Cement strength prediction using cloud-based machine learning techniques

Article *in* Journal of Structural Integrity and Maintenance · September 2020

DOI: 10.1080/24705314.2020.1783122

Vishal Naranje
Amity University Dubai

Sachin Salunkhe
Vel Tech - Technical University

Journal of Structural Integrity and Maintenance

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/tstr20

Cement strength prediction using cloud-based machine learning techniques

Nand Kumar , Vishal Naranje & Sachin Salunkhe

To cite this article: Nand Kumar , Vishal Naranje & Sachin Salunkhe (2020) Cement strength prediction using cloud-based machine learning techniques, Journal of Structural Integrity and Maintenance, 5:4, 244-251, DOI: 10.1080/24705314.2020.1783122

To link to this article: https://doi.org/10.1080/24705314.2020.1783122

Published online: 17 Sep 2020.

JOURNAL OF STRUCTURAL INTEGRITY AND MAINTENANCE
2020, VOL. 5, NO. 4, 244–251
https://doi.org/10.1080/24705314.2020.1783122

Cement strength prediction using cloud-based machine learning techniques

Nand Kumar[a], Vishal Naranje[b] and Sachin Salunkhe[c]

[a]Department of Computer Science, Birla Institute of Technology and Science, Dubai, UAE; [b]Department of Mechanical Engineering, Amity University, Dubai, UAE; [c]Department of Mechanical Engineering, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai, India

ABSTRACT
This paper describes a cloud-based software framework to predict cement strength for 2 days, 7 days and 28 days. Levenbarg-Marquardt back-propagation-artificial neural network (LMBP-ANN) is used to build a prediction model. This ANN model uses 70% of data for training (70%, 212 data records), testing (15%, 46 data records) and for validation (15%, 46 data records). A total of 16 significant input parameters are considered for the cement strength prediction. The user interface and software framework are built using the Python programming language. Multiple Python packages are used for the implementation of the ANN model. The cloud server having Ubuntu operating system has been used to host the web application for prediction of cement strength. The software application is tested using real-time data from various cement industries. The prediction of the cement strength of the proposed ANN-based software application appears to be very similar to those currently generated in experimental data in the cement manufacturing industry. The adequacy of the developed model based on the back-propagation ANN algorithm is confirmed as the Pearson correlation of experimental value and predicted value. The calculated value of R for experimentations on the data is 0.82539 and R^2 is 0.6813.

KEYWORDS
Cloud-based software; Levenbarg-Marquardt back-propagation-artificial neural network (LMBP-ANN); Python packages

Introduction

Early cement strength estimation is very useful for reducing running costs and ensuring safety. In addition, the calculation of early age cement strength makes it possible to assess building efficiency and potential longevity (Poole & Harrington, 1996). However, owing to the vast number of input process parameters involved in the manufacture of cement, it is very difficult to assess the impact of each process parameters on the final consistency of the cement. In the cement industry, the majority of rapid strength evaluation methods are used to verify the cement quality for 7 days and 28 days (Anand et al., 2018). Traditionally this function is performed manually by a specialist of highly experienced cement quality professionals. But forecasting consistency requires not only experience in cement strength measurement, but also in-depth knowledge of the cement manufacturing cycle and the effect of input parameters on cement performance. Quality of cement depends to the large extent on the proportion of percentage of the input raw material used, hence checking and predicting the cement strength is complex, time-consuming, trial and error process.

Recently, various researchers using artificial intelligence techniques such as artificial neural networks (ANNs), fuzzy logic and genetic algorithm for prediction of the cement comprehensive strength (Abd Elaty, 2014, Kosmatka et al., 2011; Kumar & Naranje, 2019). The benefits of using these AI tools are the ease of application, robustness, etc. The aim of the proposed work is to automate the prediction of cement strength using suitable tools of artificial intelligence (AI). In this paper, a software application is developed to automate the prediction of cement strength.

Literature review

During the last few decades, numerous attempts have been made by the researchers using artificial intelligence techniques to propose various prediction models for different manufacturing applications (Naranje et al., 2016; Naranje and Salunkhe, 2020; Salunkhe et al., 2019). Few researchers also used the ANN to predict the cement concrete strength and found it more accurate compared to other existing techniques. The significant models mentioned in the literature for cement concrete strength prediction have been described below.

Kewalramani and Gupta (2006) suggested a weight and UPV rapid test method to estimate the long-term compressive strength of concrete. Multiple regression analysis and artificial neural networks were used. They concluded that ANN is more efficient in predicting long-term compressive strength of concrete compared to multiple regression analysis.

Sobhani et al. (2010) used regression, neural networks (NNT) and ANFIS models to predict the 28-day compressive strength of non-slump concrete (28-CSNSC), suggesting that the NNT and ANFIS models are more feasible to predict 28-CSNSC than the proposed conventional regression models.

Yuan et al. (2014) have considered that the cement concrete quality gets affected by structured and unstructured factors. Based on these structured and unstructured factors, they have proposed two hybrid models considering the genetic algorithm and the adaptive network-based fuzzy inference system (ANFIS) to predict the cement concrete strength.

Bingöl et al. (2013) predicted the compressive strength of lightweight and Semi lightweight concretes with pumice

CONTACT Nand Kumar nandkumar@dubai.bits-pilani.ac.in

aggregate subjected to high temperatures using Artificial Neural Network (ANN).

Few other researchers have put efforts using ANN techniques based on non-destructive test results in predicting the concrete strength and concluded that the concrete strength fluctuates with the applicant needs and its effects vary with input ingredients (Chandwani et al., 2014; Chopra et al., 2016; Hendi et al., 2018; Khashman & Akpinar, 2017; Oh & Oh, 2016; Rafi & Nasir, 2016). Behnood et al. (2017) have shown the cement concrete strength development through a mathematical model. Furthermore, Khademi et al. (2017) have explored the different design mix for the cement preparation inputs and used 173 different mix design inputs. Three different models have been presented based on three different technological aspects such as multiple linear regression model (MLR), ANN and ANFIS for predicting the 28-day compressive strength of concrete.

It is observed from the literature that ANN has been preferred by many researchers for prediction approaches. But the foregoing literature review reveals that none of the research efforts have been applied to develop a cloud-based ANN model to predict the strength of cement for 2 days, 7 days and 28 days. Mostly all the ANN-based models were developed to check the compressive strength of concrete and these models are limited to four to five input parameters and used for offline data set analysis. As the ANN is an iterative process to reduce the error between the original and predicted value. However, the iterative working and model preparation process of ANN is very complex, it needs experts to handle it. Also, the computing resources required are huge for ANN-based prediction models. Therefore, researchers have carried out various analyses on high-end machines. To address these problems, the literature has been explored and found that the recent technological enhancements in computing and networks bring cloud-based application as a feasible solution. This kind of solution can significantly reduce the requirements of computing resources at the user's end and also bring the mobility to access the prediction software application at any time, from anywhere and by anyone. Therefore, in this study, a hybrid approach using cloud computing and ANN has been proposed for cement strength prediction, which does not need any expert or high-end machine to handle it. The proposed software-based application approach produces efficient and effective cement strength prediction results.

In the section "Overview of proposed approach", a brief description of the proposed approach is presented. The ANN model along with experimental result discussions are presented in Sections "ANN training, testing and software development" and "Validation of the proposed system", respectively. Finally, in Section "Conclusion", the conclusions are drawn.

Overview of proposed approach

The proposed research work is based on cement strength properties are mostly influenced by the dynamics of the input variables generally caused by the operating conditions. The proposed system is shown in Figure 1. It is composed of

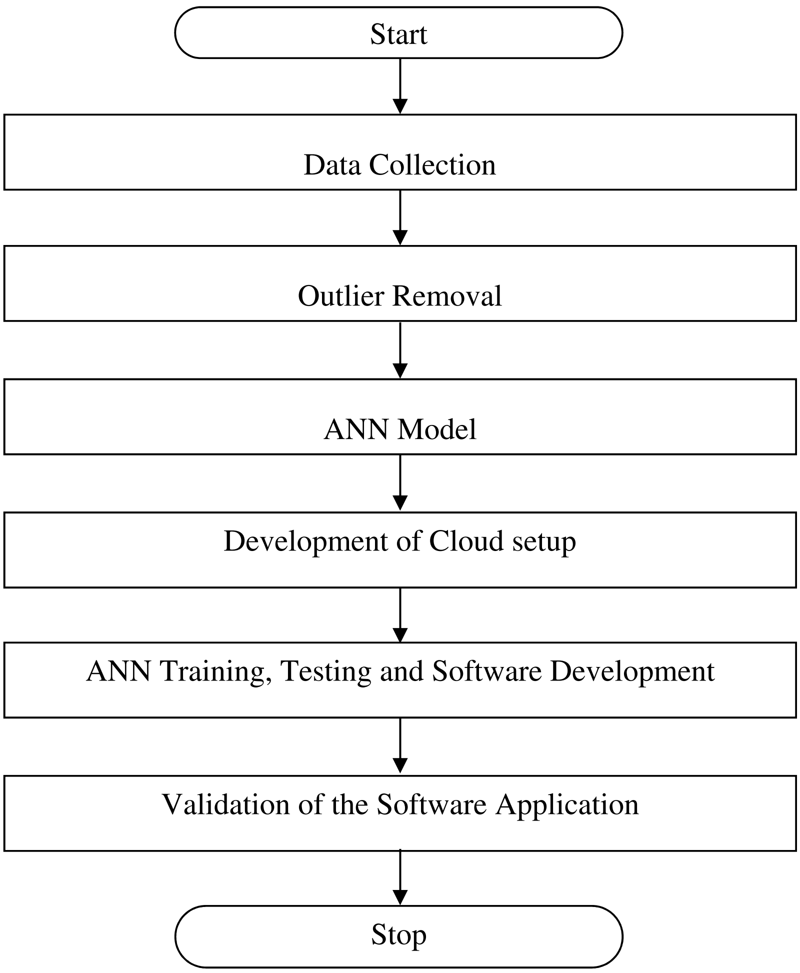

Figure 1. Framework of the proposed system.

multiple modules such as data collection, outlier removal, formation of the prediction model, regression using linear, non-linear and LMBP-ANN techniques, mean square error computation and comparison of different models, etc. The brief description of the proposed system is as below.

Data collection and pre-processing

In this study, the cement manufacturing industry experts have been consulted to understand the process of cement preparation, the process of finding cement strength, the parameters that affect the cement strength, etc. After a discussion with the cement manufacturing experts, it is found that there are a total of 16 significant parameters that affect the cement strength. These decided input parameters are blaine, temperature, residue (at 90 or 45 Âµ), water spray, moisture, LOI, SO3, C3 S, C2 S, C3A, LSF, NC, Clinker liter weight, Clinker Feed temperature, GA dosage. These input parameters are used for prediction of 2-, 7- and 28-day strength. Such statistics were gathered over the past 12 months from the two cement plants. Table 1 shows the sample data of cement plant 1.

The recorded data (304 records, excluding weekly and public holidays) may have blank or irrelevant data; therefore, pre-processing is performed to reduce such possibility. The feature extraction from the recorded data is done through Principal component analysis (PCA) method. PCA reduces the dimensionality linearly using Singular Value Decomposition (SVD) of data and the lower dimensional space is projected. The lowered dimensional data help in reducing the training time required for neural network (NN). Sometimes, because of errors few records may appear outlier which caused the NN training difficult and may result in wrong results, therefore to streamline the data outlier removal has been performed. This

Table 1. Sample data of cement plant 1.

		S. No.										
		1	2	3	4	5	6	7	8	9	…	304
Blaine cm^2/gm (*X1*)		3236	3209	3236	3288	3249	3275	3301	3249	3314	…	3281
Residue 90 Âµ % (*X2*)		1.4	1.3	1.1	1	1.1	1	1	1.2	1	…	1.3
Residue 45 Âµ % (*X3*)		13.4	12.8	11.9	11.3	11.7	10.7	10.3	12.4	12	…	12.9
NC % (*X4*)		28.8	28.8	28.8	28.6	28.8	28.8	28.8	28.6	28.8	…	28.6
L0I % (*X5*)		2.52	2.36	2.32	2.68	2.37	2.36	2.45	2.48	2.76	…	2.67
SO3 % (*X6*)		2.72	2.61	2.55	2.64	2.62	2.57	2.62	2.67	2.42	…	2.46
C3 S % (*X7*)		58.46	59.51	59.68	58.47	58.63	59.37	58.42	59.06	59.38	…	60.75
C2 S % (*X8*)		12.46	11.55	11.42	12.63	12.22	11.95	12.78	11.92	11.77	…	10.53
C3A % (*X9*)		7.347	7.241	7.241	7.321	7.178	7.188	7.241	7.321	7.099	…	7.769
LSF % (*X10*)		95.77	96.19	96.26	95.68	95.84	95.99	95.59	96.03	96.05	…	96.87
Moisture % (*X11*)		0.19	0.15	0.13	0.14	0.16	0.11	0.14	0.11	0.12	…	0.12
Clinker Litre weight gm/L (*X12*)		1366	1337	1329	1319	1323	1303	1320	1308	1290	…	1370
Clinker Feed Temp (*X13*)		153	167	149	170	167	154	168	170	154	…	132
Water Spray l/h (*X14*)		1000	1580	2250	1530	1710	2040	1960	1920	1960	…	2500
Cement Temp (*X15*)		100	102	101	100	99	100	102	108	108	…	100
GA Dosage g/mt (*X16*)		496	496	496	496	496	496	496	496	496	…	450
Strength N/mm^3	2 days	23.56	23.63	23.82	23.41	23.62	24.26	23.94	23.51	23.54	…	24.45
	7 days	38.75	38.38	38.94	38.45	38.85	39.12	38.36	39.2	38.87	…	40.95
	28 days	51.33	51.53	52.18	52.01	51.82	52.97	51.54	52.6	52.13	…	53.88

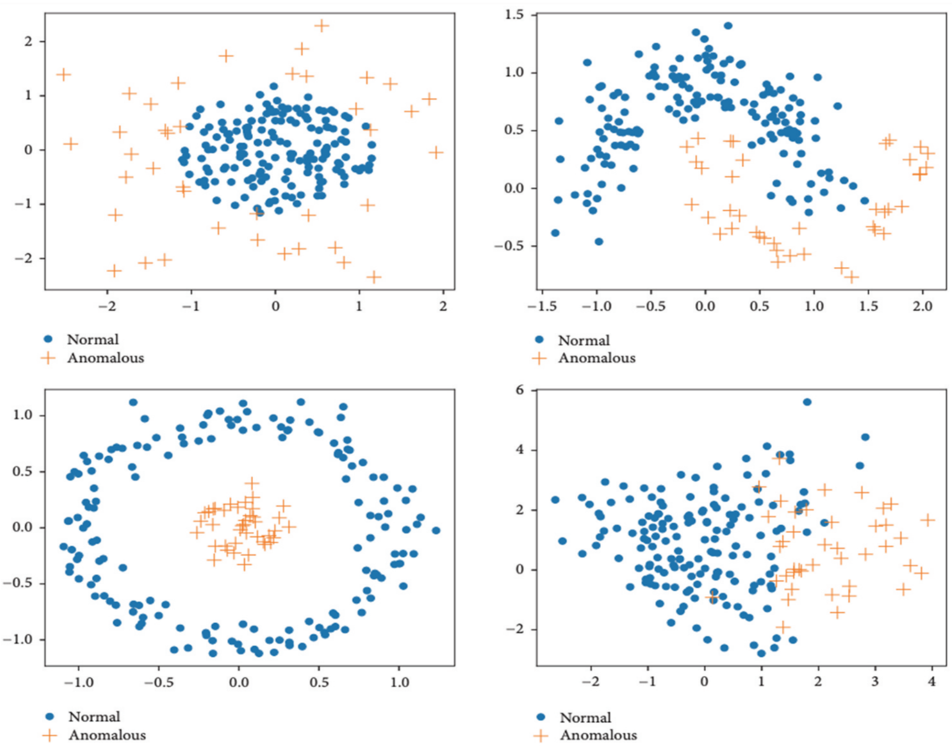

Figure 2. Gaussian, Moons, Circles and Classification distribution modes of data (Liu et al., 2018).

process of removing outlier data points improves the data quality and NN training. In this paper, the python programming language tool's scikit-learn package has been used to work with neural network. The data creation synthetically in different distribution modes (shown in Figure 2; Liu et al., 2018) is possible using scikit-learn package of the python programming tool.

Outlier removal

Recently researchers have constructed prediction models based on ANN (Abd Elaty, 2014; Chopra et al., 2016; Hendi et al., 2018). However, the outliers and overfitting in the input data are the main issue in such models. The overfitting data are the noisy data present in the input set which are too away from other data. The learning and training of NN are slowed by these overfitting data. Hence, the efficiency and accuracy of forecasting results may not be trusted. In literature, dropouts have been used to deal with such problems (Liu et al., 2018; Srivastava et al., 2014). Dropout prevents overfitting on the input set. It temporarily drops or removes out the noisy data in a NN, along with all its incoming and outgoing connections. Therefore, in this paper, a dropout-based ANN is used for cement strength prediction. It improves the learning of the network and subsequently results in a faster prediction process.

Development of Levenbarg-Marquardt back-propagation ANN

Many of the researchers are found that Levenberg–Marquardt back-propagation performs better for the nonlinear error function. The nonlinear error function E can be written as follows (Mejdoul et al., 2013):

$$E = \frac{1}{p}\sum_{p}\sum_{k} w_{pk} g_{pk}\left(d_{pk} - o_{pk}\right)^2 \tag{1}$$

where the training vector index is represented by p, k is the output vector elements, the pth desired pattern vector's kth element is d_{pk}, and when the training vector index pattern p is presented as input to the network the kth element of the output vector is represented by o_{pk}. The objective is to minimize the error function E, which can be achieved by updating rule to adjust the weights of the edges connecting neurons of different layers of NN. Finally, the testing and validation are completed using Pearson's correlation coefficient (r). It can be represented as (Mejdoul et al., 2013):

$$r = \frac{\sum_{i=1}^{N}(E_i - E_m)\cdot(F_i - F_m)}{\sqrt{\sum_{i=1}^{N}(E_i - E_m)^2 \cdot \sum_{i=1}^{N}(F_i - F_m)^2}} \tag{2}$$

where the total number of data sets is denoted by N, the experimental data set is denoted by E_i, the forecasted data set is denoted by F_i, the experimental mean value is denoted by E_m, and the forecasted mean value is denoted by F_m (Mejdoul et al., 2013). Further, to prevent the overfitting of the data the λ (lambda) regularization is used. This is a regularization parameter and denotes the degree of regularization. The Eq. (1) can be rewritten using λ regularization as follows:

$$E_{reg} = \frac{1}{p}\sum_{p}\sum_{k} w_{pk} g_{pk}\left(d_{pk} - o_{pk}\right)^2 + \lambda\sum_{p}\left|w_{pk}\right|^p \tag{3}$$

The value of the λ ranges from 0 to 1, and depending upon it marches towards 0 or 1, the degree of regularization decreases or increases. (Hassan et al., 2013; Srivastava et al., 2014)

The cement strength is predicted using the LMBP-based ANN model. The proposed LMBP ANN model is a multi-layered architecture. Input, hidden and output layers are mainly the three layers. In this analysis, the logical relation between input and output is defined, 16 input parameters and one output parameters are used in a NN. A back-propagated multi-layered neural network algorithm is used to estimate the cement strength. The number of hidden layers affects the performance of the network. The illustrative diagram of NN used for emulating the cement strength prediction process is shown in Figure 3.

The relationship between input parameters and the predicted cement strength process is determined from the results obtained using LMBP-ANN. The degree of the prediction for the proposed model is calculated by the mean prediction error percentage(E). These calculations are carried out by averaging all the individual percentage prediction errors for the test data sets. The Mean prediction error percentage (E) is represented as

$$E = \frac{1}{p}\sum_{i=1}^{p}\left(\frac{|E_{cs} - P_{cs}|}{E_{cs}}\right) \times 100 \tag{4}$$

where E_{cs} is the experimental cement strength and P_{cs} is the predicted cement strength.

Details of Cloud server setup

A cloud server has been set up to carry out the analysis and prediction of cement strength. This cloud server is utilized to host the high-end software, web application and to bring the mobility in the proposed approach. The server hardware is a HP blade machine, which has a 2.8 GHz processor, 32 GB RAM. The server platform is an Ubuntu 16.04 Linux variant operating system (OS). The figure shows the Cloud Server setup:

Traditionally this function is performed manually by a specialist of highly experienced (Kumar & Mittal, 2012). But forecasting consistency requires not only experience in cement strength measurement, but also in-depth knowledge of the cement manufacturing cycle and the effect of input parameters on cement performance.

The cloud server having Ubuntu operating system has been used to host the Python programming language. Python programming language is one of the most popular and commonly used programming languages now a day. It is an interpreted, general purpose, open-sourced programming language. Python provides huge advantages to the software developers in terms of simplicity, ease of use, code readability, and the inbuilt libraries to reduce the programming overhead, ease of deployment, etc. In this proposed work, multiple packages of Python have been used to implement neural network. These packages need to be installed on the cloud server in order to use them as and when required. The following sub-sections describe some of the important such packages.

Tensor flow package

This package of Pythons helps in developing the neural network models, because of its computational ability known as

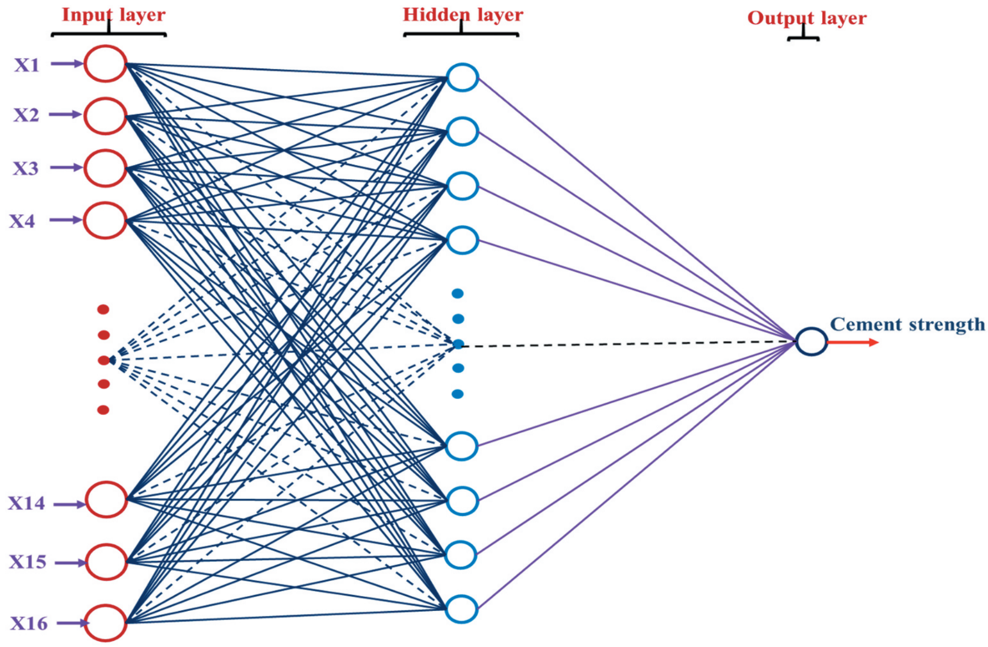

Figure 3. Multilayer architecture of back–propagation ANN used for cement strength prediction.

tensors. These tensors are multi-dimensional matrix which represents the input data. The flexibility, parallel neural network training and responsive construct make it the most important package of the python language. The Tensor Flow package internally uses NumPy package to mathematically understand and represent the input data.

Scikit-learn package

This package is used to work with huge complex data. Its feature of cross-validation, feature extraction and unsupervised learning algorithms has been used to develop a neural network model in this experimentation. This package brings adaptability in the approach by its ability of data mining and machining learning.

Pandas package

This package helps in data analysis. It is a machine learning library of Python. So many inbuilt libraries of this package for data grouping, filtering, time-series functionalities, etc., make it stand unique.

ANN training, testing and software development

The ANN prediction model is tested and validated on the real-life data collected from the cement manufacturing industry. This ANN architecture is used (70%, 212 data records), testing (15%, 46 data records) and validation (15%, 46 data). The neural network divides the total input data into training records. After the successful model testing, the Python software application is developed which includes the development of algorithm and development of front-end user interface. The option of uploading the input data into the system is given in the user interface. Afterward, the robustness of the developed software is tested for different input data sets. The details of the analysis are discussed in the following sections.

Validation of the proposed system

The developed system has been tested for a wide variety of cement data. Initially through the user interface (Figure 5) user upload the excel data sheet and click on the result tab.

As soon user clicks on the result tab the software process the data and perform the analysis and generate the report of cement strength prediction. The software uses Python programming language is for preparing the NN model and different NN architectures are tested for the cement data to find the best architecture for cement strength prediction. This is determined by finding the percentage mean prediction error for each architecture. The list of NN architectures and mean prediction error is presented in Table 2. The ANN architecture that results in minimum mean prediction error is assumed to be the best architecture for the prediction. It is observed from Table 2 that 16-8-1 (16 Input – 8 Hidden – 1 Output) NN architecture is most suitable for the prediction in this application. This ANN architecture is used for the further analysis and prediction of cement strength. In the next analysis, the prediction of the cement strength has been calculated and the results are listed in Table 3.

It is observed from Table 3, that the forecasted output by the proposed ANN-based prediction system appears to be rational and is very close to those actually produced in the

Table 2. Neural network architectures with the performances.

	1	2	3	4	5	6	7	8	9	10	11
Architecture	16-2-1	16-4-1	16-6-1	16-7-1	16-8-1	16-9-1	16–10-1	16–12-1	16–14-1	16–16-1	16–18-1
2 days	2.30	1.74	1.95	1.35	0.70	0.85	0.98	1.06	1.61	1.85	1.86
7 days	0.68	0.51	0.98	0.93	0.51	0.67	1.40	0.72	0.80	1.12	0.79
28 days	1.77	1.05	1.32	0.88	0.44	0.58	0.81	0.99	1.02	1.09	1.96

Table 3. Cement strength experiments data and ANN (16-8-1 NN architecture) model's prediction data comparison.

	Strength (MPa)											
S. No.	2 days experimental data	2 days predicted data	Error	Error %	7 days experimental data	7 days predicted data	Error	Error %	28 days experimental data	28 days predicted data	Error	Error %
1	23.56	24.79	1.23	5.21	38.75	36.51	−2.24	−5.79	51.33	53.28	1.95	3.81
2	23.63	24.13	0.50	2.10	38.38	40.09	1.71	4.44	51.53	52.84	1.31	2.54
3	23.82	24.39	0.57	2.37	38.94	39.60	0.66	1.71	52.18	52.27	0.09	0.16
4	23.41	24.18	0.77	3.30	38.45	37.87	−0.58	−1.51	52.01	52.46	0.45	0.86
…	…	…	…	…	…	…	…	…	…	…	…	…
304	24.45	24.69	0.24	0.61	40.95	40.98	0.03	0.08	53.88	54.58	0.70	1.30

experimental data in cement manufacturing industries. The graphical output generated by the software for 28-day cement strength is as shown in Figure 4. Further, the output generated by the software application in tabular form is shown in Table 4.

Figure 4 shows output generated by the software for 28-day cement strength.

It is observed from Figure 4 that only a few cement strength predictions are having the difference more than ± 1. Further, the comparison between the LMBP-ANN prediction modal results and data received from the cement industry has been carried out. Mean square error and computational time have been recorded and the results are summarized in Table 5.

It is observed from Table 4 that the MSE using the cloud-based neural network predictions of cement strength is very less, while the computational time for the cement strength prediction using LMBP-ANN models is little more. The slightly longer computational time of LMBP-ANN is acceptable considering the overall better performance. The cement manufacturing industry experts considered these results to be acceptable. It has been observed that the cloud-based python software application for the cement strength prediction is producing

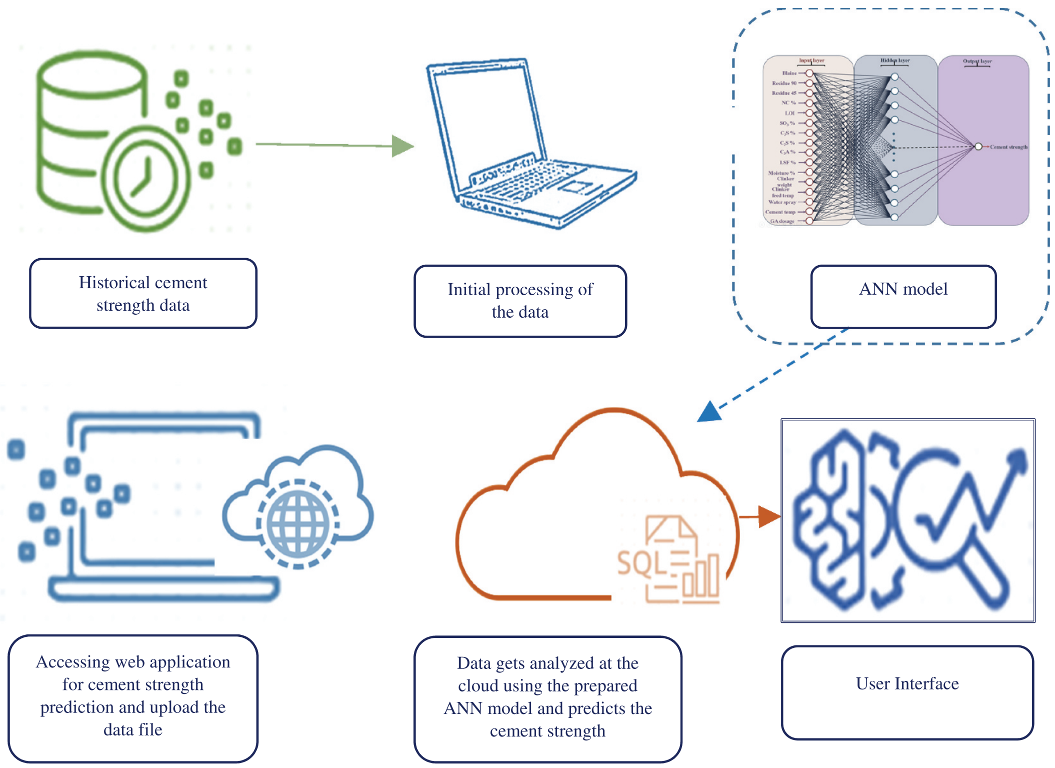

Figure 4. Cloud server setup.

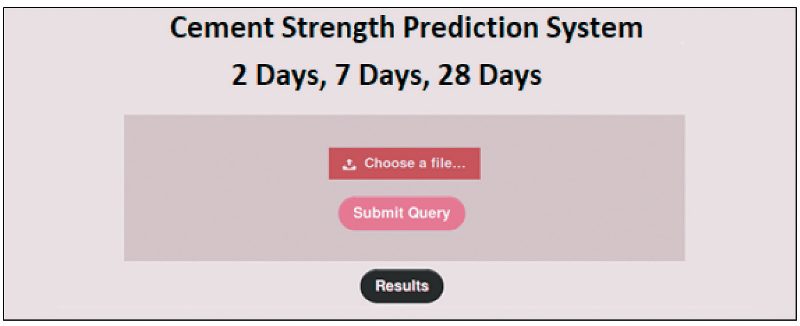

Figure 5. User interface of the proposed system.

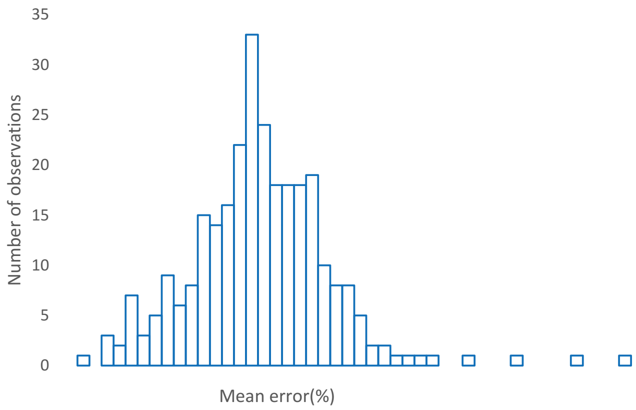

Figure 6. Output generated by the software for 28 days cement strength.

Table 4. Sample of output generated by a proposed software application (for 28-day cement strength).

Sl. No.	Predicted	Actual	Difference
1	52.128948	51.330002	−0.798946
2	52.157486	51.529999	−0.627487
3	52.626541	52.180000	−0.446541
4	52.179924	52.009998	−0.169926
5	52.211613	51.820000	−0.391613
6	52.877617	52.970001	0.092384
7	52.300045	52.970001	−0.760044
⋮	⋮	⋮	⋮
8	52.507359	52.599998	0.092640

Table 5. MSE comparison of regression and ANN model's cement strength prediction.

S. No.		Mean prediction error (%)	Computational time (in seconds)
1	2 days	0.6793	2.3645
2	7 days	0.6582	2.3719
3	28 days	0.6415	2.3607

better results and also provide the flexibility to access the application from any location to the user. In addition, before real production, different compositions of ingredients and duration can be experimented, which will reduce production time and costs.

Conclusion

In this paper, an effort has been made to develop an intelligent cloud-based system to automate the cement strength forecasting after 2, 7 and 28 days. A Python programming language has been used to develop a complete software application. The error in the cement strength predictions is within the acceptable range of ± 1 percentage. The final validation/testing of the system is carried out at manufacturing sites by checking the latest quality data available in the industry on a real-time basis. The adequacy of the developed model based on the ANN back-propagation algorithm is verified as a Pearson correlation of experimental value and predicted value. The estimated value of R for data experiments is 0.82539 and R^2 is 0.6813. It means that the expected and experimental results are directly related as the Pearson correlation coefficient falls between 0 and 1. The developed neural network model can therefore be used to predict cement strength with significant accuracy. The experimental cement concrete strength data, along with the projected LMBP-ANN data, were plotted on a scatter diagram for 2, 7 and 28 days. The ANN-based software application has been validated successfully for the input data set and produced the effective and accurate results.

Disclosure statement

No potential conflict of interest was reported by the author(s).

References

Abd Elaty, M. A. A.. (2014). Comprehensive strength prediction of Portland cement concrete with age using a new modal. *HBRC Journal*, *10*(2), 145–155. https://doi.org/10.1016/j.hbrcj.2013.09.005

Anand, K., Elangovan, S., & Rathinasuriyan, C. (2018). Modeling and prediction of weld strength in ultrasonic metal welding process using artificial neural network and multiple regression method. *Material Science & Engineering International Journal*, *2*(2), 39–46. https://doi.org/10.15406/mseij.2018.02.00032

Behnood, A., Behnood, V., Gharehveran, M. M., & Alyamac, K. E. (2017). Prediction of the compressive strength of normal and high-performance concretes using M5P model tree algorithm. *Construction and Building Materials*, *142*, 199–207. https://doi.org/10.1016/j.conbuildmat.2017.03.061

Bingöl, A. F., Tortum, A., & Gül, R. (2013). Neural networks analysis of compressive strength of lightweight concrete after high temperatures. *Materials & Design*, *52*, 258–264. https://doi.org/10.1016/j.matdes.2013.05.022

Chandwani, V., Agrawal, V., & Nagar, R. (2014). Applications of artificial neural networks in modeling compressive strength of concrete: A state of the art review. *International Journal of Current Engineering and Technology*, *4*(4), 2949–2956. http://inpressco.com/wp-content/uploads/2014/09/Paper1242949-2956.pdf

Chopra, P., Sharma, R. K., & Kumar, M. (2016). Prediction of compressive strength of concrete using artificial neural network and genetic programming. *Advances in Materials Science and Engineering*, *2016*(4), 1–10. https://doi.org/10.1155/2016/7648467. Article ID 7648467.

Hassan, S., Farhan, M., Mangayil, R., Huttunen, H., & Aho, T. (2013). Bioprocess data mining using regularized regression and random forests. *BMC Systems Biology*, *7*(1), S5. https://doi.org/10.1186/1752-0509-7-S1-S5

Hendi, A., Behravanb, A., Mostofinejada, D., Sedaghatdoostc, A., & Amini, M. (2018). A step towards green concrete: Effect of waste silica powder usage under HCl attack. *Journal of Cleaner Production*, *188*, 278–289. https://doi.org/10.1016/j.jclepro.2018.03.288

Kewalramani, M., & Gupta, R. (2006). Concrete compressive strength prediction using ultrasonic pulse velocity through artificial neural networks. *Automation in Construction*, *15*(3), 374–379. https://doi.org/10.1016/j.autcon.2005.07.003

Khademi, F., Akbari, M., Jamal, S. M., & Nikoo, M. (2017). Multiple linear regression, artificial neural network, and fuzzy logic prediction of 28 days compressive strength of concrete. *Frontiers of Structural and Civil Engineering*, *11*(1), 90–99. https://doi.org/10.1007/s11709-016-0363-9

Khashman, A., & Akpinar, P. (2017). Non-destructive prediction of concrete compressive strength using neural networks. *Procedia Computer Science*, *108*, 2358–2362. https://doi.org/10.1016/j.procs.2017.05.039

Kosmatka, S. H., Kerkhoff, B., & Panarese, W. C. (2011). *Design and control of concrete mixtures* (*14th* ed.). Portland Cement Association.

Kumar, N., & Mittal, R. K. (2012) Cloud computing setup for a campus environment. In *International conference on cloud computing technology application management* (pp. 88–91). Dubai.

Kumar, N., & Naranje, V. (2019) Prediction of cement strength using machine learning approach. In *Amity internation conference artificial intelligence* (pp. 239–243). Dubai.

Liu, Z., Liu, X., Ma, J., & Gao, H. (2018). An optimized computational framework for isolation forest. *Mathematical Problems in Engineering*, *2018*, Article ID 2318763, 13 pp. https://doi.org/10.1155/2018/2318763

Mejdoul, R., Taqi, M., & Belouaggadia, N. (2013). Artificial neural network based prediction model of daily global solar radiation in Morocco. *Journal of Renewable and Sustainable Energy*, *5*(6), 63137-1–63137-9. https://doi.org/10.1063/1.4854817

Naranje, V., Kashid, S., Kumar, S., Ghodke, A., & Hussein, H. M. A. (2016). Prediction of life of deep drawing die using artificial neural network. *Advances in Material and Processing Technologies*, *2*(1), 132–142. https://doi.org/10.1080/2374068X.2016.1160601

Naranje, V., & Salunkhe, S. (2020). A programmatic approach for the prediction of service life of deep drawing die using ANN. In H. Voruganti, K. Kumar, P. Krishna, & X. Jin (Eds.), *Advances in applied mechanical engineering. Lecture notes in mechanical engineering*(pp. 465–473). Springer.

Oh, H. S., & Oh, K. C. (2016). An empirical estimation procedure of concrete compressive strength based on the in-situ nondestructive tests result of the existing bridges. *Journal of the Korea Institute for Structural Maintenance and Inspection*, *20*(4), 111–119 (in Korean). https://doi.org/10.11112/jksmi.2016.20.4.111

Poole, T. S., & Harrington, P. J. (1996). *An evaluation of the maturity method (ASTM ca. 1074) for use in mass concrete*. Technical Report SL-96-16.

Rafi, M. M., & Nasir, M. M. (2016). Models for prediction of 28-day concrete compressive strength. *Journal of Testing and Evaluation*, *44*(3), 1217–1228. https://doi.org/10.1520/JTE20140139

Salunkhe, S., Rajamani, D., Balasubramanian, E., & Chandrasekhar, U. (2019). Prediction of life of piercing punches using artificial neural network (ANN) and adaptive neuro fuzzy inference systems (ANFIS). *International Journal of Materials Engineering Innovation*, *10*(1), 20–33. https://doi.org/10.1504/IJMATEI.2019.097892

Sobhani, J., Najimi, M., Pourkhorshidi, A., & Parhizkar, T. (2010). Prediction of the compressive strength of no-slump concrete: A comparative study of regression, neural network and ANFIS models. *Construction and Building Materials*, *24*(5), 709–718. https://doi.org/10.1016/j.conbuildmat.2009.10.037

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, *15*(56), 1929–1958. http://jmlr.org/papers/v15/srivastava14a.html

Yuan, Z., Wang, L.-N., & Ji, X. (2014). Prediction of concrete compressive strength: Research on hybrid models genetic based algorithms and ANFIS. *Advances in Engineering Software*, *67*, 156–163. https://doi.org/10.1016/j.advengsoft.2013.09.004

Zhou, H., Deng, Z., Xia, Y., & Fu, M. (2016). A new sampling method in particle filter based on Pearson correlation coefficient. *Neurocomputing*, *216*, 208–215. https://doi.org/10.1016/j.neucom.2016.07.036