Temporal Fusion Transformer for thermal load prediction in district heating and cooling networks

Fabian Behrens [a], Stefan Leiprecht [a], Jonas Brantl [a], Matthias Finkenrath [a,*]

[a] *University of Applied Sciences Kempten*
matthias.finkenrath@hs-kempten.de

Abstract

Accurate forecasting of thermal loads is a critical factor for operating district heating and cooling networks economically, efficiently and with minimized emissions. If thermal loads are known with high accuracy in advance, use of renewable energies can be maximized, and fossil generation, in particular in peaking units, can be avoided. Machine learning has already proven to be an efficient tool for time series forecasting in this context. One recent advancement in machine learning is the "Temporal Fusion Transformer" (TFT), which shows especially good results in the area of time series forecasting. This paper examines the performance of TFT in the concrete context of thermal load forecasting for district heating and cooling networks. First, a brief summary of differences between TFT and other machine learning methods is given. Secondly, it is described how the method can be adopted to train a machine learning model for thermal load forecasting. The data to train and evaluate the neural network is based on 8 years of hourly operating data made available from the district heating network of the city of Ulm in Germany. The presented technique is used to produce 72 hours of heating load forecasts for three different district heating grids in the city of Ulm. The results are compared to forecasts of other machine learning methods that have been previously made as part of the publicly funded research project "deepDHC", in order to evaluate if TFT is an improvement to further reduce forecasting uncertainties.

1. Introduction

Precise forecasting of thermal loads is crucial for operating district heating networks efficiently, economically and environmentally friendly. If precise load forecasts are available to the operator, the use of fossil-fuelled peaking boilers can be significantly reduced. In addition, integration of fluctuating renewable into the grid can be maximized. A precise long-term load forecast several days ahead also simplifies fuel ordering, or planned maintenance. Hence this work focuses on thermal load forecasts throughout 72 hours in advance. The data used for the process is based on hourly data from the district heating network in Ulm, a medium-sized city in southern Germany with about 130,000 inhabitants.

2. Related Work

Accurate prediction of heat loads has become an interesting field of application for modern time series forecasting methods. Its importance even increases with a rising global energy demand, decreasing reserves of fossil fuels and the impact of using fossil fuels on climate change (Benalcazar and Kamiński, 2019). District heating and cooling can be a sufficient way to reduce carbon dioxide emissions by optimizing fuel consumption (Werner, 2017). Machine learning has proven to be an attractive option for generating accurate thermal load predictions also in the context of district heating and cooling (e.g. (Saloux and Candanedo, 2018; Leiprecht et al., 2021)).

Different algorithms have been evaluated in recent years for this purpose, such as Adaptive Boosting (AdaBoost) (Freund et al., 1996) and its derivative Extreme Gradient Boosting (XGBoost) (Friedman, 2000), recurrent neural networks (RNNs) like Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) and the Seasonal Autoregressive Integrated Moving Average Exogenous model (SARIMA) (Fang and Lahdelma, 2016).

While the traditional machine learning methods produce decent results, they all have the problem of not being significantly better than statistical methods on many time series problems. In many cases the larger overhead of implementing these methods make them economically less attractive than statistical approaches (Lim and Zohren, 2021). Current research tries to solve this issue by improving the abilities of models of learning from the past, which helps these methods to further improve their accuracy in time series forecasting by reducing their overfitting.

One approach to this issue is the Temporal Fusion Transformer (TFT) (Lim et al., 2021). This new method is an attention based network. TFT is already used in a number of areas for time series forecasting, like meteorology (Wu et al., 2022), medicine (Phetrittikun et al., 2021) and the stock market (Hu, 2021). While there will be some commonalities in input data with forecasting in meteorology, right now there is no research about the performance of TFT for energy demand forecasting. This paper therefore aims to give a first estimation of what results can be expected in this area.

Most of the work that was done on the topic of energy demand forecasting focused on a 24 hour time horizon (Benalcazar and Kamiński, 2019; Xue et al., 2019). This paper instead focuses on an extended forecasting period of up to 72 hours, in order to allow further optimised dispatch planning of power plants and thermal energy storages.

3. Temporal Fusion Transformer

TFT (Lim et al., 2021) is a new approach explicitly developed for time series forecasting. Therefore it brings a number of qualities that are very helpful for training robust forecasting models. Usually machine learning methods use information of the past to learn the behaviour of a time series in order to create accurate forecasts. In order to learn patterns in historic data, neurons have to memorize data they have seen earlier during training. Most of the time this is currently achieved with RNNs (Hochreiter and Schmidhuber, 1997; Jaeger, 2001). However RNNs oftentimes face the problem of expecting that all input data is known, even for time steps in the future. However this is not always possible. For example the temperature in the future can not be known for sure. It can be estimated with the help of weather forecasts, but there exist parameters for which there is no way of knowing them upfront. TFT on the other side does not belong into the class of RNNs, it instead uses a transformer architecture (Vaswani et al., 2017). Transformers use a more advanced method to learn patterns in historic data. As a consequence they support a variety of different types of input data, which enables the usage of features whose values can not be known during prediction. Next some of the advantages of TFT for time series forecasting will be explained.

Firstly TFT supports multi step forecasting. This means that multiple forecasts can be done in one prediction call. For example in the case of this paper one prediction creates forecasts for the next 72 hours with an interval of one hour. Single step methods on the other side would only predict one step at a time. To predict further ahead than one hour this would require the user to do one prediction, add it to the input data and then run the next prediction until the size of the targeted prediction interval is reached. This is undesirable because prediction errors in early time steps can influence the prediction of later time steps. Many new machine learning methods support this type of forecasting, but it is still important to have and can't be done with every of the methods mentioned in this paper. As transformers are not RNNs, another way to learn relations between historic data is needed. TFT uses an attention based method for solving this issue.

TFT supports three different types of input data: temporal data which is known in the future, temporal data which is unknown in the future and static variables. The first group is the most common type of data as known from other forecasting problems. For example the hour of the day for which a prediction is made is such a feature. It is known for historic data during training, but it can also be determined for every future time step. The second type of data is only needed during training, but can be missing when the model is used for predictions. A good example is weather data. Usually the historic weather is known but can not be determined for the future. Other machine learning algorithms would require to guess the weather data or for example use a weather forecast instead of real weather data. However these approaches don't deliver the actual correct values. The model however is trained assuming the provided values are correct, which leads to the predictions being inherently wrong when a forecast is used as input instead. TFT on the other hand makes it possible to use any feature in training even when it is not possible to provide it during prediction phase. The last type allows to add static data that will not change over time, e.g. the holidays of the location of the prediction or the location itself.

Another feature is the support of predicting multiple time series at once. Usually every time series that should be predicted needs its own neural network that is fitted to the training data, in order to create the best possible forecast based on the provided data. TFT provides the possibility to add multiple sets of input data to a model. The model then learns which dataset is used for predicting which time series and fits its model in a way that can predict all time series at the same time. This process can be very helpful because this can save a lot of time. Usually in the process of fitting a model, the hyperparamters will be optimized to. If each time series would need its own neural model multiple of this hyperparameter optimizations would be required. In TFT only one for the model as a whole is needed.

Additionally TFT tries to make the process of working with it more interpretable. Usually neural networks are black boxes that can not be understood in their way of calculating a result. This makes the process of improving a model especially tough when the model just does not seem to get better. TFT solves this issue by a so called multi head attention mechanism. This process works as follows. TFT always calculates the importance of different input features as part of its attention system. These importances can be analysed and can be provided to the machine learning developer. They can then examine which features are important or which impact different features had during one training. Altering features and then evaluating the impact of the change to the performance of a parameter makes it much easier to optimize and understand a model.

The prior explained advantage already includes a last advancement TFT provides. Since the TFT calculates the importance of all input features, it can also realize that a given feature has no importance to the prediction problem. In this case TFT can weigh the effect of the feature with a zero which leads to the feature having no effect in training and prediction. This can also save a lot of time, because the right features do not have to be selected up front by a data scientist.

4. Model Training

The training was done with a little bit more than six years of historic data beginning at 02.09.2014 until the 31.12.2020. This time frame was split into a training and validation dataset with the first 70% being the training dataset and the last 30% being used for validation.

As TFT can analyse features itself in terms of their importance for the problem, almost all features that were available to us were used to train the model. In total those were more than 37 features. Some of the most important ones can be found in Tab. 1

Table 1: Used features for TFT training.

Name	Description
Last Load	The thermal load of the prior hour in MW
avgLoad6/12/24	The average thermal load over the last 6/12/24 hours in MW
Temperature	Current air temperature in Celsius
avgTemp6/12/24	The average temperatur over the last 6/12/24 hours in Celsius
Dewpoint	Current dewpoint in Celsius
Season Sin	One period of a sine wave mapped onto the period of a whole year

Key features are temperature and the current thermal load profile of the district heating network. The temperature is the main factor that changes heating behaviour, especially in residential areas. Therefore the required thermal load strongly correlates with the temperature. The dew point acts like an amplifier of the temperature. In our case both values are very similar most of the time so it also is a good indicator for how the heating demand evolves.

The last load is a good indicator, because most of the time the thermal load demand does not change drastic over a short period of time. Therefore, it usually acts as a good estimation of the next thermal load required. Both of these parameter can also be used as averages over the last few hours. These averages can indicate the overall trend of the current thermal load demand which can help to estimate if the load demand will rise or decline over the next few hours.

The Season Sin feature encodes which day of the year it is at a given prediction point. This can be helpful in improving the understanding of time in the neural networks. While the weather is not exactly the same at the same day over multiple years it can be similar, because it is usually around the same time of the year when the weather gets warmer or colder. Season Sin helps to learn to take this periodic behaviour into account.

The implementation of TFT was not done by ourselves. Pytorch(Paszke et al., 2019) already provides an implementation of TFT which was used in this paper.

5. Methodology

5.1. Metrics

Benchmarking the forecasts is not an easy task, since there is no standardized metric available. The Mean Absolute Percentage Error (MAPE) is probably the most commonly used metric for measuring forecast accuracy. It is widespread in finance or other forecasting applications, especially if enough data is available. The MAPE is dimensionless and independent of the magnitude of the values considered. At the same time, it can be clearly interpreted. A MAPE of zero corresponds to a perfect forecast (Clark, 2013; Armstrong and Collopy, 1992). Its equation can be seen in (1). It is the mean of the sum of the absolute error e_i divided by the real value d_i. n is the number of prediction-load pairs that are used to calculate the error.

$$MAPE = \frac{1}{n}\sum_{i=0}^{n}\frac{|e_i|}{d_i} \tag{1}$$

In addition the Mean Absolute Error (MAE)(Willmott and Matsuura, 2005) was used as a second metric. It is the mean of the sum of n errors. The errors are the absolute deviation of the prediction y_i from the real thermal load x_i. The MAE oftentimes has the disadvantage of being hard to interpret. In many cases the range of values the target of a prediction can have is not known. In this situations it is hard to argue if the measured absolute improvement is significant or not. However in our context this is not the case. For each of the district heating networks considered the thermal loads that can be expected are known. Moreover the unit of the MAE in this case is Megawatts, a unit that is very easily interpretable. The MAE should not be used to compare different district heating networks, because their load profile can differ significantly, however for each individual network the metric can be very helpful for the power plant. It knows which ways it has to provide the thermal load to a given district heating network and how much energy each of these options can provide. In this context absolute values can be very helpful to optimize the energy production for a given district heating network.

$$MAE = \frac{\sum_{i=1}^{n}|y_i - x_i|}{n} \tag{2}$$

So while the MAPE is a good indicator for making general assumptions about the performance of a machine learning method for district heating networks, the MAE can be used for closer evaluation for concrete scenarios. Moreover the MAE can relativise a high MAPE if the absolute target values are quite low.

5.2. District Heating Networks

For the comparison three different district heating networks of the city of Ulm were used. The first network acts as a good general baseline for the performance of a machine learning method, as the network is very consistent and most algorithms evaluated so far perform best on it. It has a total length of 40 km and provides space heating for over 7500 households with an average annual heating demand of 75 GWh, and a heat load ranging from 2 to 22 MW. Water with a temperature between 70°C and 110°C is used as a heat transfer fluid.

Additionally two more networks were selected which have a more complex thermal load profile. The second district heating network uses a combination of steam and hot water for heat transfer. Steam transfer uses steam at a heat of 130°C while the temperature of the water varies between 70°C and 110°C. The network provides space heating to over 13.000 households and has a heat load ranging from 1 to 19 MW.

The third and last district heating network considered, mainly supplies industrial buildings instead of private households, which leads to one more different thermal load profile. It is run with 120°C hot water as transfer fluid. Additional to many factories the district heating network supplies 220 households. It has a heat load ranging from 2 to 25 MW.

While still providing promising forecasts, many of the machine learning methods evaluated prior produced far less optimal results on district heating network two and three. Evaluating the results of TFT on these networks too, can show if the strengths of its new approach help dealing with overall harder to predict scenarios. Also the addition of a more industrial focused district heating network provides more insights for a wider range of use cases.

5.3. Time Frames

As time frame for the comparison the whole year of 2021 was used. With this time frame the evaluation should hold meaningful results for the active usage of the model in a power plant by covering many different scenarios and load profiles of different seasons. Moreover the data is very new, thereby the results can be extrapolated into the future of the net more easily than an older time frame.

In addition to the comparison over the whole year, several shorter time frames are evaluated too. These are:

- Winter (01.01.2021-28.02.2021 and 01.12.2021-31.12.2021)
- Spring (01.03.2021-31.05.2021)
- Summer (01.06.2021-31.08.2021)
- Fall (01.09.2021-30.11.2021)

These four intervals resemble four parts of the year which have different load profiles. Evaluating them makes it possible to further investigate how TFT performs in different scenarios. For example the load profile is very consistent in winter and summer which resulted in pretty good predictions for the already evaluated methods. However the older approaches struggle far more in the spring and fall time frame. During these periods the load profile is much more ambivalent. This could be a problem for these older models because they tend to overfit. Comparison of these time frames will show if TFT can adapt better to learning more unpredictable time series.

5.4. Machine Learning Methods

TFT will be compared to the results of three machine learning methods. Namely LSTM, AdaBoost and XGBoost. Since these methods can not abstract which features are not important to them, these methods were not trained by providing all possible features available as input. To find the most important features for the given machine learning method a feature reduction was performed. The used method was the scikit learn (Pedregosa et al., 2011) implementation of Recursive Feature Elimination (Guyon et al., 2002) with cross validation. The features used for each method can be seen in Tab. 2

Table 2: Features used for training of different machine learning methods.

Method	Used Features
LSTM	loads of the last 6 hours, temperature, season sin, avgTemp24, hour, temperature forecast, dewpoint forecast
AdaBoost	loads of the last 6 hours, season sin, avgTemp12, avgLoad24, hour, temperature forecast for the next 5 hours
XGBoost	season sin, avgTemp12, avgLoad24, hour, loads of the last 6 hours, temperature forecast for the next 3 hours

The features correspond to the features explained in section 4. The amount of last loads and forecasts used in AdaBoost and XGBoost vary between the models for each of the three different nets in order to further improve the results of the nets. The range in which those parameters lie is three to six hours. The forecasts of all LSTM neural networks are the same as the LSTM predicts the load as a multi step while AdaBoost and XGBoost use a single step method.

6. Discussion of Results

The promising results of TFT can be seen in Fig. 1. TFT beats all other machine learning methods on every district heating network evaluated. Moreover this is special, because it is the first time in our investigation that one method is the best one for any district heating network it was tested on. For example without TFT, LSTM would be the best way to predict network one while XGBoost is the best method for network two and three. Moreover the reduction of the error is impressive. On the easiest network it beats LSTM by two percent points. This is a good result, however it is not that relevant for the facility, because looking at the absolute error, it already lies beneath one MW. However the improvement is much stronger in the tougher to predict networks two and three. In both cases the MAPE of TFT is almost half as high as the MAPE of XGBoost. This indicates how the attention-based approach of TFT is way better in generalizing the problem than older machine learning methods and thereby avoids overfitting.

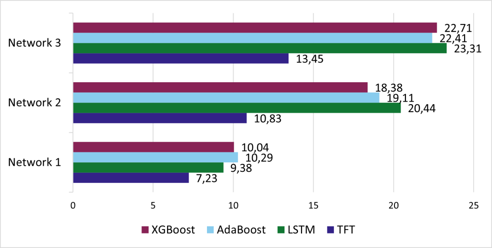

Figure 1: MAPE for the whole year of 2021

As a next step the four different time frames will be evaluated to gain a better understanding why TFT beats the other machine learning methods by such a huge margin. The results for the spring time frame can be seen in Fig. 2. The results of this time frame early in the year are quite similar to the overall results of the evaluation. TFT performs best on every net, but while the difference is only around two percent points better for network one the error is around 4 percent points better for network two and 8 percent points better for network three. The MAPE for all methods is higher in the spring than in the overall year. The reason is that spring and fall is more difficult to predict because the load profile does not behave as similar as it does in the summer and winter months.

Comparing the difference between the overall MAPE and the spring MAPE for each of the machine learning methods shows that the difference in percent points is quite similar across all methods. This indicates that TFT is not per definition better in predicting spring times. The big improvement is only so huge because the predictions of TFT are overall better.

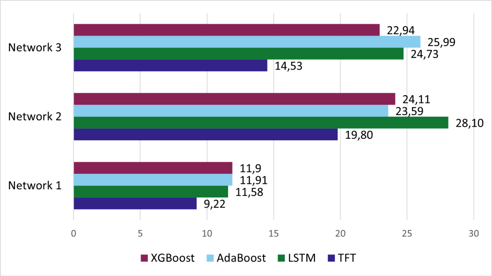

Figure 2: MAPE of the spring time frame.

Fig. 3 shows the different results for the summer time frame. Errors in the summer time frame are generally lower and closer to the overall year MAPE than the results of the spring time frame. Again TFT is the best machine learning method evaluated for all three district heating networks. Comparing the summer results with the spring results, district heating network three stands out. While in network one and two the MAPE is much lower in summer than in spring, in district heating network three the reduction of the error is not that significant. TFT even performs worse in summer than in spring on network three. All three district heating networks are located in the same city, so the weather during this period was the same for all three networks and should not have been more unpredictable for network three. This leaves two possibilities for why TFT is worse in summer than in spring. Either the network changed which leads to the historic data being less optimal for a prediction of the current form of the district heating network, or the reported load values during the prediction time frame were erroneous, which lead to bad input values for the prediction. The load data during the prediction time frame did not have any issues, so the first problem probably caused this result.

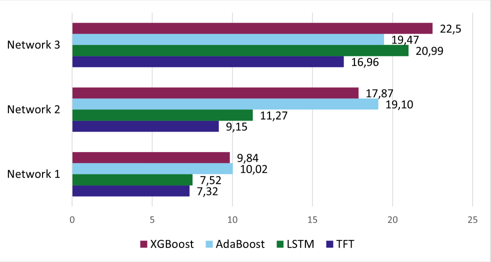

Figure 3: MAPE of the summer time frame.

Fig. 4 shows the results of all machine learning methods for the fall time frame. Similar to the spring time frame TFT really performs much better than its alternatives during this period. For network two and three the error is reduced by almost 50% over the second best method and even on the already well performing network one the error is reduced by around 30%. Together with the results of Fig. 2 the assumption of TFT being especially good for less predictable time frames can be proved. This is a very good trait for a machine learning method used for thermal load forecasting, because some of the most important features are weather data and weather forecasts. Even if a forecast is very accurate, the nature of a forecast is, that it is never a safely known value. This makes a very adaptable system like TFT preferable. If compared with the spring time frame also all MAPEs are a bit better. In both time frames the thermal load provided by the district heating networks is quite similar, which again would indicate that the spring time frame of 2021 was less predictable in its behaviour compared to the fall time frame. Moreover when compared to the summer time frame it seems like the predictions in fall would be better than in the summer. This is as misconception created by the MAPE having a percentage as error unit. In summer the thermal load is far lower in all of the evaluated district heating networks. This results in larger MAPEs even for small absolute errors. Considering the absolute errors, the predictions for the summer are actually better than for fall, e.g. with TFT for network one the MAE for summer is 0.22MW and for fall it is 0.49MW. So the summer predictions are actually better even if the MAPE is worse.

Lastly the winter time frame is considered. The results can be seen in Fig. 5. Again TFT is the strongest method for each of the three inspected networks. For network two and three TFT beats XGBoost by about 10 percent points. Furthermore, TFT also is a notable improvement in network one even though the error reduction is just 2.3 percent points. Moreover the MAPEs in the winter time frame is the best of the whole year. This has two reasons. First the prediction is overall very good. 5.52% MAPE resembles a MAE of 0.78MW which is fairly low in the context of the evaluated district heating network. The second fact is again one of the properties of the MAPE. In the winter months the thermal load is quite high, which leads to lower MAPEs even on similar large absolute errors.

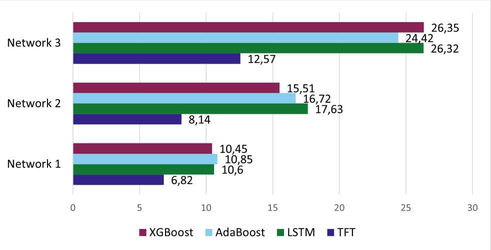

Figure 4: MAPE of the fall time frame.

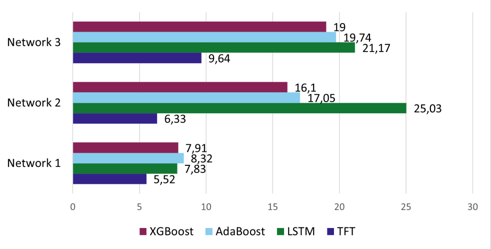

Figure 5: MAPE of the winter time frame.

For more insight on the actual absolute error, Tab. 3 shows all predicted MAEs for TFT. A good indicator for the performance of the TFT is that most errors are below 1MW. Only in the very unpredictable network three the absolute error lies on average above 1MW. The networks considered have thermal loads in the range of 10 to 20 MW for most of the year, so the calculated error is very little. The absolute errors of summer also show the issue of comparing the MAPEs of different time frames. The MAEs are the lowest for every network in the summer time frame but still the summer MAPEs were the worst.

Table 3: Absolute errors of TFT in MW.

Timeframe	Network 1	Network 2	Network 3
Overall	0.57	0.56	1.00
Spring	0.79	0.96	1.28
Summer	0.22	0.19	0.36
Fall	0.49	0.45	0.92
Winter	0.78	0.67	1.44

Evaluation of all time frames has shown that out of the list of evaluated methods, TFT is the best machine learning method for the use case of time series forecasting in any scenario. Further comparing the MAPEs in different time frames showed that TFT is not really better in predicting any of the time frames as it also struggled with predicting spring and fall more than summer and winter. However TFT predictions were overall always way better than their competitors. This results in overall lower errors in all time frames which leads to very competitive overall errors because the reduction of the error in spring and fall also reduces the overall error far more than improvements in the summer and winter forecasts. A low overall error translates to a more robust system.

7. Conclusion-and-Future-Work

Use of TFT could contribute to further improve thermal load forecasting. This paper presents first results of benchmarking TFT against different machine learning-based forecasting approaches for district heating and cooling networks. As a starting point, the predictions of these different methods were analysed on multiple time frames and over multiple district heating networks. All measured data, including statistically optimized point weather forecasts, were automatically pre-processed prior to the actual training and validation steps. The models predicted 72 hours in advance. The predictions were benchmarked against three other machine learning methods that where evaluated in previous works. TFT showed to have better MAEs and MAPEs over all experiments, making it a very strong candidate for thermal load forecasting in any scenario. Especially the improvements in spring and fall forecasts above other methods is a big improvement.

This paper used one specific model for each of the networks to predict the thermal loads. One of the advantages of TFT is to be able to train one model for the prediction of multiple time series. As a next step it should be investigated if a model trained to predict the thermal load of multiple networks still holds the same results. This could be a very important step in making machine learning for thermal load forecasting more viable, because it would reduce training effort and cost for the power plants immensely.

Acknowledgments

This work was funded by the Federal Ministry of Economics and Energy of Germany under the funding code 03EN3017, with additional financial and in-kind support by the project partners AGFW, Fernwärme Ulm GmbH and ZAK Energie GmbH. The responsibility for the content of this publication lies with the authors.

References

Armstrong, J. and Collopy, F. (1992), ‘Error measures for generalizing about forecasting methods: Empirical comparisons’, *International Journal of Forecasting* **8**(1), 69–80. doi: https://doi.org/10.1016/0169-2070(92)90008-W.

Benalcazar, P. and Kamiński, J. (2019), ‘Short-term heat load forecasting in district heating systems using artificial neural networks’, *IOP Conference Series: Earth and Environmental Science* **214**, 012023. doi: 10.1088/1755-1315/214/1/012023.

Clark, M. (2013), ‘A comparison of correlation measures’, *Center for Social Research, University of Notre Dame* **4**.

Fang, T. and Lahdelma, R. (2016), ‘Evaluation of a multiple linear regression model and sarima model in forecasting heat demand for district heating system’, *Applied Energy* **179**, 544–552. doi: https://doi.org/10.1016/j.apenergy.2016.06.133.

Freund, Y., Schapire, R. E. et al. (1996), Experiments with a new boosting algorithm, *in* ‘icml’, Vol. 96, Citeseer, pp. 148–156.

Friedman, J. (2000), ‘Greedy function approximation: A gradient boosting machine’, *The Annals of Statistics* **29**. doi: 10.1214/aos/1013203451.

Guyon, I., Weston, J., Barnhill, S. and Vapnik, V. (2002), ‘Gene selection for cancer classification using support vector machines’, *Machine Learning* **46**, 389–422. doi: 10.1023/A:1012487302797.

Hochreiter, S. and Schmidhuber, J. (1997), ‘Long Short-Term Memory’, *Neural Computation* **9**(8), 1735–1780. doi: 10.1162/neco.1997.9.8.1735.
URL: *https://doi.org/10.1162/neco.1997.9.8.1735*

Hu, X. (2021), Stock price prediction based on temporal fusion transformer, *in* ‘2021 3rd International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI)’, pp. 60–66. doi: 10.1109/MLBDBI54094.2021.00019.

Jaeger, H. (2001), ‘The" echo state" approach to analysing and training recurrent neural networks-with an erratum note”, *Bonn, Germany: German National Research Center for Information Technology GMD Technical Report* **148**.

Leiprecht, S., Behrens, F., Faber, T. and Finkenrath, M. (2021), ‘A comprehensive thermal load forecasting analysis based on machine learning algorithms’, *Energy Reports* **7**, 319–326. The 17th International Symposium on District Heating and Cooling. doi: https://doi.org/10.1016/j.egyr.2021.08.140.

Lim, B., Arık, S. Ö., Loeff, N. and Pfister, T. (2021), ‘Temporal fusion transformers for interpretable multi-horizon time series forecasting’, *International Journal of Forecasting* **37**(4), 1748–1764. doi: https://doi.org/10.1016/j.ijforecast.2021.03.012.

Lim, B. and Zohren, S. (2021), ‘Time-series forecasting with deep learning: a survey’, *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* **379**(2194), 20200209. doi: 10.1098/rsta.2020.0209.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J. and Chintala, S. (2019), Pytorch: An imperative style, high-performance deep learning library, *in* ‘Advances in Neural Information Processing Systems 32’, Curran Associates, Inc., pp. 8024–8035.
URL: *http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf*

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V. et al. (2011), ‘Scikit-learn: Machine learning in python’, *Journal of machine learning research* **12**(Oct), 2825–2830.

Phetrittikun, R., Suvirat, K., Pattalung, T. N., Kongkamol, C., Ingviya, T. and Chaichulee, S. (2021), Temporal fusion transformer for forecasting vital sign trajectories in intensive care patients, *in* ‘2021 13th Biomedical Engineering International Conference (BMEiCON)’, pp. 1–5. doi: 10.1109/BMEiCON53485.2021.9745215.

Saloux, E. and Candanedo, J. A. (2018), ‘Forecasting district heating demand using machine learning algorithms’, *Energy Procedia* **149**, 59–68. 16th International Symposium on District Heating and Cooling, DHC2018, 9–12 September 2018, Hamburg, Germany. doi: https://doi.org/10.1016/j.egypro.2018.08.169.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. and Polosukhin, I. (2017), ‘Attention is all you need’, *CoRR* **abs/1706.03762**.
URL: *http://arxiv.org/abs/1706.03762*

Werner, S. (2017), ‘International review of district heating and cooling’, *Energy* **137**, 617–631. doi: https://doi.org/10.1016/j.energy.2017.04.045.

Willmott, C. and Matsuura, K. (2005), ‘Advantages of the mean absolute error (mae) over the root mean square error (rmse) in assessing average model performance’, *Climate Research* **30**, 79. doi: 10.3354/cr030079.

Wu, B., Wang, L. and Zeng, Y.-R. (2022), ‘Interpretable wind speed prediction with multivariate time series and temporal fusion transformers’, *Energy* p. 123990. doi: https://doi.org/10.1016/j.energy.2022.123990.

Xue, P., Jiang, Y., Zhou, Z., Chen, X., Fang, X. and Liu, J. (2019), ‘Multi-step ahead forecasting of heat load in district heating systems using machine learning algorithms’, *Energy* **188**, 116085. doi: https://doi.org/10.1016/j.energy.2019.116085.