**Wind Speed Prediction Report**

**November 2018**

1. **Introduction**

**1.1 The Goal of Project**

The goal of this project is to revise the wind speed from all kinds of weather sources with extra information, such as additional information nearby the turbines, to build a baseline model. After building the baseline model, add a second-layer linear or non-linear model to combine the results from the first layer to get a better result.

**1.2 Dataset**

There are two wind farms’ data. In the farm ID (57f2a7f2a624402c9565e51ba8d171cb), there are 58 turbines. And in the farm ID (WF0010), there are 66 turbines. For each turbine, there are information from 5 weather sources (EC0, GFS0, WRF0, IBM0 and ENS\_AVG0), including basic information and some shift data. Note that for weather source IBM0, it does have less information. During the Part 5, we add one new weather source - WRF\_EC\_SW\_ML0.

**1.3 The Flow of Project**

In this part, we briefly introduce the project. In part 2, we would build a second layer (Linear or Non-Linear) to combine the wind speed from the first layer. Next, we want to improve models at part 3. In part 4, we consider doing more feature engineering to explore more information from data. Then, in Part 5, we add the simple neural network – fully connected layer to get better results. In Part 6, we try to use sequence model RNN to capture some sequence information among data. At last, conclude the project and describe what we could do in the future. The flow of this project is as Figure 1.

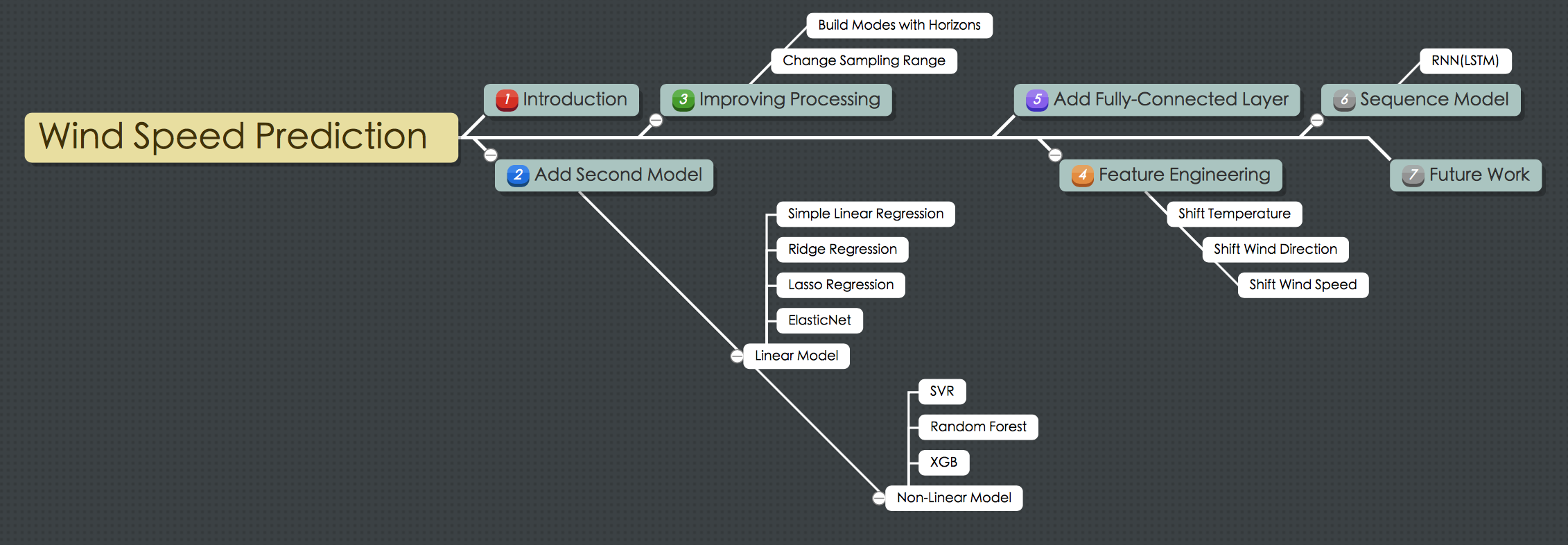


Figure 1. The Flow of Project

**1.4 Overview of Models**

In this project, we try to train and test many different models to find better one. As figure 2, there is an overview of models that we have used or plan to try in the future. In the later section, we will concretely introduce the models and their performance results.

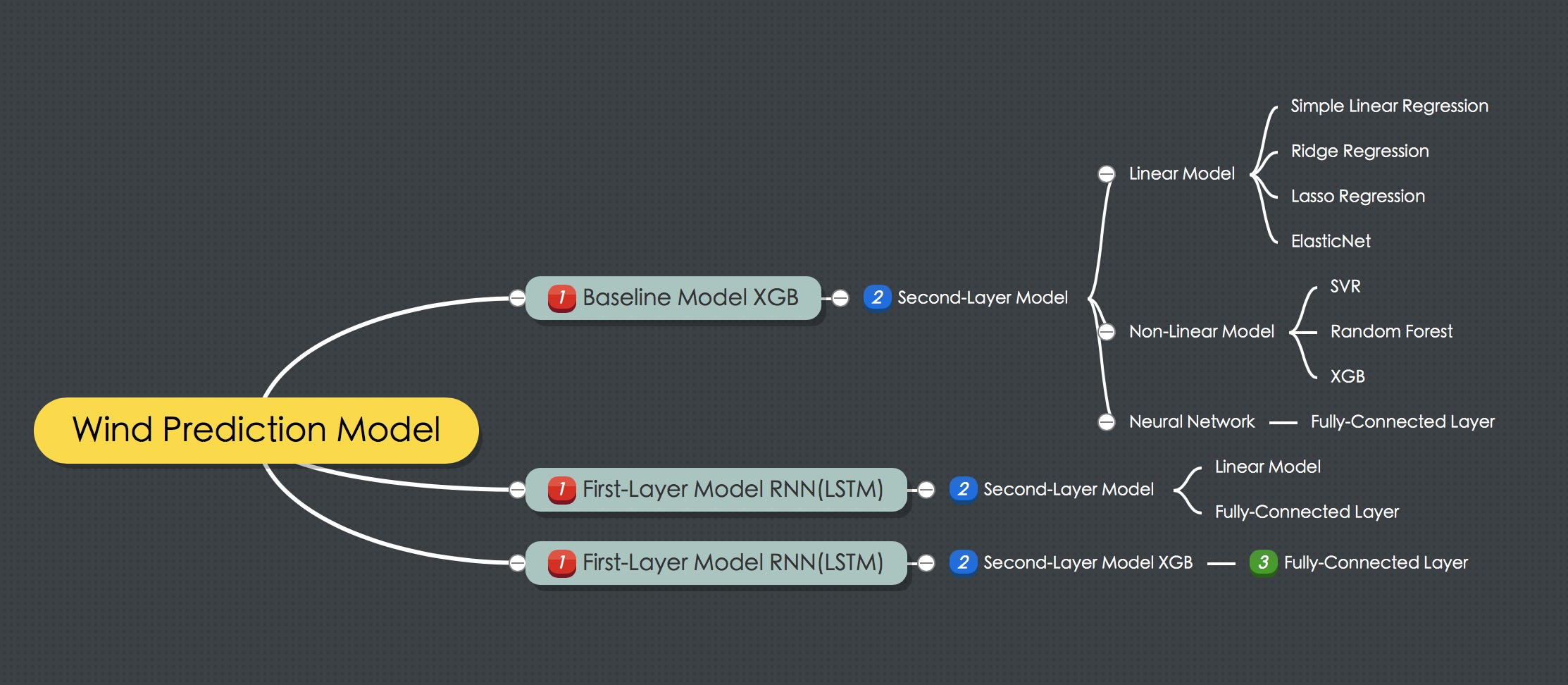
****

Figure 2. Overview of Models

1. **Experimental Models with 5 Weather Sources** 
   1. **Baseline Model**

The baseline model is just to use XGB model to revise the wind speed from each weather source. The results of baseline model are in Table 1. The training data are one-year, and testing data are one-month.

Table 1. The Results of Baseline Model

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| farm\_57f2a7f2a624402c9565e51ba8d171cb 58 Turbines Sampling 4-12 | | | | | |
|  | EC0 | ENS\_AVG0 | GFS0 | IBM0 | WRF0 |
| Original (Std) | 2.139 | 2.285 | 2.196 | 1.954 | 2.110 |
| Revised (Std) | 1.849 | 1.921 | 2.012 | 1.895 | 1.807 |
| Original (RMSE) | 3.609 | 4.244 | 3.365 | 2.897 | 2.264 |
| Revised (RMSE) | 1.887 | 2.027 | 2.071 | 2.082 | 1.859 |

We could see from the Table 1, the baseline model with extra information could reduce the value of STD and RMSE for all weather sources.

* 1. **Add Linear Models as the Second Layer**

We firstly consider adding one linear model as the second layer to improve the performance. There are four linear models applied as Table 2.

Table 2. Four Linear Models

|  |  |  |
| --- | --- | --- |
| Index | Model | Regularization |
| 1 | Linear Regression | None |
| 2 | Lasso Regression | L1 |
| 3 | Ridge Regression | L2 |
| 4 | ElasticNet | L1 + L2 |

For the Lasso and Ridge, there are different regularizations L1 and L2. L1 is the absolute value of penalty factor, and L2 is the square of penalty factor. We use the *LassoCV* and *RidgeCV* in the experiment. The reason is that they are able to automatically choose the best parameter.

We use basic information from each weather source and information after data pre-pre-processing to revise the wind speed with XGB model. Then, we combine the results from first layer to get a better result with linear model. The structure of XGB and linear model is as Figure 3. The specific input attributes please see appendix A.1.

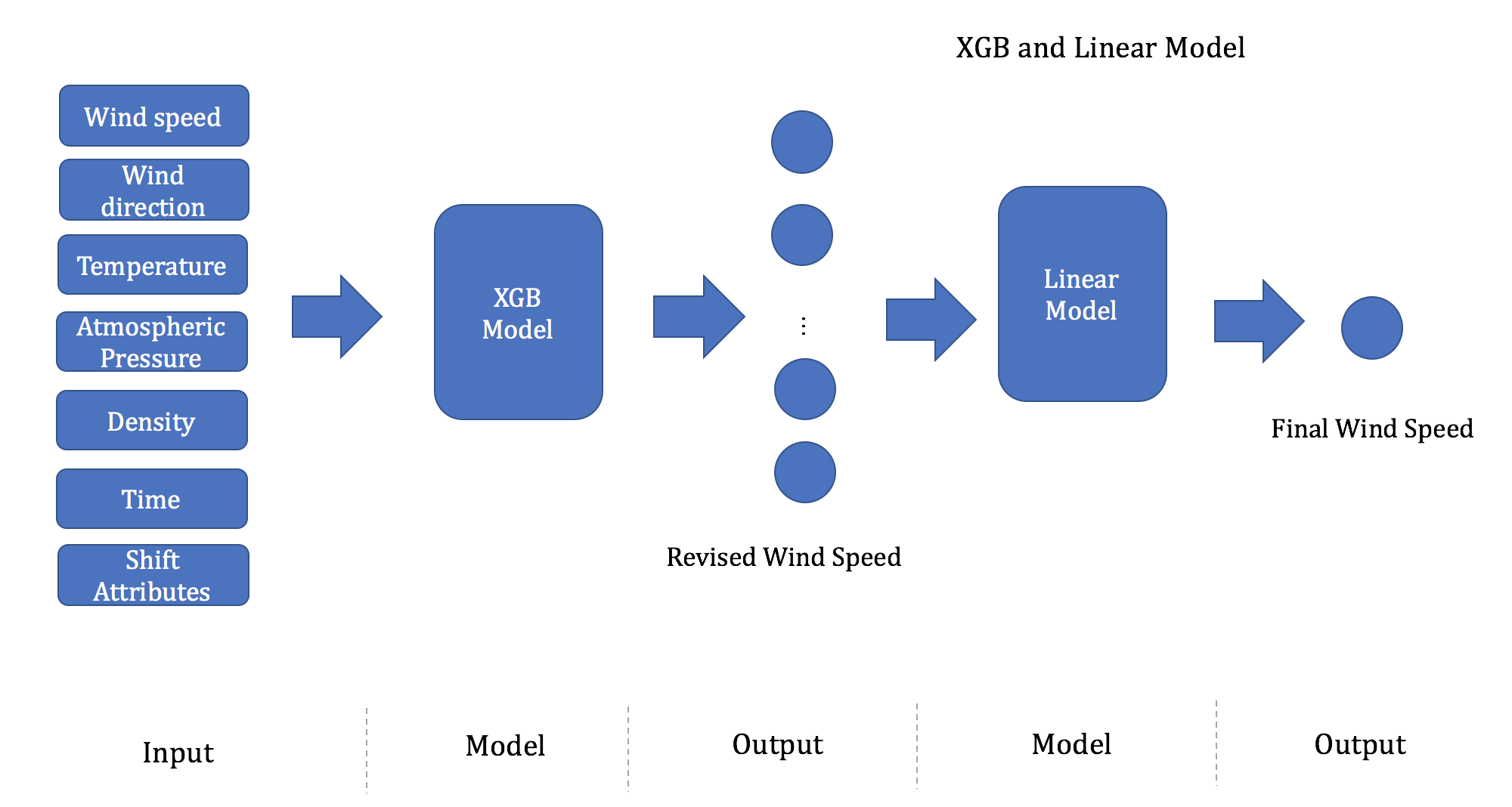


Figure 3. The Structure of XGB and Linear Model

The results of four linear models as Table 3.

Table 3. The Results of Four Linear Models as the Second Layer

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Models | Std | Std (3-15m/s) | RMSE | RMSE (3-15m/s) | Parameters |
| Linear | 1.763 | 1.458 | 1.869 | 1.909 |  |
| Ridge | 1.768 | 1.456 | 1.873 | 1.910 | a= 1000 |
| Lasso | 1.766 | 1.457 | 1.867 | 1.899 | a= 0.1 |
| ElasticNet | 1.812 | 1.456 | 1.859 | 1.794 | a=1.0ratio=0.7 |

* 1. **Add Non-Linear Models as the Second Layer**

For the Non-linear models, we choose three models to test, including SVR, Random Forest (RF) and XGB model. the Structure of XGB and non-linear model is similar with last one as Figure 4.

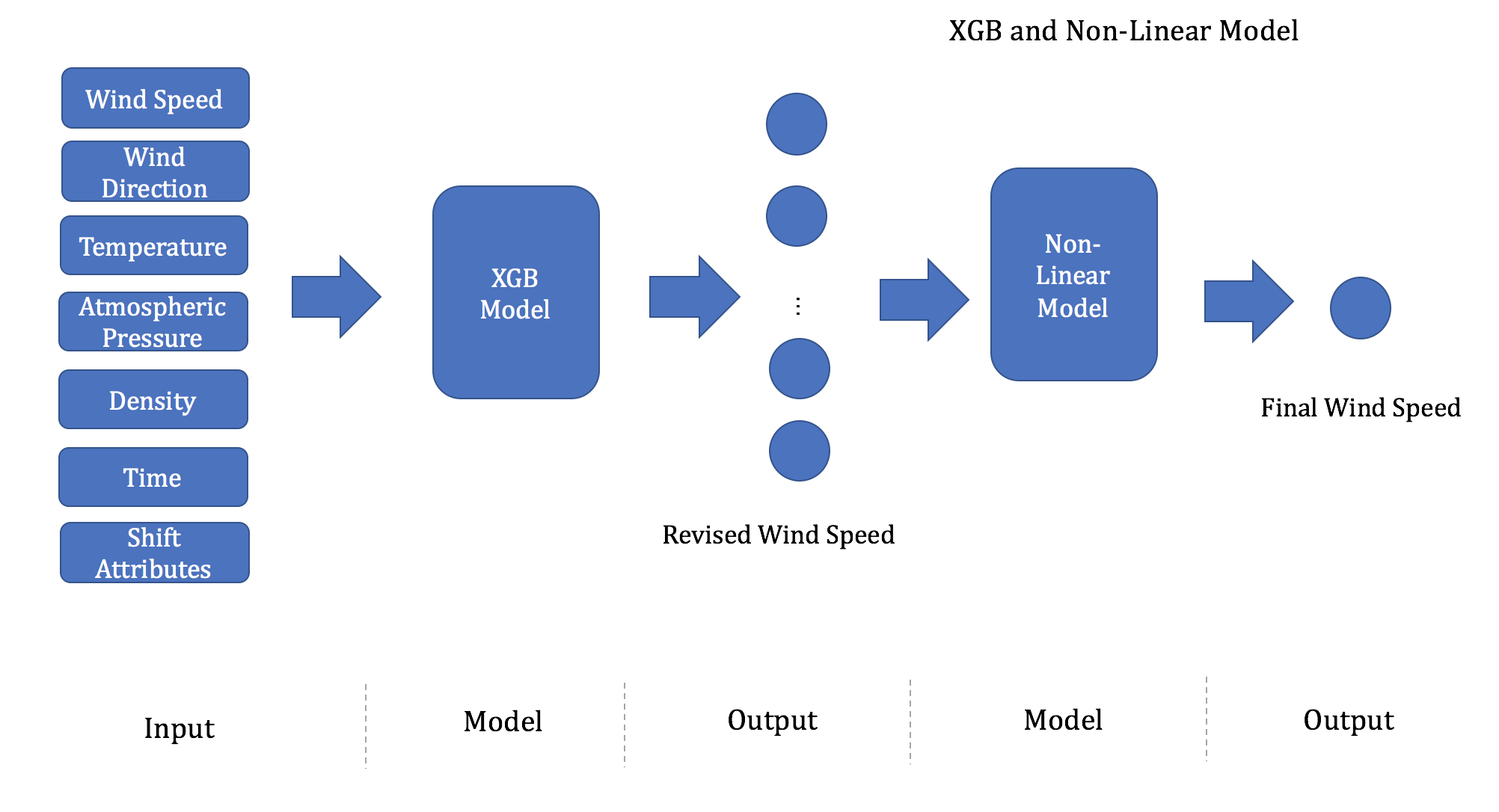


Figure 4. The Structure of XGB and Non-Linear Model

The results of three non-linear models as Table 4.

Table 4. The Results of Three Non-Linear Models as the Second Layer

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Models | Std | Std (3-15m/s) | RMSE | RMSE (3-15m/s) | Parameters |
| SVR | 1.800 | 1.548 | 1.918 | 1.988 | epsilon=0.2 |
| Random Forest | 1.828 | 1.511 | 1.898 | 1.895 | n =200 |
| XGB | 1.809 | 1.526 | 1.917 | 1.963 | n=200 |

We could see from the Table 4 that the performance of non-linear models does not perform very well. We figure the reason is that for the results from the first layer are only 5-dimensional data, and the non-linear models are too complex for them. Hence, non-linear models do not outperform the linear models and we discard to use non-linear models at the second layer.

1. **Improving Processing**

**3.1 Build Different Models with Different Horizons**

Consider that different time may have different change of the wind speed, and hence we want to build different models for different horizons. The horizon is from 16 to 39, so we totally build 24 models for one day. The results of different linear models with horizons are as Table 5.

Table 5. The Results of Second Layer for Linear Model with Horizons

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Models | Std | Std (3-15m/s) | RMSE | RMSE (3-15m/s) | Parameters |
| Linear | 1.766 | 1.453 | 1.836 | 1.826 |  |
| Ridge | 1.780 | 1.453 | 1.871 | 1.885 | a= 20/100/200 |
| Lasso | 1.771 | 1.459 | 1.863 | 1.883 | a= 0.1 |
| ElasticNet | 1.823 | 1.456 | 1.865 | 1.872 | a=1.0ratio=0.7 |

We could compare the results with Table 3, the target RMSE (3-15m/s) reduces for all models except ElasticNet. Hence, we can get that horizon processing has some good effect on performance. However, when the size of data is not large, it is difficult to use models with horizons.

* 1. **Change the Sampling Range on Training Set**

The original sampling range is from 4 to 12, and now we change it into [3, 15]. Restate the results above, and please see the Figure 5.

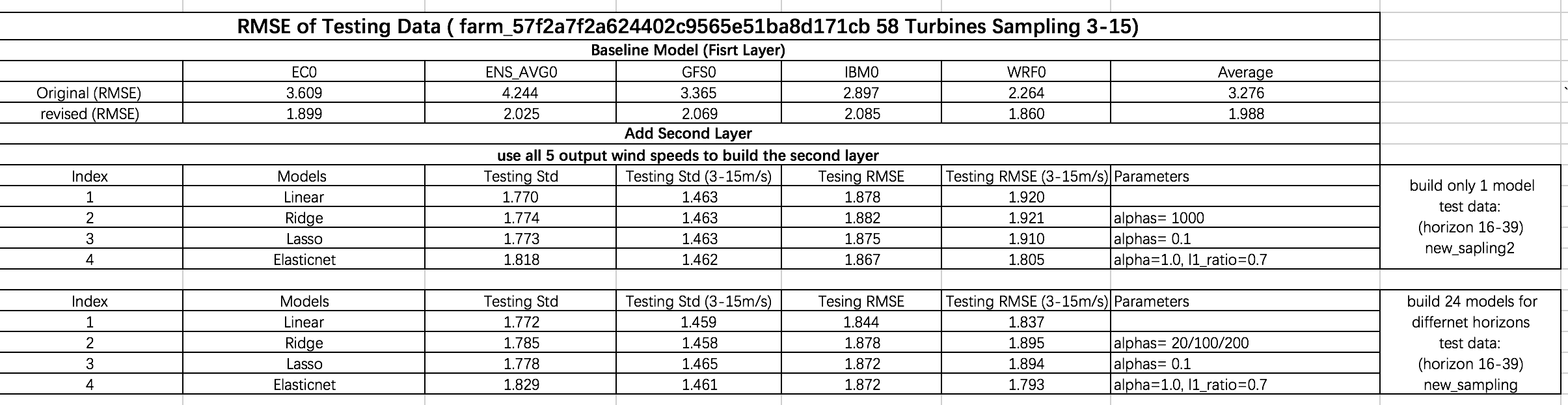
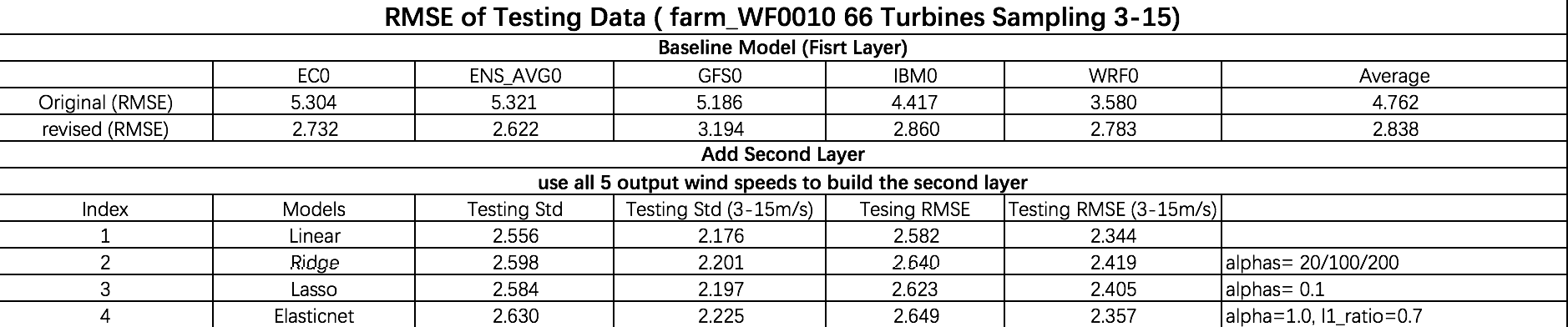


Figure 5. The Results of Second Layer for Linear Model with New Sampling Range

We could find that the sampling range does not have much effect on final results. Because we mainly focus on results of 3-15m/s wind speed, and hence, we change the sampling range to 3-15 for later experiment.

1. **Feature Engineering**

In this part, we want to do more feature engineering to mine more information from the data. We shift wind direction, temperature and wind speed from nearby 4 monitoring points. For the wind direction, we need to change the real values into sin/cos values specially.

In the experiment, we do one kind of feature engineering every time, and compare the results with original ones. The results of different feature engineering are as Figure 6.

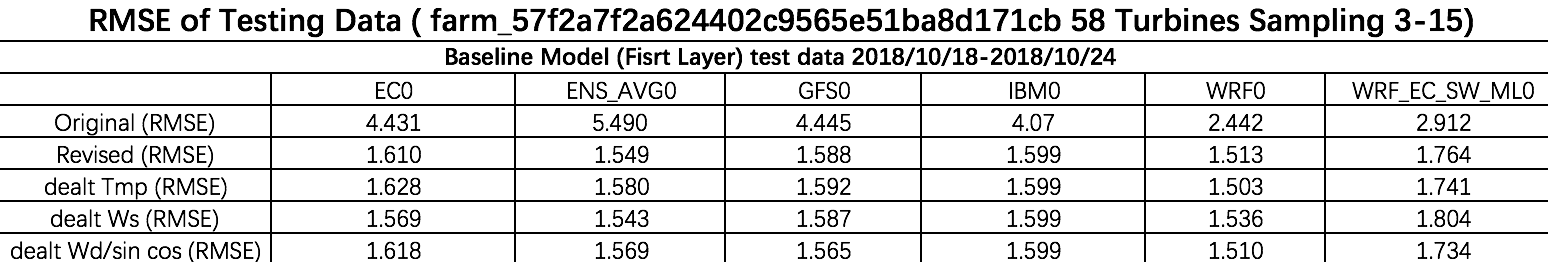


Figure 6. The Results of Different Feature Engineering

From the Figure 6, we could see that above all feature engineering does not improve performance significantly. We figure that it has been already very difficult to improve performance by means of feature engineering, but we could consider using more weather sources.

1. **Validate Models with 6 Weather Sources**

At last part, we find that it is difficult to attain better results with more feature engineering. Hence, we add a new weather source, and now we have 6 weather sources totally. The following results are based on 6 weather sources.

* 1. **Validation on 4 Linear Models as the Second Layer**

We use new data (add WRF\_EC\_SW\_ML0 weather source) to retest the performance of models. The difference here is that we only use one-week data as the test data. The results are as Figure 7.

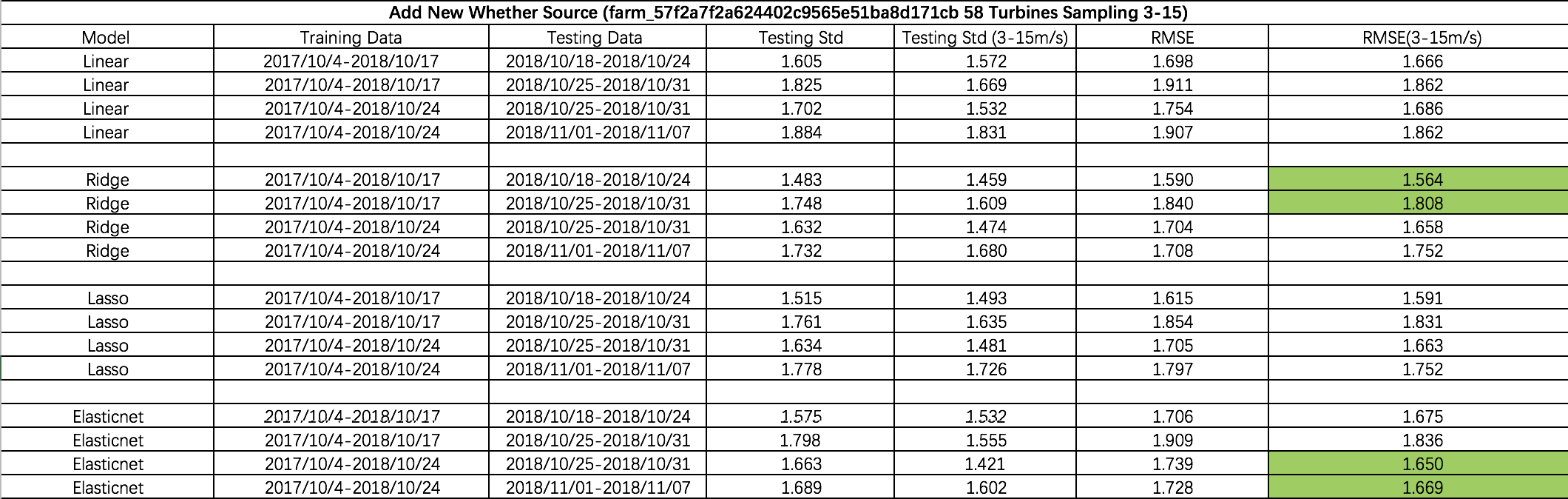


Figure 7. The results of new test data

From Figure 7, we could see that linear models (Bridge and ElasticNet) have better performance among 4 linear models. Hence, we choose these two linear models as the second layer.

**5.2 Add Fully-Connected Layer as the Second Layer**

After the first layer XGB, we could get multiple revised wind speeds, and want to combine them to get an improved result. Here, we use simplest neural network – fully connected layer to combine 6 inputs from the last layer. The structure of the second layer is as Figure 8. There are 6 inputs at the input layer. For hidden layers, we design 2 layers with different units. Finally, at the output layer, we output 1 value as the combined wind speed.

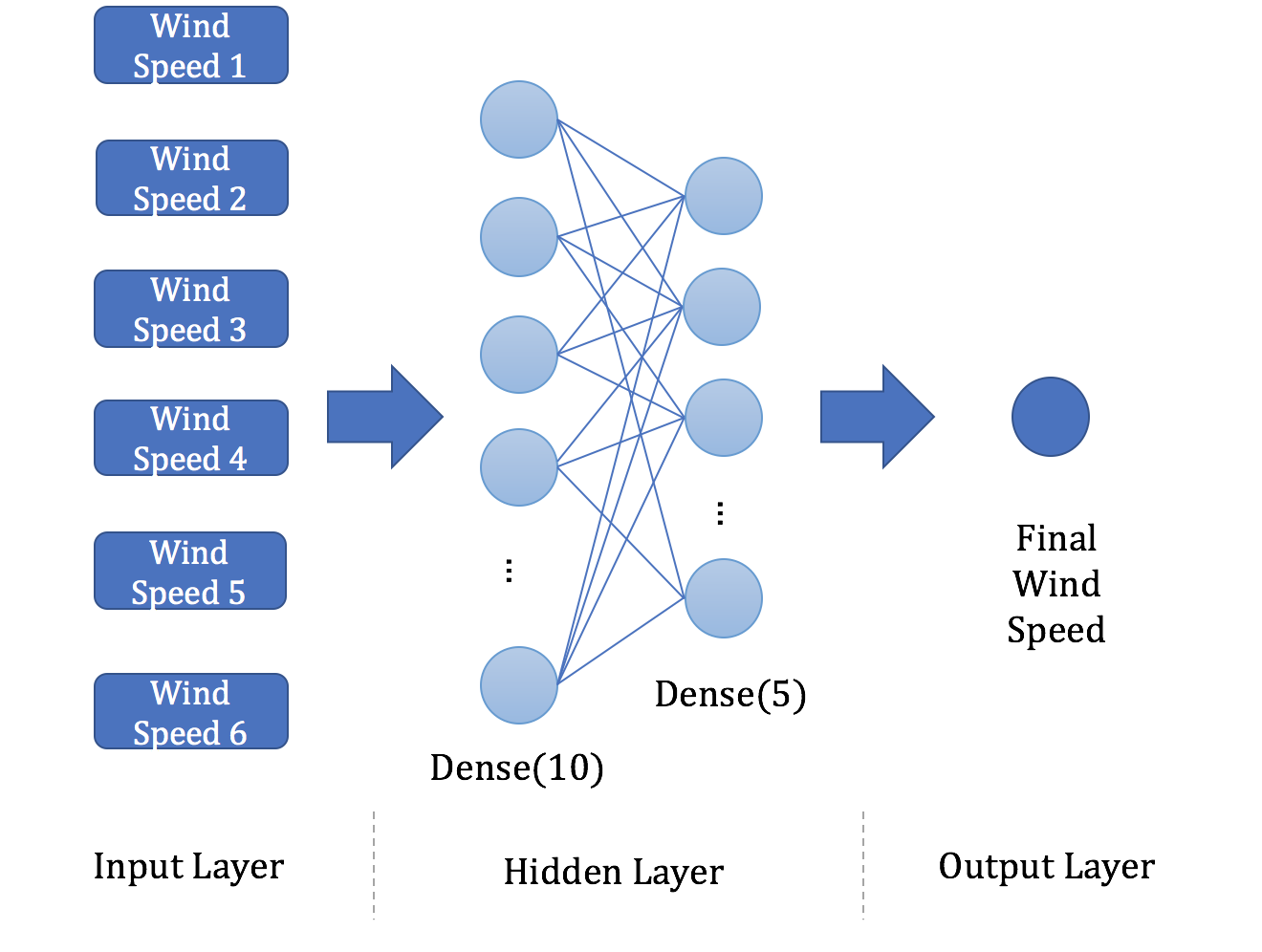
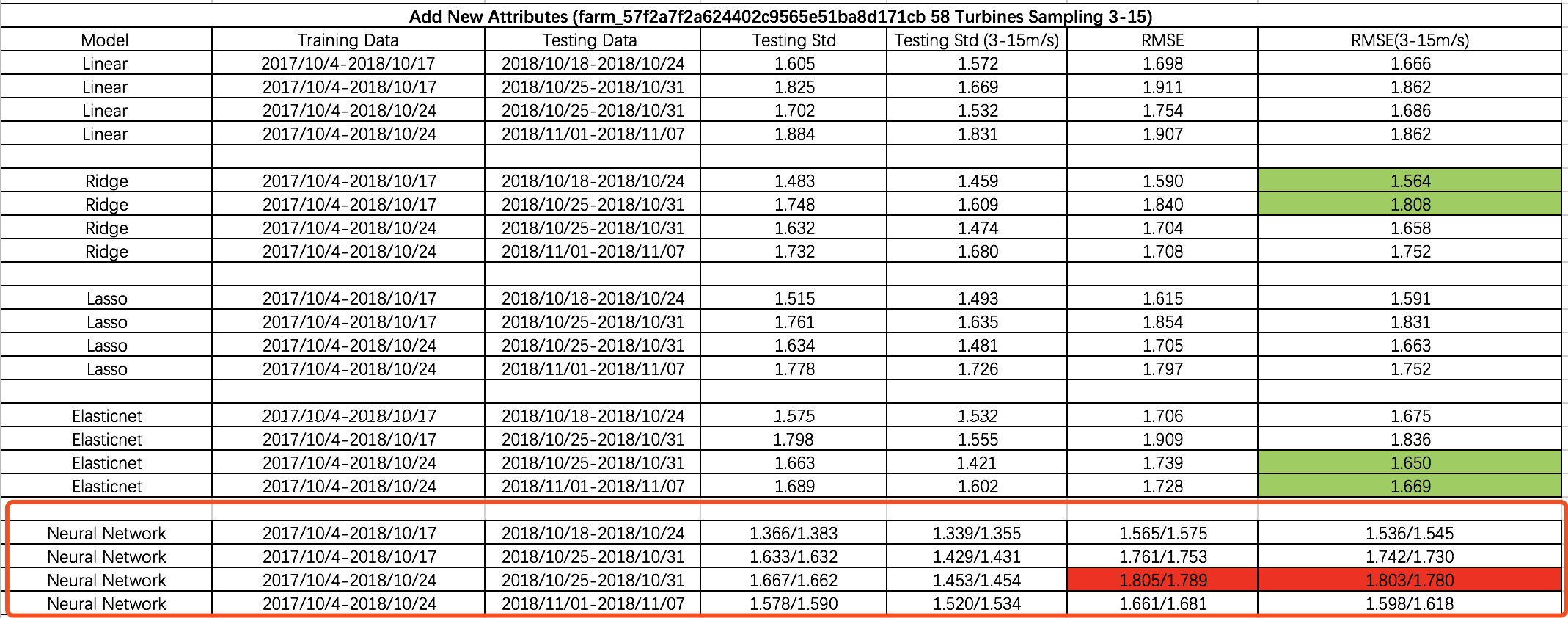


Figure 8. The Structure of Neural Network as Second Layer

Table 6. The testing results of neural network model

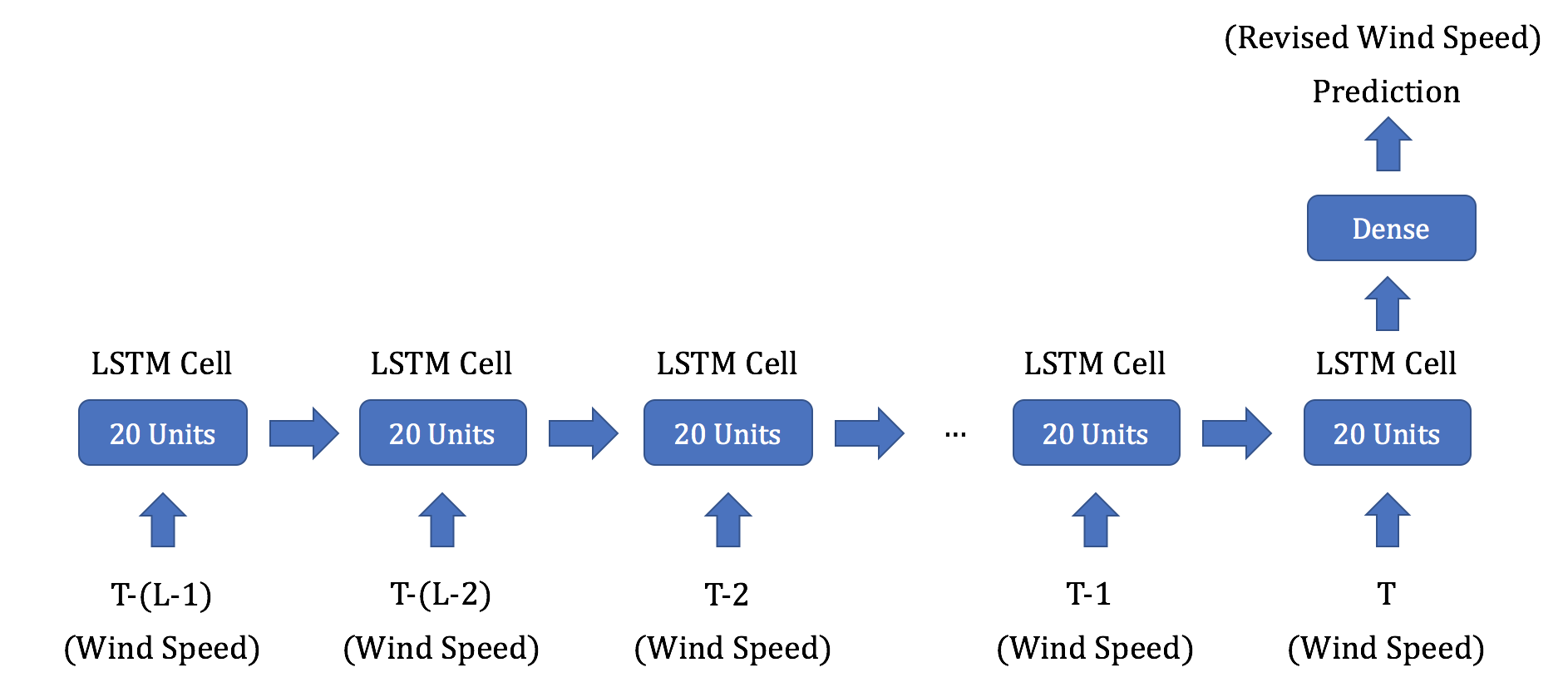


From the Table 6, we could see that the testing results of neural network are significantly better than other linear models as the second layer. Note that the right one in the neural network is with no delta. Hence, we want to change to use NN model to combine the wind speeds from 6 weather sources.

1. **Improvement on Model**

In this part, we consider using Recurrent Neural Network (RNN) to capture sequence information among data. We use LSTM at the first layer and test the performance of model.

There are two structures of the input. The first one is to just use wind speed and revise each one with former information. And the second one is that we use basic information of each weather source to revise wind speed at each time. The structure of models is as Figure 9.



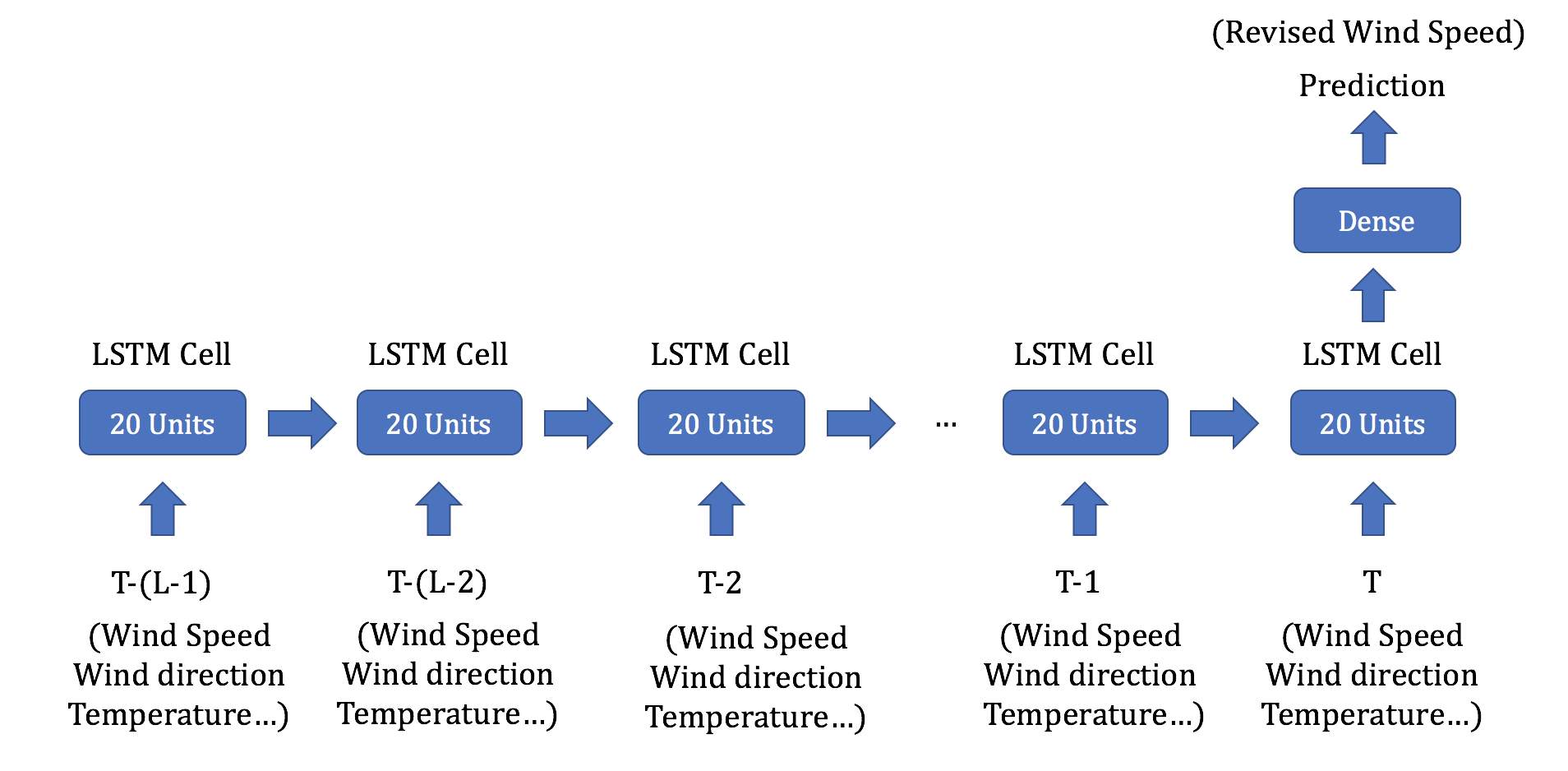


Figure 9. The Structure of RNN with Different Inputs

We could see that the difference of two structures is we use different inputs. After experiment, we find that the whatever the test data are one-year or just two-month, the first one (only use wind speed) is better than the second one significantly. The results are as Figure 10.

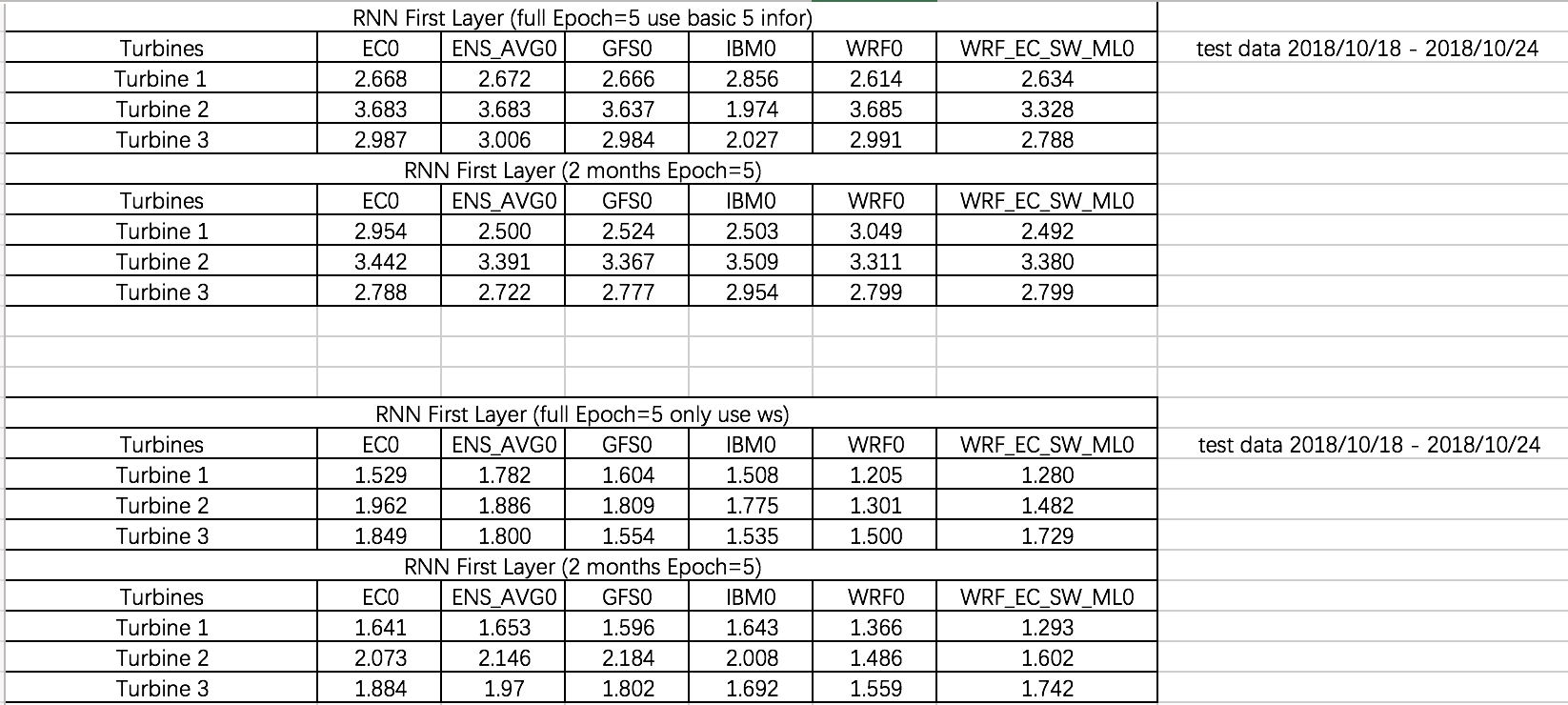


Figure 10. The Results of RNN with Two Different Inputs

Now, we use the first structure of RNN (only use wind speed) as the first layer and the linear model as the second layer. Train models on all data of farm ID 57f2a. The specific input attributes are shown in appendix A.2. The results of testing are as Figure 11.

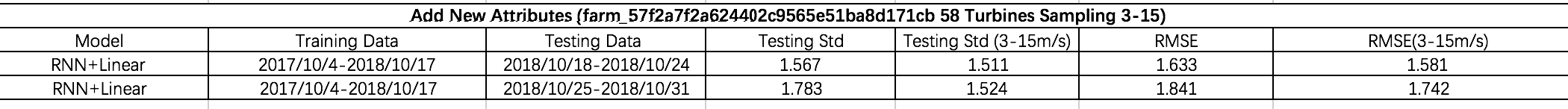


Figure11. The Results of RNN and Linear Model

We could compare results with the Table 6. We could find the result is not bad, but there is not improvement. We figure that here we do not use any other information except wind speed. However, the time of training is very large, and for example, for 58 turbines data, it would spend about 11 hours to train with normal CPU.

1. **Conclusion and Future Work** 
   1. **Conclusion**

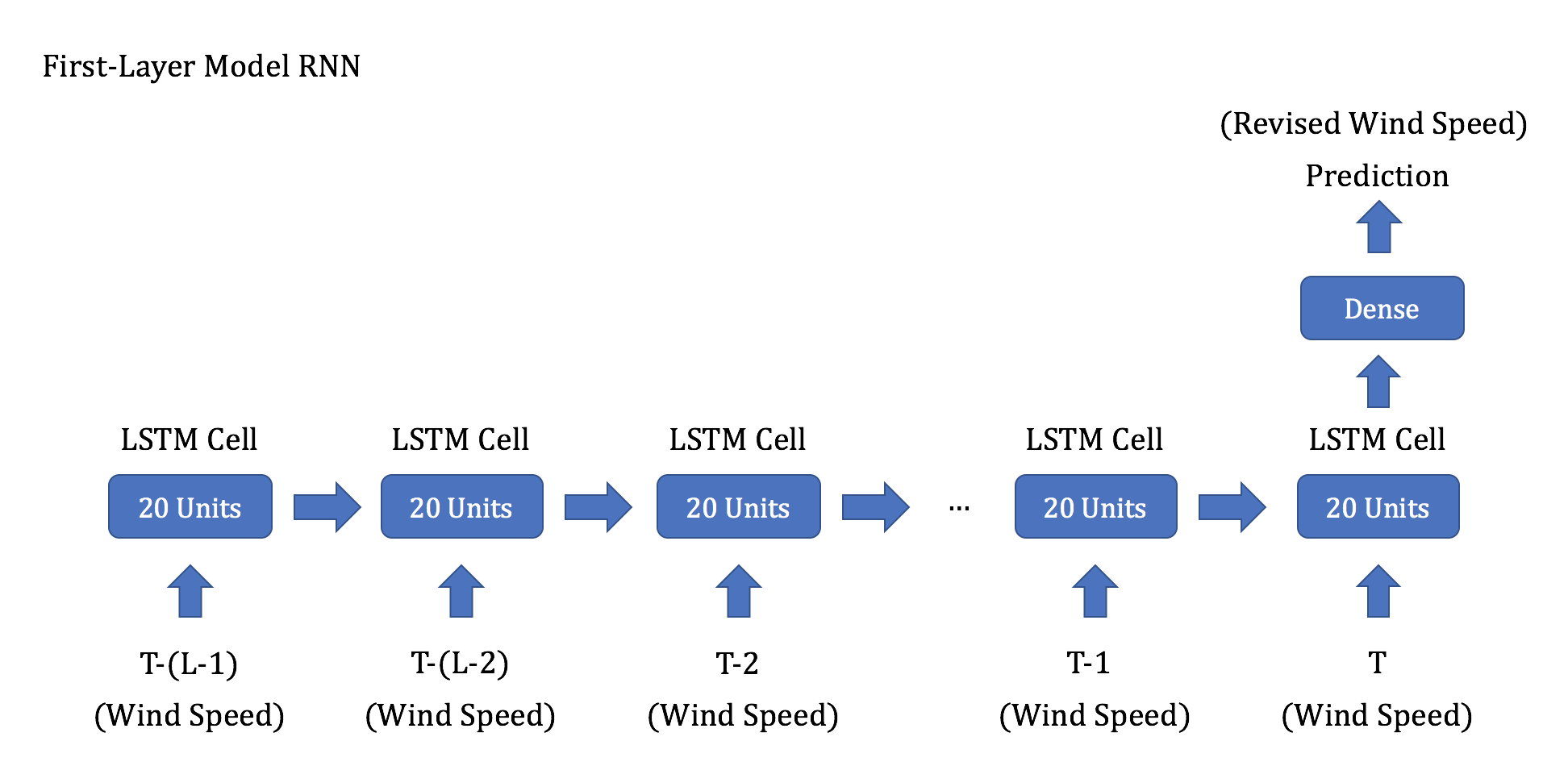
In this project, we firstly use XGB model to revise the wind speed of each weather source. After that, we consider combining these to get a better result. There are many different trials, and we could get conclusions as following: 1) For the second layer, the linear model is relatively better than non-linear models, such as SVR and RF. The reason is that there are not complex data after the first layer. The simple linear models, like ridge regression, could have a great performance. Hence, we simply use linear models to combine the output from the first layer. 2) We try to do more feature engineering, for example, shifting temperature from nearby points. However, the results do not satisfy us and we do not get much improvement. Luckily, we find adding new weather source could get an improvement. 3) After experiment, we could see that Bridge Regression and ElasticNet have a better performance than other linear models. And we choose these two models temporarily. In addition, simple neural network – fully connected also has a great performance. 4) Apply sequence models, like RNN, to capture more sequence information. We apply LSTM at first layer, and not use XGB model. We could get similar results as Bridge and ElasticNet. However, the time consumption of RNN is nearly 10 times of linear models. Hence, trade-off performance and time, we prefer to use simple linear models.

* 1. **Future Work**

In the last experiment, we have tried to use LSTM model to capture sequence information, and revise the wind speed at each time. The disadvantage is that we do not use any other information except wind speed. In this part, we propose two models that may have a better result.

* + 1. **RNN and Fully-Connected NN**

From the Table 6, we could see that Fully-Connected as the second layer is a little better than traditional linear models. Hence, we could try to use NN on both layers. The structure of this model is as Figure 12. We keep LSTM models to revise wind speed of each weather source, and use fully-connected neural network to combine the outputs from the first layer.



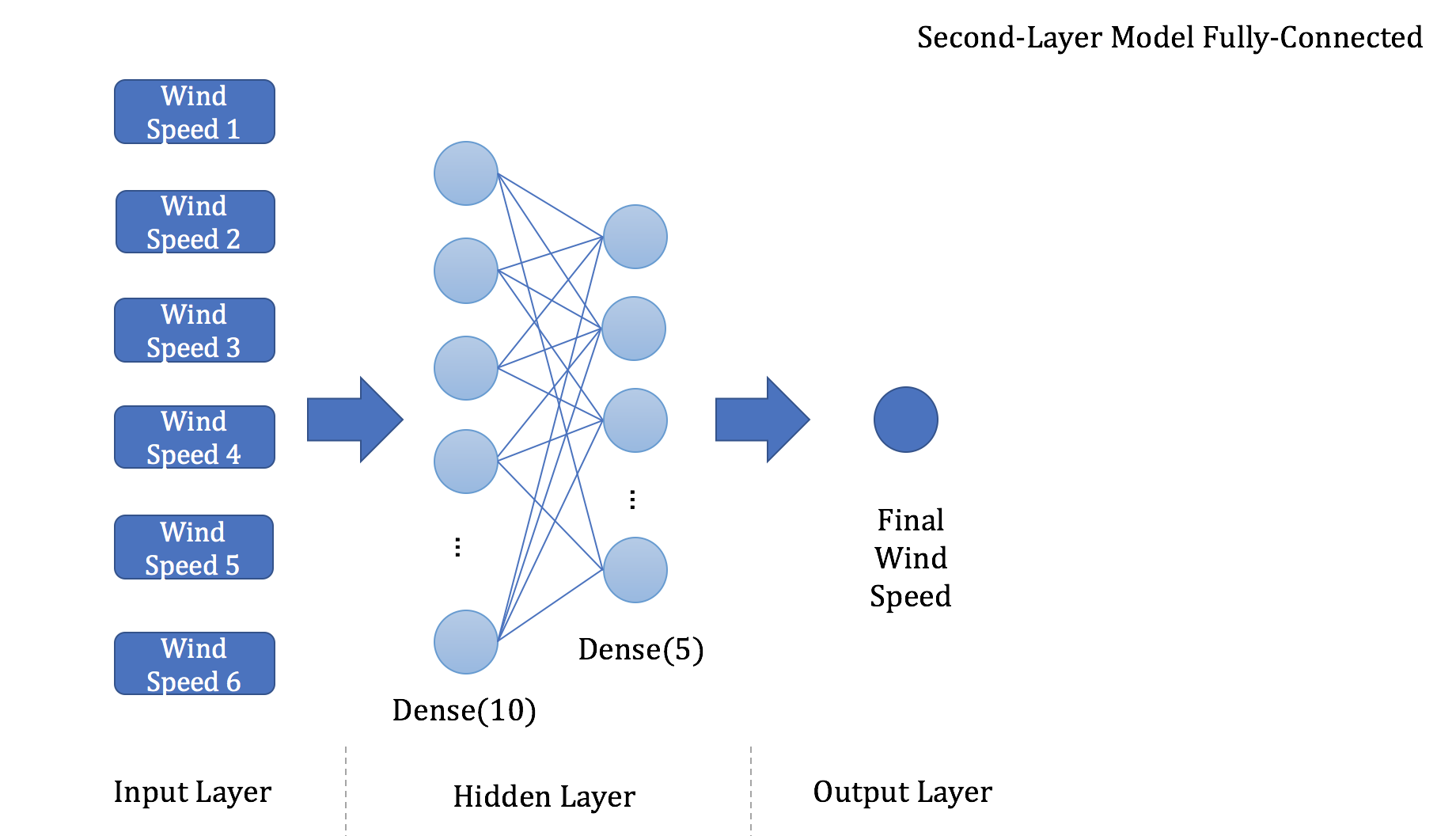


Figure 12. Apply NN Models on Both Layers

* + 1. **RNN XGB and Fully-Connected NN**

We could keep LSTM as the first layer, and use other information on the second layer XGB. Finally, we still remain to use fully-connected layer to combine the wind speed. The integrated structure of model is as Figure 13.

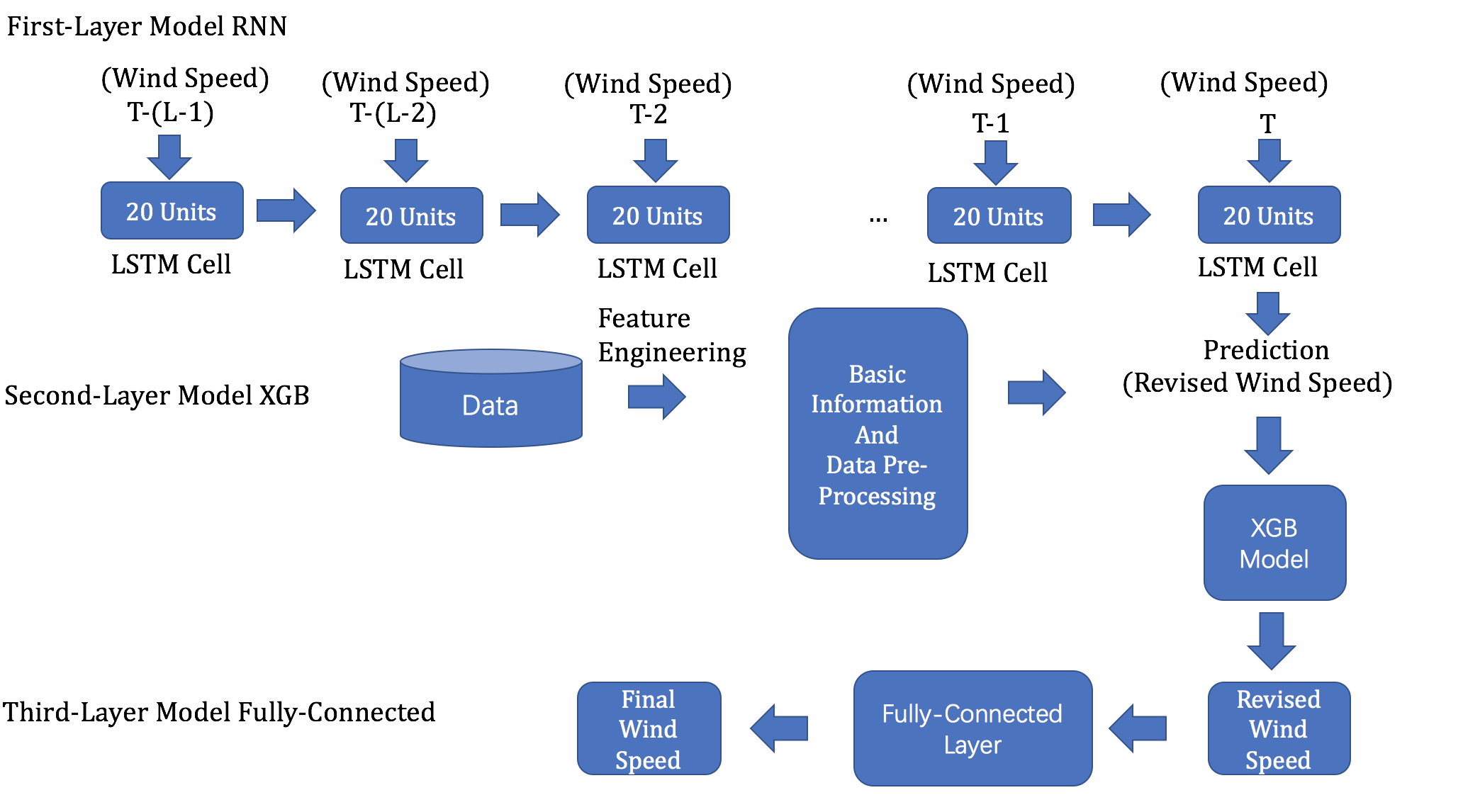


Figure 13. Three-layer model

**Appendix**

**A.1 Specific Inputs Attributes List with 5 Weather Sources**

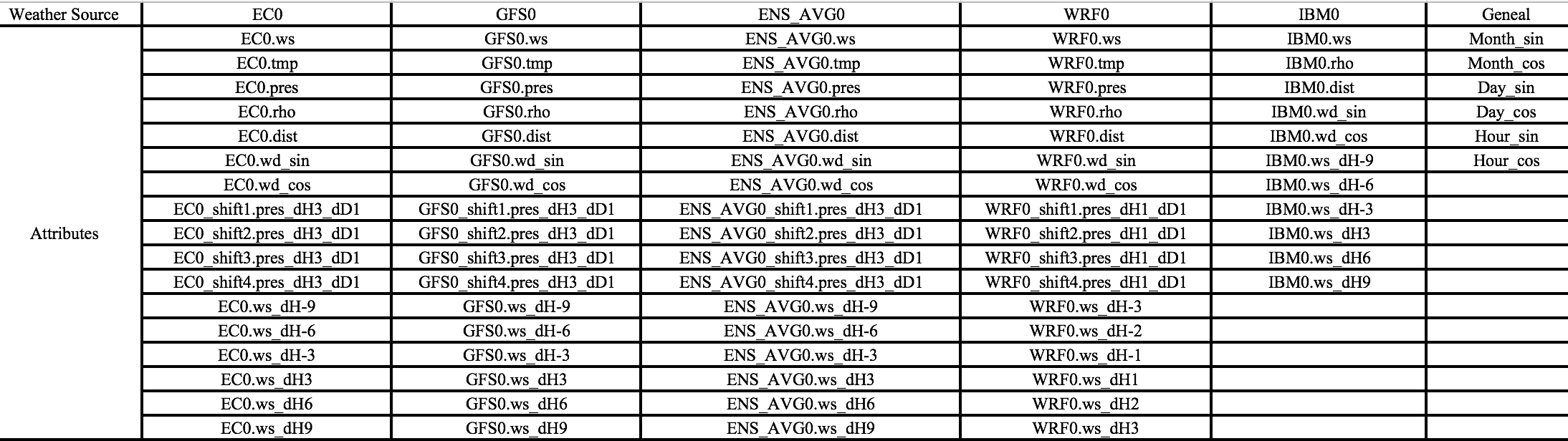


Figure 14. List of Inputs Attributes with 5 Weather Sources

**A.2 Specific Inputs Attributes List with 6 Weather Sources**

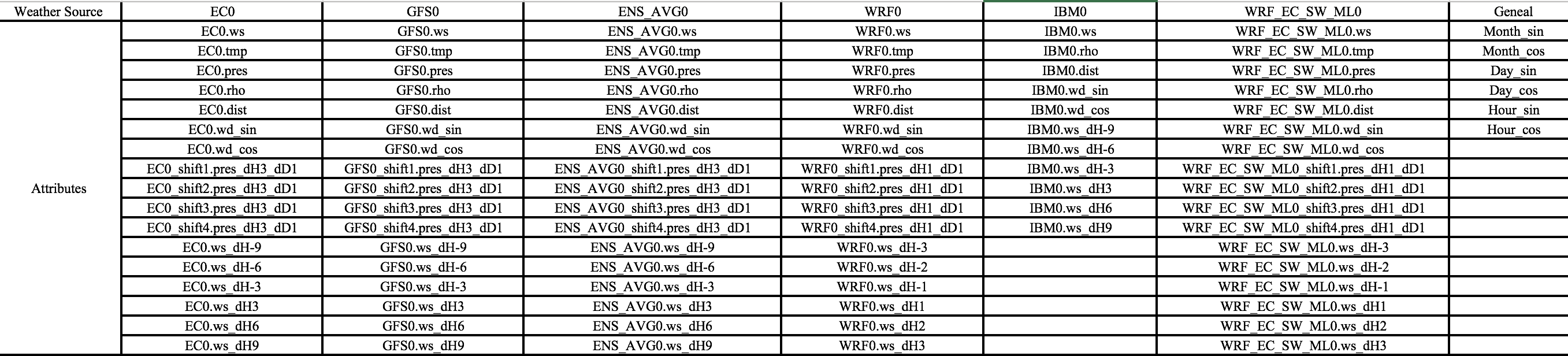


Figure 15. List of Inputs Attributes with 6 Weather Sources