Seduced by the Transformer, Humbled by Simplex: Revisiting Attention for Time Series Forecasting

Finn Alberts[a], Ewoud Vosse[a], Arjan de Weerd[a]

[a] *Open University of the Netherlands, Heerlen, The Netherlands*

Abstract

In many domains, time series forecasting (TSF) is used to predict future events and support decision-making processes. With the rise of deep learning, neural networks have been explored as alternatives to traditional mathematical models such as the Simplex Algorithm, though with mixed results. Recent work has investigated the potential of Transformer architectures to capture complex temporal patterns. However, studies have shown that Transformers often underperform compared to simpler models like the Simplex Algorithm in TSF tasks. To bridge this gap, we introduce a Multi-Head Simplex Attention (MHSA) mechanism designed to mimic the behaviour of the Simplex Algorithm within a Transformer framework. We conducted two baseline experiments using the Simplex Algorithm and a standard Convolutional Transformer, and sixteen additional experiments with MHSA, varying the number of attention heads (1, 2, 4, and 8) and the distance metrics (Euclidean, Cosine Similarity, Manhattan, and Infinity norm). While the best-performing MHSA configuration—Euclidean distance with eight attention heads—showed relatively improved performance within the MHSA group, it still lagged significantly behind the baselines, with MSEs ranging from 80 to 158, compared to just over 1 for both baseline models. These results highlight the challenge of replicating traditional model behaviours within deep learning architectures and point to areas for future improvement.

Keywords: Transformers, Time Series Forecasting, Simplex Projection, Attention Mechanisms, Multi-Head Simplex Attention, Distance-based Similarity, Causal Convolution, Autoregressive Models

1. Introduction

Time series forecasting (TSF) is a crucial task with applications spanning diverse fields. From predicting energy consumption for efficient resource allocation, estimating traffic flow for urban planning, and managing supply chains in retail, accurate time series forecasts enable better decision-making and optimization. While traditional statistical methods like ARIMA (Nelson, 1998) and Exponential Smoothing have been foundational, deep learning has emerged as a powerful alternative. Recurrent Neural Networks (RNNs) have been successfully applied, as demonstrated by DeepAR (Salinas et al., 2020), and architectures like N-BEATS (Oreshkin et al., 2020) have offered interpretable deep learning solutions. Moreover, competitions like the M4 competition (Smyl, 2020; Makridakis et al., 2018) have spurred advancements by highlighting effective hybrid approaches, such as combining Exponential Smoothing with LSTMs.

A significant advancement in sequence modelling came with the Transformer architecture (Vaswani et al., 2017), which revolutionized Natural Language Processing (Devlin et al., 2019; Brown et al., 2020). The core self-attention mechanism which is adept at capturing long-range dependencies, seemed a natural fit for TSF. Numerous variants like Informer (Zhou et al., 2021), Autoformer (Wu et al., 2021), and PatchTST (Wu et al., 2022) have been proposed to tackle TSF challenges using this mechanism.

Despite their sophistication, recent studies have questioned if the superiority of Transformers in other domains also holds for TSF challenges. Notably, Zeng et al. (2023) demonstrated that Transformer models might struggle with fundamental aspects of time series data and can be outperformed by significantly simpler linear models on benchmark datasets. This raises a critical question: Why do these complex attention-based models sometimes fail to outperform simpler forecasting techniques like Simplex projection (Sugihara and May, 1990), which also rely on identifying patterns by finding nearest neighbours in an embedded state space?

This paper investigates the performance discrepancies between Transformer-based models and simpler pattern-matching approaches in TSF. We focus specifically on the self-attention component, hypothesizing that its standard formulation might be less suited for certain time series characteristics compared to more direct distance-based pattern matching like Simplex. To explore this, we propose a novel attention mechanism, Multi-Head Simplex Attention (MHSA), incorporating ideas like causal convolutions (Li et al., 2019). This mechanism is designed to bridge the conceptual gap between Simplex projection and Transformer self-attention. MHSA aims to mimic Simplex's nearest-neighbour logic within the attention framework. We explicitly isolate the attention mechanism, keeping other Transformer components consistent with standard architectures.

Our central hypothesis is that replacing the standard dot-product similarity with a mechanism (MHSA) that finds similar

moments in time (cause) and focusses on what happened next (effect) leads to performance improvements. We conduct experiments comparing the performance of a Transformer decoder architecture equipped with MHSA and various distance metrics against two baselines: the Simplex algorithm (Petchey, 2016) and a standard Transformer decoder using convolutional self-attention as proposed by Li et al. (2019). These experiments are performed on a synthetic dataset composed of stacked sine waves, specifically designed to evaluate pattern recognition and forecasting capabilities. This work contributes an analysis of the Transformer's attention component within the TSF domain, offering insights into its operational characteristics compared to a classic forecasting technique.

The remainder of this paper is structured as follows: Section 2 reviews related work in TSF and Transformer models and compares our new approach to existing research. Section 3 details the baseline methods and our proposed MHSA implementation. Section 4 presents the experimental setup and comparative results using MSE and MAPE metrics. Section 5 discusses the implications of our findings, and Section 6 concludes the paper by highlighting limitations and potential avenues for future research.

2. Related work

Time series forecasting has a rich history, with key developments spanning several decades. One of the earliest notable contributions was by Sugihara and May (1990), who introduced non-linear forecasting to differentiate chaotic dynamics from measurement noise. Their approach leveraged delay embeddings to identify nearest neighbours in the reconstructed state space for predicting future values. Building on this foundation, Nelson (1998) proposed the ARIMA framework, which provided a systematic method for handling non-stationary time series through differencing. This transformation enabled the use of moving averages, effectively accounting for autocorrelated errors. Both methodologies underpin the theoretical basis of our work.

With the rise of deep learning, machine learning approaches have been increasingly adopted to enhance time series forecasting. Notable contributions include DeepAR by Salinas et al. (2020), which leverages recurrent neural networks to demonstrate that deep learning can outperform traditional statistical methods. Similarly, N-BEATS by Oreshkin et al. (2020) introduces a fully connected, interpretable deep learning architecture that achieves strong performance on univariate forecasting tasks without relying on domain-specific knowledge. The M4 competition further advanced the field, with the winning entry by Smyl (2020) combining exponential smoothing and Long Short-Term Memory (LSTM) networks to achieve significant improvements in forecast accuracy. Importantly, the competition results highlighted that hybrid models integrating statistical techniques with machine learning tend to yield the highest accuracy. Collectively, these works establish neural networks as a competitive alternative for time series forecasting, and they motivate the exploration of Transformer-based architectures with Simplex Attention Heads as yet another promising direction.

In 2017, the ground-breaking paper "Attention Is All You Need" by Vaswani et al. (2017) introduced the Transformer model, offering an alternative to recurrent and convolutional networks. Since then, numerous applications of the Transformer architecture have been explored across various domains. A well-known example is the BERT model proposed by Devlin et al. (2019), which has become a cornerstone in Natural Language Processing (NLP) tasks. Beyond NLP, BERT has demonstrated remarkable performance across a wide range of tasks following fine-tuning. However, this success is closely tied to extensive pre-training on large-scale datasets, highlighting the architecture's strong dependency on data availability. These developments provide valuable insights into the design and potential limitations of attention-based models, informing our own approach in adapting Transformer mechanisms to time series forecasting contexts.

Another prominent application of Transformer models—and the central focus of our study—is time series forecasting. While Masini et al. (2023) highlight the potential of non-linear machine learning models in this domain, Transformers continue to present notable challenges. As pointed out by Zeng et al. (2023), these models often overestimate their ability to capture the nuanced temporal dependencies inherent in time series data. Their findings further show that simpler models can, in many cases, outperform Transformers, offering competitive accuracy with reduced computational cost. To address the shortcomings of standard Transformer architectures, several enhancements have been proposed. For example, Li et al. (2019) identify two key limitations: the memory bottleneck and the limited sensitivity of point-wise dot-product attention to local temporal context. To mitigate these issues, they introduce convolutional self-attention, which better captures local dependencies while easing memory demands. Building on this line of inquiry, our study proposes replacing the standard attention mechanism with Simplex Attention Heads, offering a distance-based alternative to dot-product similarity that aims to improve locality awareness and forecasting performance.

Other state-of-the-art solutions that provide the foundation for our study include the Informer (Zhou et al., 2021), Autoformer (Wu et al., 2021), FEDFormer (Zhou et al., 2022), and PatchTST (Wu et al., 2022) models. The Informer was designed to efficiently forecast long sequence time series (LSTF) and introduced a generative mechanism that predicts the full output sequence in one step, rather than token-by-token. Autoformer, another model tailored for LSTF tasks, processes segments of the time series in parallel and concatenates the results. It also replaces the standard attention mechanism with an auto-correlation mechanism, inspiring our consideration of Simplex Attention Heads as an alternative. FEDFormer (Frequency Enhanced Decomposed Transformer) further expands on this idea by combining Transformer architecture with seasonal-trend decomposition, offering a mixture-of-experts approach that aligns with our motivation for incorporating Simplex Attention to a Transformer model. Lastly, PatchTST treats time series in a manner analogous to tokenised text in NLP, reducing architectural complexity while slicing input sequences into patches—an approach that influences our input processing strategy.

Recently, Baljan and Rasoolzadeh Baghmisheh (2024) conducted a comprehensive analysis of Transformer architectures, aiming to identify and implement modifications that would improve their performance in time series forecasting. Their findings suggest that the use of shifted attention mechanisms is overly simplistic, limiting the model's ability to accurately predict subsequent time steps. Nevertheless, the study identifies several promising directions for enhancing Transformers in this domain, particularly the use of modified kernels as alternatives to standard attention mechanisms. Building on this foundation, our research explores the integration of a more effective alternative—Simplex-based attention—into the Transformer architecture. Investigating this approach forms the central focus of our study.

3. Methods

3.1. Simplex Algorithm (Baseline)

As a first baseline model, we use the Simplex algorithm as explained by Petchey (2016). This Simplex algorithm is a simple yet effective method for predicting the next value in a time series. It relies on identifying similar patterns in past observations and using them to estimate future behaviour. In our implementation, each state is using 4 embeddings, identifies the 4 nearest neighbours in the reconstructed state space, and predicts the next value using a weighted average of their subsequent values.

3.2. Transformer Model (Baseline)

As a second baseline model, we use the Transformer model adapted from Li et al. (2019), which introduces convolutional self-attention to better model locality in time series data. Unlike the canonical Transformer, which computes attention based solely on point-wise values, this variant generates queries and keys using causal convolutional layers. This modification allows the model to incorporate local contextual information—such as trends and shapes—into the attention mechanism, thereby improving its ability to detect meaningful temporal patterns and enhancing robustness to anomalies.

3.3. Transformer with Multi-Head Simplex Attention (MHSA)

To bridge the gap between standard Transformer attention mechanisms and Simplex-style projections, we propose Multi-Head Simplex Attention (MHSA) — a novel attention module designed specifically for autoregressive time series modelling. Unlike traditional multi-head attention, MHSA does not use separately learned projections for queries, keys, and values. Instead, we operate directly in the embedding space to preserve the temporal structure and interpretability of the original signal.

Inspired by the work of Li et al. (2019), we introduce a causal 1D convolution over the normalized input sequence for each attention head to get the queries and keys. This allows the attention mechanism to consider local temporal patterns across small groups of time steps, rather than relying on single-point comparisons. The result is a set of per-head, pattern-aware query and key representations. Note that we still use the original (normalized) input for the values.

To complete the causal interpretation, we introduce a rightward shift of the attention matrix, effectively aligning each attention score with the effect of a similar cause observed in the past. This enables the model to attend not just to similar points in the sequence, but to what followed those points, similar to the Simplex projection.

Each attention head learns its own convolutional kernel, enabling diverse pattern representations across the sequence. The outputs of all heads are concatenated to form the final output of MHSA. Within the Transformer architecture, this output is then normalized and passed through a fully connected layer.

More formally, given input embeddings as $X \in \mathbb{R}^{B\times L\times D}$, where B represents batch size, L sequence length, and D embedding dimension we perform the following steps:

1. Normalize the input.
2. Split the embedding dimension D into H attention heads, each of dimensionality $d_h = D/H$.
3. For each head $h \in \{1, \ldots, H\}$, apply a 1D causal convolution, with stride 1 and only padding on the left to prevent data leakage.

$$\hat{Q}_h = \hat{K}_h = \text{Conv1D}_h(X_h) \tag{1}$$

4. Calculate the attention matrix A_h by calculating the pairwise distance between $\hat{Q}_h$ and $\hat{K}_h$, inverting it and applying a Softmax. We add a small constant $\epsilon = 0.001$ for numerical stability.

$$A_h = \text{Softmax}\left(\frac{1}{\text{distance}(\hat{Q}_h, \hat{K}_h) + \epsilon}\right) \tag{2}$$

5. Apply masking to A_h to ensure that each position in the sequence can only attend past time steps, preserving autoregressive structure. Note that this approach differs from regular masking, as the diagonal of A_h will also be masked.
6. Shift the attention matrix A_h one position to the right, to obtain the shifted attention matrix A', aligning each attention score with the effect of a similar past event.
7. Use A' and $V_h = X_h$ to calculate the final output

$$\text{Output}_h = A'_h \cdot V_h = A'_h \cdot X_h \tag{3}$$

8. Concatenate the outputs across all heads to get the final output O for the MHSA.
9. Add residual X and O together.

$$X' = X + \text{MHSA}(X) \tag{4}$$

10. Normalize X' and pass through fully connected layer (FFN).

$$X'' = X' + \text{FFN}(\text{LayerNorm}(X')) \tag{5}$$

The distance between Q and K can be calculated using different distance metrics, such as Euclidean, Manhattan, the infinity norm, or cosine similarity. Figure 1 shows a visual summary of our proposed architecture.

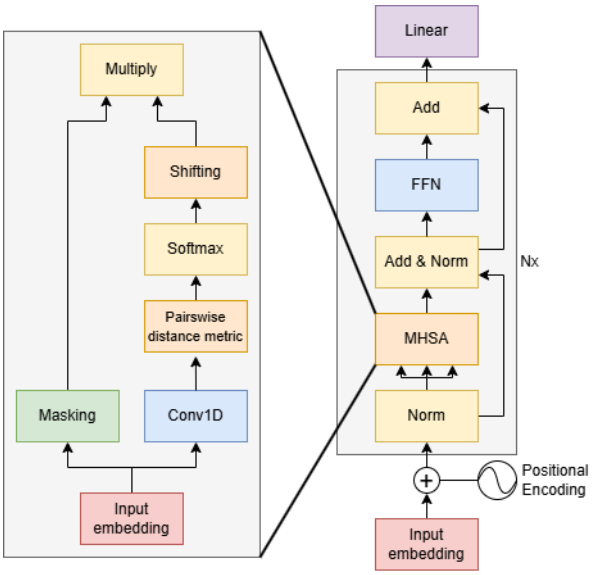

Figure 1: Overview of the Transformer architecture with our proposed Multi-Head Simplex Attention (MHSA). The right panel shows a standard Transformer decoder block where MHSA replaces the typical dot-product attention. The left panel zooms in on the internal mechanism of MHSA: the input embeddings are convolved with causal 1D kernels to extract local temporal patterns, after which a distance-based attention matrix is computed, masked, and shifted to the right. The resulting attention is then applied to the original values to produce the attended output. Each attention head operates independently with its own convolutional kernel.

4. Evaluation

4.1. Experiment setup

The primary goal of our experiments is to evaluate the performance of the proposed MHSA mechanism within the Transformer architecture against two baselines: the Simplex algorithm (Petchey, 2016) and a standard transformer decoder employing convolutional self-attention (Li et al., 2019). The experiments aim to test our hypothesis regarding the performance of different attention mechanisms in time series forecasting using a controlled synthetic dataset.

We utilize a synthetic time series dataset generated according to Equation 6. This dataset is adapted from Li et al. (2019) and Baljan and Rasoolzadeh Baghmisheh (2024) and consists of stacked sine waves with parameters $A_1 = 5$, $A_2 = 100$, $A_3 = 3$, $A_4 = \max(A_1, A_2) = A_2$. The dataset includes additive Gaussian noise ($N_x \sim \mathcal{N}(0, 1)$) and an offset of +72. The total length of the generated time series was 6141 with a sequence length of 140 and predictions shift of 1. For the transformer models, the full dataset was split into training (4500 samples, 75%), validation (500 samples, 8,3%), and test (1000, 16.7%) sets. For the Simplex evaluation, the time series was split sequentially, using the first 5730 data points as historical context to predict the following 250 steps. This setup ensures that the predicted segment of the sine wave aligns with the experiments conducted using MHSA.

Transformer Model Hyperparameters	**Value**
Input length (t_0)	140
Number of layers	1
Embedding dimension	64
FFN hidden dimension	256
Dropout	0.1
Kernel size	5

Table 1: Hyperparameters used for transformer models

Transformer Training Hyperparameters	**Value**
Batch size	32
Learning rate	0.0005
Epochs	300 with early stopping with patience = 45 and $\Delta = 1$
Optimization method	AdamW optimizer
Learning rate step size	10
Learning rate gamma	0.95

Table 2: Hyperparameters used during training for the experiments

$$f(x) = \begin{cases} A_1 \sin(\pi x/6) + 72 + N_x & x \in [0, 12), \\ A_2 \sin(\pi x/6) + 72 + N_x & x \in [12, 24), \\ A_3 \sin(\pi x/6) + 72 + N_x & x \in [24, t_0), \\ A_4 \sin(\pi x/12) + 72 + N_x & x \in [t_0, t_0 + \text{horizon}), \end{cases} \tag{6}$$

For our experiments we compared the following models:

- **MHSA Transformer:** A decoder-only Transformer architecture where the standard attention is replaced by our proposed MHSA module.The hyperparameters can be seen in Table 1. We experimented with variations across:
  - Distance metric: Euclidean, Cosine Similarity, Manhattan, Infinity norm.
  - Number of attention heads: $H \in \{1, 2, 4, 8\}$.
- **Simplex Baseline:** The Simplex algorithm (Petchey, 2016) was implemented with an embedding dimension of 4 and using $k = 4$ nearest neighbours for prediction.
- **Transformer with convolutional attention baseline:** A decoder-only Transformer baseline using the convolutional attention mechanism by Li et al. (2019). It uses the exact same hyperparameters as in Table 1.

The transformers were trained using the hyperparameters in Table 2. The prediction was done using the parameters in Table 3. The model's performance is evaluated on the test set using two standard metrics: Mean Squared Error (MSE) and Mean Absolute Percentage Error (MAPE).

4.2. Experiment results

Looking at the experiment results in Table 4, we can see the performance of all MHSA models being much lower than both

Prediction Parameters	Value
Input length (t_0)	140
Horizon	100

Table 3: Hyperparameters used during prediction for the experiments

the Standard Transformer and Simplex algorithm in terms of mean squared error (MSE). Although the mean absolute percentage error (MAPE) seems to indicate performance similar to that of the Standard Transformer, these results are misleading. When looking at a plot of the prediction, for example, Figure 2 and Figure 3, we can see that our model fails to accurately predict the correct values. Plots of all other experiment setups can be found in Appendix A.

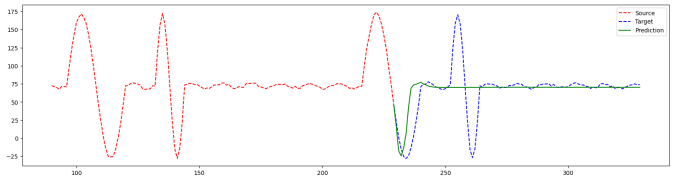

Figure 2: Example of predictions for MHSA model with Euclidean distance and $H = 8$ attention heads.

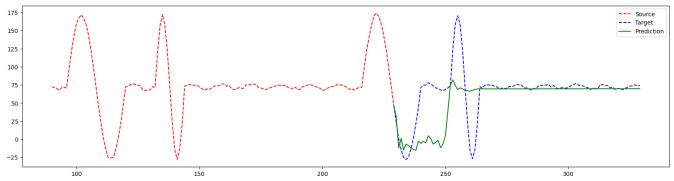

Figure 3: Example of predictions for MHSA model with cosine similarity and $H = 8$ attention heads.

When inspecting the attention matrices for these predictions, we can partly see why this is happening. As previous predictions heavily influence which previous timesteps get attention, incorrect predictions stack up relatively quickly, causing autoregressive predictions to get significantly worse over time.

As an example, we can see in Figure 4 that the incorrect predictions up till timestep $t_0 + 11$ cause the model to give the majority of attention to recent moments, instead of using relevant information. An even worse example can be seen in Figure 5, where all attention is on recent moments.

5. Discussion

When interpreting the results of our research, there are several factors which deserve consideration. Most notably, computational limits heavily influenced the amount of experiments we were able to conduct. As an effect, it was not possible to repeatedly perform an experiment, to allow the possibility for statistical testing using t-tests. These same computational limits also prevented us from performing a gridsearch for optimal hyperparameters. Although unlikely based on our analysis of the results, we cannot rule out the possibility of better performance using different hyperparameters.

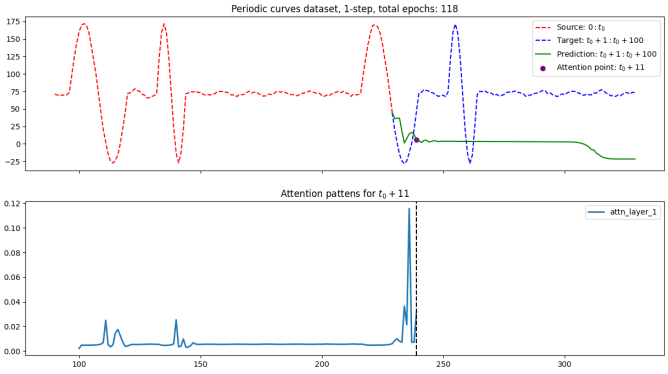

Figure 4: Attention head for Euclidean distance with $H = 1$ attention heads at timestep $t_0 + 11$.

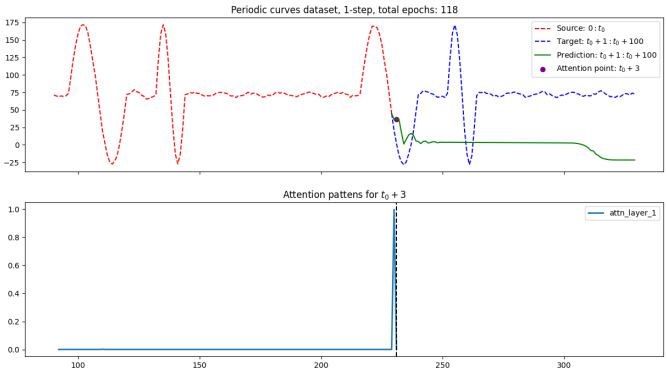

Figure 5: Attention head for Euclidean distance with $H = 1$ attention heads at timestep $t_0 + 3$.

Although the predictions were generated autoregressively one step at a time, we did not explore predicting multiple future steps simultaneously (i.e., direct multi-step forecasting), which may yield different results. Future research could focus on exploring N-step predictions and investigating if this improves performance.

Another aspect which requires further analysis is the MAPE values. Although the plots clearly indicate poor performance, the MAPE values do not show this. The reasons for this discrepancy remain unclear. A possible hypothesis is that MAPE is less vulnerable for big errors when values approaching zero.

6. Conclusion

This paper investigated the performance discrepancy between complex transformer models and a simpler traditional model in time series forecasting. Using the findings suggesting Transformers may not outperform simper approaches on TSK challenges (Zeng et al., 2023), we focussed on changing the attention mechanism. We introduced Multi Head Simplex Attention to try and bridge the gap between Transformer self-attention and Simplex projection by incorporating distance-based similarity, causal convolutions, and an explicit cause and effect shift. Our contribution lies in the novel mechanism and the analysis comparing its performance against Simplex projec-

Model	Distance metric	Number of heads	MSE	MAPE
MHSA	Euclidean	1	139.160	0.080
MHSA	Euclidean	2	120.958	0.279
MHSA	Euclidean	4	105.915	0.241
MHSA	Euclidean	8	80.959	0.268
MHSA	Cosine similarity	1	113.555	0.363
MHSA	Cosine similarity	2	157.932	0.323
MHSA	Cosine similarity	4	134.821	0.319
MHSA	Cosine similarity	8	112.892	0.303
MHSA	Manhattan	1	146.691	0.436
MHSA	Manhattan	2	130.097	0.313
MHSA	Manhattan	4	93.717	0.280
MHSA	Manhattan	8	110.355	0.288
MHSA	Infinity norm	1	133.183	0.469
MHSA	Infinity norm	2	141.015	0.274
MHSA	Infinity norm	4	113.182	0.256
MHSA	Infinity norm	8	90.345	0.259
Standard Transformer with convolutional attention	-	8	1.116	0.267
Simplex Projection	-	-	1.031	0.422

Table 4: Experiment results with Euclidean, Cosine Similarity, Mahattan distance and the infinity norm as distance metric and $H \in \{1, 2, 4, 8\}$ attention heads. Also included are the standard Transformer with convolutional attention as introduced by Li et al. (2019) and Simplex Projection as baselines to compare with. Evaluation is done using mean squared error (MSE) and mean absolute percentage error (MAPE).

tion and a standard convolutional attention Transformer baseline.

Our experiments on a synthetic dataset show that the MHSA performance is worse than both the Simplex Projection and convolutional attention baseline in terms of MSE. Our analysis suggests that the performance of MHSA could be due to the error accumulation in the autoregressive forecasting process.

The main takeaway is that trying to customize the attention mechanism to work more like Simplex projection is not the right way forward if the goal is to improve the performance of a Transformer model when forecasting simple synthetic times series. This approach appears to be too naive, and indicates that more sophisticated approaches are necessary.

Future research could explore the use of more diverse training data. In our study, the models were trained on variations of sine functions, which are relatively simplistic and highly correlated. In contrast, real-world time series data tend to be more complex and less structured. This increased diversity could potentially enhance the Transformer's ability to generalise and extract meaningful patterns. Furthermore, access to a larger and more varied dataset may help the Transformer better capture underlying dynamics that are not present in basic sine functions.

7. Code

To conduct our experiments, we built upon the work from Baljan and Rasoolzadeh Baghmisheh (2024). We implemented additional Python classes for MHSA, the encoder block containing it, and a Simplex Transformer class which connects the decoder block with the fully connected layer. In addition, we implemented the Simplex Algorithm. The experiments are set up as Jupyter notebooks, adhering to the foundations as laid out in the original framework.

The complete source code is available at `https://github.com/FinnAlberts/attention_analysis` for further reference.

For the Standard Transformer with convolutional attention we use as a baseline, we adapted code based on the work from Li et al. (2019), to match the hyperparameters from our experiment setup. The complete source code with these adaptations is available at `https://github.com/FinnAlberts/Transformer_Time_Series`.

References

Baljan, J., Rasoolzadeh Baghmisheh, F., 2024. Analysis of attention mechanism in time series forecasting .

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al., 2020. Language models are few-shot learners. Advances in neural information processing systems 33, 1877–1901.

Devlin, J., Chang, M.W., Lee, K., Toutanova, K., 2019. 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, pp. 4171–4186. URL: `https://aclanthology.org/N19-1423/`.

Li, S., Jin, X., Xuan, Y., Zhou, X., Chen, W., Wang, Y.X., Yan, X., 2019. Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting, in: Advances in Neural Information Processing Systems, pp. 5244–5254. URL: `https://arxiv.org/abs/1907.00235`.

Makridakis, S., Spiliotis, E., Assimakopoulos, V., 2018. The m4 competition: Results, findings, conclusion and way forward. International Journal of Forecasting 34, 802–808. URL: `https://www.sciencedirect.com/science/article/pii/S0169207018300785`, doi:`10.1016/j.ijforecast.2018.06.001`.

Masini, R.P., Medeiros, M.C., Mendes, E.F., 2023. Machine learning advances for time series forecasting. Journal of Economic Surveys 37, 4–46. URL: `https://onlinelibrary.wiley.com/doi/10.1111/joes.12429`.

Nelson, B.K., 1998. Time series analysis using autoregressive integrated moving average (arima) models. Academic Emergency Medicine 5, 739–744. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1553-2712.1998.tb02493.x, doi:https://doi.org/10.1111/j.1553-2712.1998.tb02493.x.

Oreshkin, B.N., Carpov, D., Chapados, N., Bengio, Y., 2020. N-beats: Neural basis expansion analysis for interpretable time series forecasting, in: International Conference on Learning Representations. URL: https://arxiv.org/abs/1905.10437.

Petchey, O.L., 2016. Simplex projection walkthrough. URL: https://doi.org/10.5281/zenodo.57081, doi:10.5281/zenodo.57081.

Salinas, D., Flunkert, V., Gasthaus, J., Januschowski, T., 2020. Deepar: Probabilistic forecasting with autoregressive recurrent networks. International Journal of Forecasting 36, 1181–1191. URL: https://arxiv.org/abs/1704.04110.

Smyl, S., 2020. A hybrid method of exponential smoothing and recurrent neural networks for time series forecasting. International Journal of Forecasting 36, 75–85. URL: https://ideas.repec.org/a/eee/intfor/v36y2020i1p75-85.html.

Sugihara, G., May, R.M., 1990. Nonlinear forecasting as a way of distinguishing chaos from measurement error in time series. Nature 344, 734–741. URL: https://pubmed.ncbi.nlm.nih.gov/2330029/.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I., 2017. Attention is all you need, in: Advances in Neural Information Processing Systems. URL: https://papers.nips.cc/paper/7181-attention-is-all-you-need.

Wu, H., Xu, J., Wang, J., Long, M., 2021. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting, in: Advances in Neural Information Processing Systems, pp. 22419–22430. URL: https://proceedings.neurips.cc/paper/2021/hash/bcc0d400288793e8bdcd7c19a8ac0c2b-Abstract.html.

Wu, H., Xu, J., Wang, J., Long, M., 2022. A time series is worth 64 words: Long-term forecasting with transformers. arXiv preprint arXiv:2211.14730 URL: https://arxiv.org/abs/2211.14730.

Zeng, A., Chen, M., Zhang, L., Liu, Q., Zhou, T., Jiang, Y., Xu, Y., Yan, J., Sun, L., 2023. Are transformers effective for time series forecasting?, in: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 11210–11218. URL: https://ojs.aaai.org/index.php/AAAI/article/view/26317.

Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., Zhang, W., 2021. Informer: Beyond efficient transformer for long sequence time-series forecasting. Proceedings of the AAAI Conference on Artificial Intelligence 35, 11106–11115. URL: https://arxiv.org/abs/2012.07436.

Zhou, T., Ma, Z., Wen, Q., Wang, X., Sun, L., Jin, R., 2022. Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting, in: Proceedings of the 39th International Conference on Machine Learning, pp. 27268–27286. URL: https://arxiv.org/abs/2201.12740.

Appendix A. Plots of experiment results

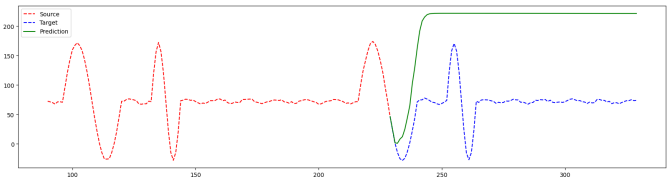

Figure A.6: Example of predictions for MHSA model with Euclidean distance and $H = 1$ attention heads.

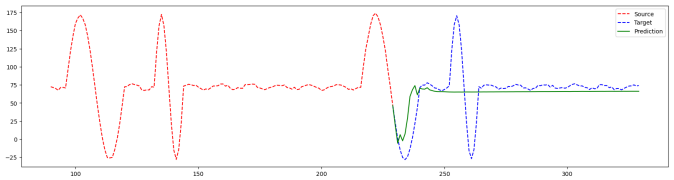

Figure A.7: Example of predictions for MHSA model with Euclidean distance and $H = 2$ attention heads.

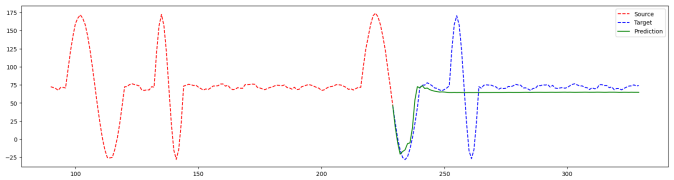

Figure A.8: Example of predictions for MHSA model with Euclidean distance and $H = 4$ attention heads.

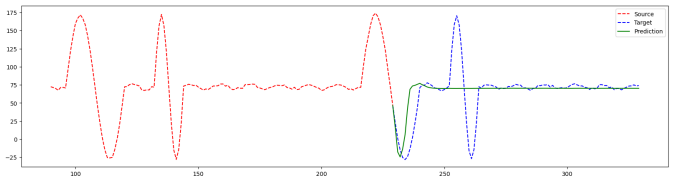

Figure A.9: Example of predictions for MHSA model with Euclidean distance and $H = 8$ attention heads.

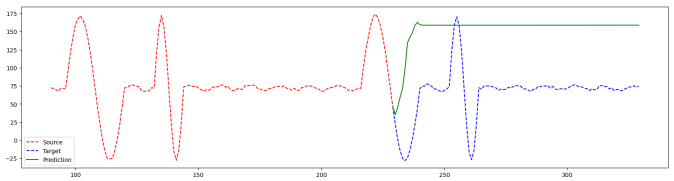

Figure A.10: Example of predictions for MHSA model with cosine similarity and $H = 1$ attention heads.

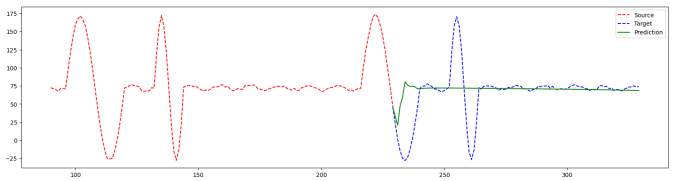

Figure A.11: Example of predictions for MHSA model with cosine similarity and $H = 2$ attention heads.

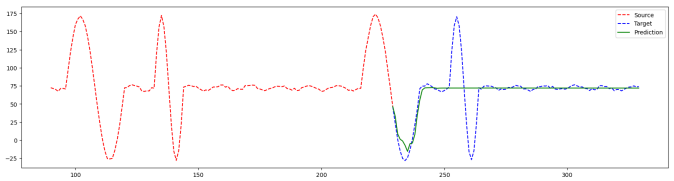

Figure A.12: Example of predictions for MHSA model with cosine similarity and $H = 4$ attention heads.

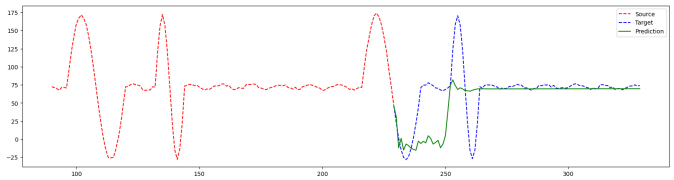

Figure A.13: Example of predictions for MHSA model with cosine similarity and $H = 8$ attention heads.

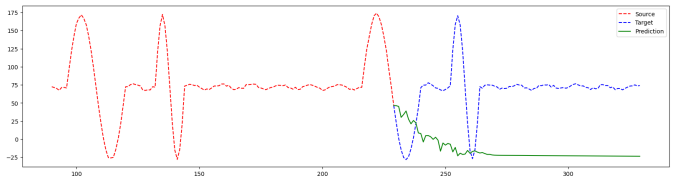

Figure A.14: Example of predictions for MHSA model with Manhattan distance and $H = 1$ attention heads.

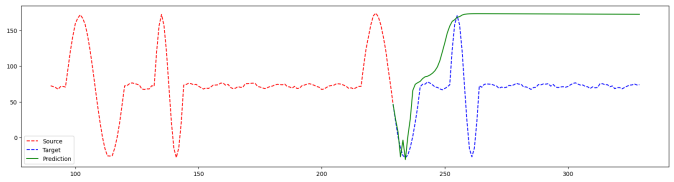

Figure A.15: Example of predictions for MHSA model with Manhattan distance and $H = 2$ attention heads.

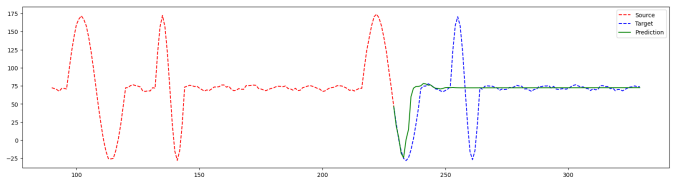

Figure A.16: Example of predictions for MHSA model with Manhattan distance and $H = 4$ attention heads.

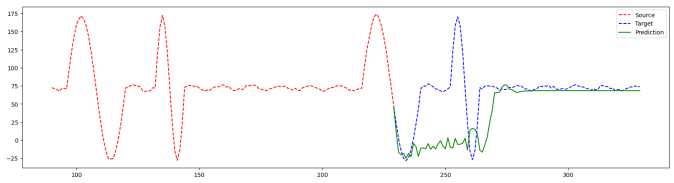

Figure A.17: Example of predictions for MHSA model with Manhattan distance and $H = 8$ attention heads.

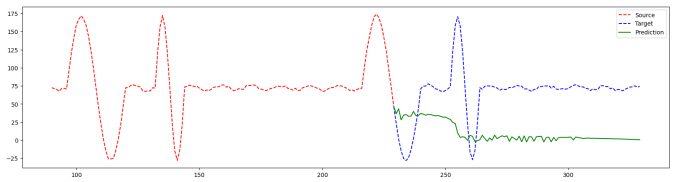

Figure A.18: Example of predictions for MHSA model with the infinity norm and $H = 1$ attention heads.

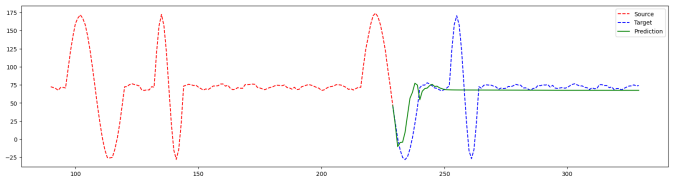

Figure A.19: Example of predictions for MHSA model with the infinity norm and $H = 2$ attention heads.

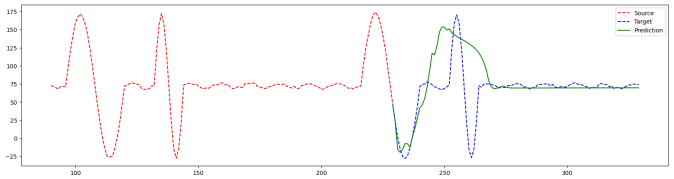

Figure A.20: Example of predictions for MHSA model with the infinity norm and $H = 4$ attention heads.

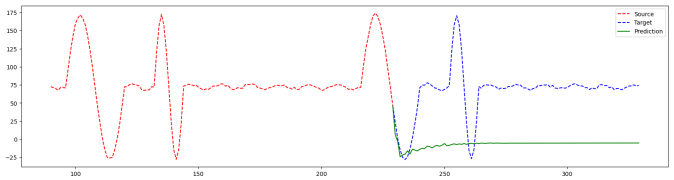

Figure A.21: Example of predictions for MHSA model with the infinity norm and $H = 8$ attention heads.

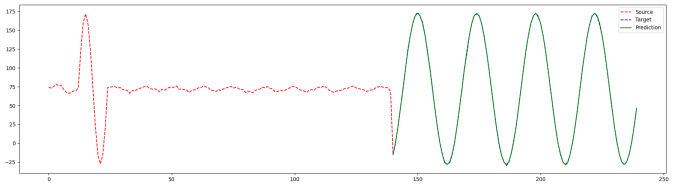

Figure A.22: Example of predictions for Standard Transformer with convolutional attention using $H = 8$ attention heads.

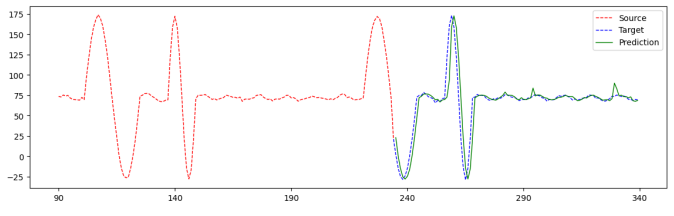

Figure A.23: Example of prediction by Simplex Algorithm.