ghonia, Nadhir Vincent Hassen, Anderson Schneider, et al. Lag-llama: Towards foundation models for time series forecasting. *arXiv preprint arXiv:2310.08278*, 2023.

David Salinas, Valentin Flunkert, Jan Gasthaus, and Tim Januschowski. Deepar: Probabilistic forecasting with autoregressive recurrent networks. *International journal of forecasting*, 36(3):1181–1191, 2020.

Student. The probable error of a mean. *Biometrika*, pp. 1–25, 1908.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023a.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023b.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 2017.

Shiyu Wang, Haixu Wu, Xiaoming Shi, Tengge Hu, Huakun Luo, Lintao Ma, James Y Zhang, and Jun Zhou. Timemixer: Decomposable multiscale mixing for time series forecasting. *arXiv preprint arXiv:2405.14616*, 2024.

Dennis Wu, Jerry Yao-Chieh Hu, Weijian Li, Bo-Yu Chen, and Han Liu. Stanhop: Sparse tandem hopfield model for memory-enhanced time series prediction. *arXiv preprint arXiv:2312.17346*, 2023a.

Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Advances in Neural Information Processing Systems*, 34:22419–22430, 2021.

Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. Timesnet: Temporal 2d-variation modeling for general time series analysis. In *International Conference on Learning Representations*, 2023b.

Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022.

Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, et al. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*, 2021.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pp. 11121–11128, 2023.

Xiaohua Zhai, Xiao Wang, Basil Mustafa, Andreas Steiner, Daniel Keysers, Alexander Kolesnikov, and Lucas Beyer. Lit: Zero-shot transfer with locked-image text tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18123–18133, 2022.

Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. Vinvl: Making visual representations matter in vision-language models. *CoRR*, abs/2101.00529, 2021. URL `https://arxiv.org/abs/2101.00529`.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022a.

Tianping Zhang, Yizhuo Zhang, Wei Cao, Jiang Bian, Xiaohan Yi, Shun Zheng, and Jian Li. Less is more: Fast multivariate time series forecasting with light sampling-oriented mlp structures. *arXiv preprint arXiv:2207.01186*, 2022b.

Yunhao Zhang and Junchi Yan. Crossformer: Transformer utilizing cross-dimension dependency for multivariate time series forecasting. In *The eleventh international conference on learning representations*, 2022.

Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 11106–11115, 2021.

Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *International Conference on Machine Learning*, pp. 27268–27286. PMLR, 2022.

Tian Zhou, Peisong Niu, Xue Wang, Liang Sun, and Rong Jin. One fits all: Power general time series analysis by pretrained lm. *Advances in Neural Information Processing Systems*, 36, 2023a.

Zhihan Zhou, Yanrong Ji, Weijian Li, Pratik Dutta, Ramana Davuluri, and Han Liu. Dnabert-2: Efficient foundation model and benchmark for multi-species genome. *arXiv preprint arXiv:2306.15006*, 2023b.