Ground water level prediction using LSTM model

The objective of the work was to predict the pre-monsoon and post-monsoon water levels in wells of three different localities included in Kollam, Pathanamthitta and Thiruvananthapuram. For the prediction, we mainly considered the last 15 years of water level of pre-monsoon and post-monsoon data. Here the nature of the data is sequential type, which is collected over a period of 15 years from two different types of wells (Dick well and bore well) during pre(2007 -2022 excluded 2020 because the sample was not collected during covid period) and post(2007- 2021, excluded 2022 data: data no available) monsoon seasons. By using this type of data, we can make future predictions by different methods. One common method is the regression models, which are statistical models that are used to predict a continuous value (such as water level) from a set of independent variables (such as time)[1]. Another method is the machine learning algorithms, which include different models that can be used for prediction by learning the models[2]. Long Short-Term Memory (LSTM) models are one of the best machine learning-based models that have been widely used for the prediction of sequential data. Which is designed by recurrent neural network (RNN) architecture to overcome the limitations of traditional RNNs in capturing and remembering long-term dependencies in sequential data. LSTMs were first introduced by Sepp Hochreiter Jürgen Schmidhuber in 1997[3]. The key idea behind LSTMs is the introduction of memory cells, which are capable of storing information over long periods of time. These memory cells are connected through a series of gates that regulate the flow of information, allowing the network to selectively remember or forget information as needed. The gates include an input gate, a forget gate, and an output gate. The input gate determines how much of the new input should be stored in the memory cell. The forget gate controls the amount of information to be discarded from the memory cell, and the output gate regulates the amount of information to be output from the memory cell to the next time step. LSTMs are particularly effective in processing and predicting sequences of data, such as in natural language processing (NLP) tasks, speech recognition, and time series analysis. LSTM models included different types, which mainly included Vanilla LSTM or Standard LSTM, Stacked LSTM, Bidirectional LSTM, and CNN LSTM[4]. All these models are different from one another based on the data used for prediction. For the groundwater level prediction, here we used the model was

Vanilla LSTM. Because the Vanilla LSTM is one of the best models for predicting univariate data *(*a type of data that consists of observations on only a single characteristic or attribute). Here the data is a sequential type. In the problem space, we mainly took the yearly based water content of each locality of three different districts. Our aim was to predict the future water level content of each locality of the district based on the given pre and post-monsoon data. Here we mainly considered the last 15 years of data based on location from three different districts with 127 observations of pre and post-monsoon data. Here the data has only one feature (attribute), the yearly based water level. The process of the vanilla LSTM model based on the present study encompasses various aspects such as pre-processing, model architecture, training, and evaluation of the model. The main step involved in the program as follows

1. Read the data from an Excel file.
2. Split the data into two sets: a training set and a test set.
3. Build the LSTM model
4. Train the LSTM model based on the training set.
5. Evaluate the model's performance based on the test set.
6. Make predictions on the test set.

The input Excel file contained the details regarding district, location, year and well types. First, we read the data for each sheet into a dictionary of data frame. Then read each location's pre and post-monsoon data from the data frame for pre-processing the data. Here we considered the location-based water level data from each location to predict the water level of the coming year of each location. Then we split the data for training and testing. In the vanilla LSTM model, we used an equal number of data for training and testing from the actual data set of each location based on the year. Then we stored the data in X and Y variables. Here we have two types of data sets, pre-monsoon and post-monsoon data. Our aim was to forecast each locality's pre and post-monsoon water level based on the previous years of pre and post-monsoon data from each location. So here we generated two sets of training and test data from these data sets for each location and stored them as separate arrays of data, such as X1 and y1 for training and test set of pre-monsoon data and X2 and y2 for training and test set of post-monsoon data. Then we reshape the training data set of each input data set by using reshape() function, which is necessary for the data to be reshaped to match the input

shape expected by the lSTM model. Then an LSTM model was built using sequential model API from Keras. The model consists of an LSTM layer with 50 units and Relu activation followed by a dense output layer. The model was compiled using Mean squared error(MSE) and the Adam optimizer. Then the model was trained using pre-monsoon and post-monsoon data('X1'y1')and ('X2','y2 ') for 100 epochs and a batch size of 32. The function print the MSE values for both seasons and returns the mean of the predictions for both seasons. From the prediction, we could identify that the predicted model showed promising results in forecasting the pre and post-monsoon value of the groundwater prediction of each location.

Model Performance Analysis

The model performance was evaluated using Mean Squared Error (MSE). The results varied across locations:

- Ailara and Akkal showed low MSE values, indicating stable seasonal groundwater behavior.
- Achenkovil showed comparatively higher pre-monsoon error, suggesting greater variability or possible sensitivity to limited training samples.

Since the evaluation was performed on the training dataset, these results reflect model fitting performance rather than generalization performance.

Limitations of the Study

- Limited dataset size (15 yearly observations per location)
- Univariate modeling (no rainfall, temperature, or climatic variables included)
- Evaluation performed on training data
- No cross-validation or walk-forward validation applied

Future Scope

Future work could include:

- Incorporating climatic variables (rainfall, temperature)
- Applying walk-forward validation
- Comparing with ARIMA or statistical time-series models
- Hyperparameter tuning and model optimization

1. Schneider, A., Hommel, G., & Blettner, M. (2010). Linear regression analysis: part 14 of a series on evaluation of scientific publications. Deutsches Ärzteblatt International, 107(44), 776.
2. Sarker, I. H. (2021). Machine learning: Algorithms, real-world applications and research directions. SN computer science, 2(3), 160.
3. Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780.
4. https://colah.github.io/posts/2015-08-Understanding-LSTMs/