Western Norway
University of
Applied Sciences

Project report in DAT255 – Deep Learning Engineering

Norwegian Residential Rent Price Predictor using DNN and XGBoost + Trend Models

Date: *25 April 2025*

Candidate number 532

I confirm that the work is self-prepared and that references/source references to all sources used in the work are provided, cf. Regulation relating to academic studies and examinations at the Western Norway University of Applied Sciences (HVL), §10.

Problem description

In this project I’ve built an end-to-end system that can predict the average monthly rent (NOK) for Norwegian homes based on region, size (rooms) and year.

- **Motivation** Most lease contract in Norway rarely exceeds three years, yet national statistics are published with roughly a 1-year delay. This mismatch makes it difficult to predict the market as both a landlord and a tenant.
- **Project Goal** Deploy a lightweight Streamlit service that both displays historic SSB tables (1) and can predict leasing prices for a region and number of rooms (2).
- **Why Deep Learning?** The price surface is non-linear in year and region and exhibits different curvatures in price based on the number of rooms. Deep learning is well suited to this problem due to the complex interaction between the datapoints.

Data

The dataset was sourced from SSB and downloaded into an excel file. The data was then processed so that it could enhance the training and testing of the models. Additionally, macroeconomic data on Norges Bank's interest rated (styringsrente and utlansrente) was manually constructed and included in the training and test sets.

- **Why these datasets?** The data was chosen for multiple reasons. Firstly, it was public domain data, it was readily available and from trusted sources.

  Pros:
  - Covers multiple regions and room sizes.
  - Contains over 1000 samples across 12 years.
  - Granular yearly prices enable temporal modelling.

  Cons:

  - Only yearly resolution (no monthly data).
  - "Unprecise" data for bigger regions where leasing prices can vary a lot.

- **Data augmentation and Missing data** There were some missing datapoints for multiple regions, which made it necessary to approximate leasing prices. This was done by looking at leasing price increases or decreases from the "hele landet" data. From there a percental decrease or increase from a known datapoint in the region was used to approximate a missing one. In some regions there were no data for different room sizes, and they were not approximated since there were no obvious data points to approximate from.
- **Preprocessing included:**
  - Reshaping from wide to long format.
  - Merging with interest rate data.
  - One hot encoding for categorical features (Region).
  - Deriving new features such as InterestSpread and RoomSize.

Sources	Period	Granularity
SSB table 09895	2012-2024	Region, RoomType and year
Norges Bank MPC series	2012-2024	Yearly

Model implementation

We implemented and compared two primary models:

1. DNN (Deep Neural Network)
2. XGBoost + Linear Trend Hybrid

DNN

- Implemented in Keras with TensorFlow backend.
- Architecture:
  - Dense(256, relu) + BatchNorm + Dropout(0.4)
  - Dense(128, relu) + BatchNorm + Dropout(0.3)
  - Dense(64, relu)
  - Output: Dense(1)
- He normal initialization used to improve convergence.
- Optimizer: Adam(learning rate = 0.0003)
- Loss: Mean Squared Error (MSE), Metric: Mean Absolute Error (MAE)
- Early stopping (patience = 10) applied to avoid overfitting.

XGBoost + Trend

- Gradient boosting on structured data.
- Yearly trend captured with separate LinearRegression model.
- Final predictions is an average of XGBoost and Trend model outputs.

Training Strategy

- Train/validation split on data from 2012-2024 (20% validation).
- Separate models trained per room type (1-5 rooms).
- Scikit-learn’s train_test_split, StandardScalar and ColumnTransformer used.

Evaluation

Evaluation metric R^2 score (coefficient of determination).

R^2 was chosen for its interpretability in regression tasks. It quantifies how much change in rental prices can be explained by the model.

Room Type	DNN R^2	XGBoost + Trend R^2
1 rom	-1.822	0.829
2 rom	0.574	0.672
3 rom	0.272	0.678
4 rom	-0.117	0.629
5 rom eller flere	0.122	0.698

As is clearly visible from the table the XGBoost + Trend model consistently outperformed the DNN model in all room categories. Particularly when going beyond 2024, the trend component gave the XGBoost model a clear edge over DNN.

Improvements The DNN performed poorer than expected with data that should be advantageous for DNN models. I could have tried to remove outliers from the data to improve accuracy. I could also have given the model more datapoints from 2012 to 2024 as there were too few datapoints that were also possibly unrelated.

Deployment

The solution was deployed using a Streamlit application, offering a functional prototype for public demonstration or stakeholder feedback.

Framework:

- Streamlit provides seamless UI integration with Python backend.
- Rapid prototyping, interactive inputs, and metrics display.

Features:

- Room types, region and model selection for the user.
- Real time model prediction with R^2 scores.
- Cache management.

Future enhancements:

- Hosting for easier public access.
- CI/CD pipeline for automated data and model updates.
- Adding more analytical tools for the user to interpret the results.

Conclusion

The project successfully demonstrated the performance of two regression models for predicting rental prices with the same data. While DNN may have been able to capture complex patterns, it failed to recognize more general patterns compared to XGBoost + Trend.

To get different results using the same models the parameters for the data handling would have to be changed.

Overall, the project delivered a functional prototype through a Streamlit application, and lays a solid foundation for future enhancements such as public deployment, continuous updates, and more advanced analytics.

References

- Norges Bank historical interest rates https://www.norges-bank.no
- Statistics Norway (SSB): https://www.ssb.no/
- OpenAI ChatGPT (Writing assistance and debugging)
- Deepseek Coder (code suggestions and validation)