1st Croatian Conference on Earthquake Engineering
1CroCEE
22-24 March 2021 Zagreb, Croatia

DOI: https://doi.org/10.5592/CO/1CroCEE.2021.203

A machine learning framework for automated ground motion prediction

Moheldeen A. Hejazi[1], Serra Tinbir[2], Pooya Ghaffari Khalifani[3], Ali Sari[4]

[1] *Ph.D. Student*, Civil Engineering Department at Istanbul Technical University, Turkey, *hejazi19@itu.edu.tr*
[2] *Graduate student*, Earthquake Engineering Department at Istanbul Technical University, Turkey, *tinbir19@itu.edu.tr*
[3] *Ph.D. candidat*e, Earthquake Engineering Department at Istanbul Technical University, Turkey, *Pghaffari@itu.edu.tr*
[4] *Associate professor*, Civil Engineering Department at Istanbul Technical University, Turkey, *asari@itu.edu.tr*

Abstract

The continuous expansion of seismic catalogs is increasingly challenging the validity of existing ground motion prediction equations. For such real-time data, the transition of ground motion modeling toward automation became eminent. This article put forth a dynamic intelligent ground motion prediction system, automated through a novel hybridization of neural networks with a multi-objective swarm intelligence optimization to facilitate updated predictions. Under the proposed framework, acceleration data are parsed from catalogs in real-time and the continuous stream of seismic data is analyzed to both (i) optimize model predictions and (ii) minimize its computational demand. Though built adaptive to different geographical locations, the system in this article is presented in the context of Turkey. Therefore, real-time strong ground-motion records are obtained from the AFAD database, where the peak ground acceleration (PGA), velocity (PGV), and displacement (PGD) are examined against various seismic variables including earthquake magnitude, source to site distance, average shear-wave velocity, and focal mechanism. The model predictions were verified against a broad testing sample and predictions by various GMPE models for Turkey. Thereafter, the model stability was examined through an investigation into the sensitivity of the PGA, PGV, and PGD predictions to data and parameters' discrepancies. Finally, this article discusses the future potentials and challenges facing the developed framework.

Key words: ground-motion parameters, prediction equations, machine learning, optimization algorithms, Earthquake catalogs

1 Introduction

In 1999 an earthquake occurred at Kocaeli causing a huge life and property loss and triggering a massive shift and expansion in the country's seismic network. In the wake of the 1999's earthquake, seismological studies and research gained momentum and the Turkish seismic network expanded drastically to address the periodic seismic activity of the North Anatolian fault system (NAF). Since then, Turkey has sustained many large earthquakes (2011 Van Mw=7.1, 2012 Fethiye Mw=6.0, 2017 Bodrum Mw=6.6, etc.) and the last months of 2019 witnessed significant seismic activities. In September 2019, an earthquake occurred at Silivri, Marmara Sea with a magnitude of 5.8 and felt in Istanbul and nearby cities. In January 2020, the Elazığ earthquake followed this major earthquake on the eastern Anatolian fault system (EAF) with a 6.5-moment magnitude. In October 2020, the largest earthquake occurred at the Aegean Sea with a 7.0-moment magnitude. This event affected Izmir in Turkey and it caused 177 deaths, 1034 injures and 15000 people became homeless. Considering the frequency and the long record of Turkey's seismic activities (and countries with similar patterns), the development of accurate seismic modeling necessitates the inclusivity of new and upcoming records.
Hitherto, many studies tackled the modeling of the ground motion prediction equations (GMPE) both in Turkey and worldwide[1–4]. These models are crucial for estimating earthquake parameters to figure out the earthquake forces that will be used for structural design. Nevertheless, the stagnant nature of the preceding models challenges their accuracy with time. In Turkey, the seismic network has grown so dramatically since 1999 that the current network consists of 799 operating accelerometers. Moreover, the seismic database is expanding exponentially as depicted in Figure 1.

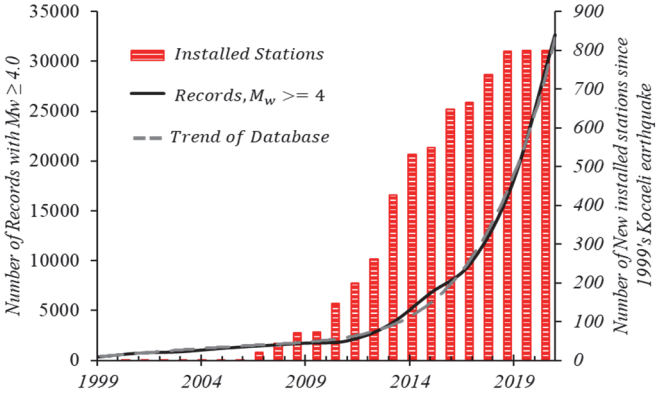

Figure 1. The seismic network and database expansion following 1999's earthquake

To cope with this expanding nature, this study presents a dynamic intelligent ground motion prediction system, capable of updating its structure and optimizing its predic-

tion with minimal computational demand. In contrast to other machine learning-based models in the literature which still adapts the bulk single time training with constant data. The proposed intelligent system is built to monitor, retrieve and inform its prediction spontaneously by implementing optimized data scraping techniques with online machine learning algorithms. Therefore, though the system can be deployed on a global scale, this study introduces CATALOGIST (Catalogues' intelligent system for tectonics predictions) in the context of Turkey, to be the first fully automated system for ground motion prediction.

This study is structured as follows, the next section establishes a brief background on the seismology and major fault systems in Turkey and the Anatolian plateau. Section three "Model parameters and seismic catalog" highlights the up-to-date available data in the Turkish ground motion database (AFAD-TADAS) and its statistics and parameters of significance to the developed model. Meanwhile, section four introduces "The automated data monitoring system (ADMS)" and its optimization for the data retrieval from the web-based database. Thereafter section five delivers the machine learning model and its training for the prediction of future ground motions. The sixth section provides the comparisons between the introduced model and a model from the literature developed in 2003. The comparison is made with (i) only the data available to the 2003's model, and (ii) up-to-date records in the Turkish database to investigate the degradation of accuracy with time for stagnant data-based GMPEs.

2 Background

Turkey is located mainly on the Anatolian tectonic plate. Surrounded by the Arabian plate on the east and the African plate on the south, the Anatolian plate is forced constantly to move to the west. However, the northern side of Turkey is bounded by the Eurasian plate, and the Interaction between these plates (the Anatolian and the Eurasian) defines the sources for the major tectonic activities in Turkey. The major earthquake sources of Turkey are the North Anatolian fault (NAF), East Anatolian fault (EAF), and Aegean fault zone (AFZ). NAF has a right-lateral strike-slip fault mechanism and since the 1939 Erzincan earthquake, it has a periodic activity within 20 years on average. EAF is located on the Southside of Turkey. Due to the interaction between the Arabian-African and Anatolian plates, EAF converges with NAF around Karliova, Bingol which extends to Erzurum. AFZ is a set of normal faults and they are converging on the Marmara Sea with NAF as some smaller fault sets[5,6].

To model and predict ground motion parameters associated with such activities, the ground motion prediction equations (GMPEs) are developed. GMPEs relate the ground-motion parameters (PGA, PGD, PGV) to the different independent incident and site parameters like earthquake magnitude (with different magnitude scales), distance from the source to the site (with its various measures), local site conditions, earthquake source characteristics and fault mechanism, and the wave propagation. In most cases,

the physical parameters of stress drop, rupture propagation, directivity, basin effects, and nonlinear soil behavior are not addressed in these models. A standard way to generate the GMPEs from the recorded strong-motion data is using regression analysis with fixed or mixed effect for inter and intra-event variability. In each form, a straightforward statistical model is also developed to explain the tendency of the ground-motion parameters with other parameters at the station.

Correlating PGA, PGV, and PGD with the predictor parameters during a mathematical form isn't a straightforward task, this is directly associated with the highly nonlinear relationships. Additionally, the numerous limitations of the statistical techniques strongly affect the capabilities of the regression-based GMPEs. Most typically used regression analyses can have large uncertainties. Its major drawbacks for the idealization of complex processes, approximation, and averaging widely varying prototype conditions. Furthermore, the regression analysis tries to model the character of the corresponding problem by a pre-defined linear or nonlinear equation (or set of equations). Another major restriction in the application of the regression analysis is that the assumption of normality of residuals. Thus, the developed attenuation models are often limited in their ability to reliably simulate the complex behavior of the ground-motion parameters. The aforementioned deficiencies indicate the requirement of employing more comprehensive methods, that are adaptive to decrease the errors for the ground-motion parameters estimates[4,7].

3 Model parameters and seismic catalogue

In this project, an automated catalog monitoring system is tasked with retrieving the ground motion records in real-time, and with storing and processing the various parameters to feed them in an intelligent ground motion prediction model. Therefore, it is essential at this point to introduce the main parameters commonly used when establishing empirical attenuation models. The ground-motion prediction equation (GMPE) intuition is best presented with the following conventional structure Eq (1).

$$\log Y_{ij} = f\left(M_i, r_{ij}, \overline{\theta}\right) + \varepsilon_{ij} \tag{1}$$

In this expression, Y_{ij} denotes the response of interest (PGA, PGV, PGD ...) for an event i and record j, while M_i is the event's i magnitude and is the coefficients matrix and are the residual error. Many models were developed to incorporate more parameters and to further account for the uncertainties and correlations among the various records from a single station or associated with a single event. However, the conventional structure was implemented in this investigation with the inputs being moment magnitude, epicentral distance, and shear wave velocity in the upper 30 meters. The outputs were selected as the peak ground acceleration (PGA), peak ground velocity (PGV), and the peak ground displacement (PGD) [7].

3.1 Data catalog

In Turkey, due to the tectonic activities in this region, and the increase of awareness since 1999's Kocaeli earthquake, the Turkish seismic network witnessed a great expansion. Nowadays, strong ground motion records are published and maintained by the disaster and emergency management presidency of Turkey (AFAD). Moreover, recently in 2020, AFAD launched the Turkish Accelerometric Database and analysis system (TADAS) which provides access to records from 1976 with daily updates and five-day publication gaps for the new records. TADAS provides both raw acceleration records and processed data for all stations operating in the Turkish network. The processed data files are available in ASCII, ASDF, MINISEED, or SAC formats. In this study, the TADAS database is the main source for data retrieval.
The main parameters of interest in TADAS database are, date–time, location (longitude, latitude), elevation, depth, magnitudes (, , , ,), source to site distance (, , , in km), (m/s) and soil classification according to Eurocode, sampling interval, lowpass, and high-pass Butterworth filtering values, peak ground acceleration (cm/s2), peak ground velocity (cm/s), peak ground displacement (cm), fault mechanism.

3.2 Investigated parameters and model definition

In this study, the parameters selected are the magnitude, which is defined in terms of moment magnitude which eliminates the saturation effects for magnitudes greater than 6.0. for the records with missing moment magnitude and available local magnitude the local magnitude is used for all events with M less than or equal to 6.0. Here it is assumed that for magnitudes less than 6.0 the ML is approximately equal to Mw as done by Ozbey[7]. In this study, only the records with a moment magnitude higher than 5.0 were selected for analysis. On the other hand, the epicentral distance was implemented for this investigation as it was found available for all the records. However, this would require more elaborate investigation in the future since the differences between the several distance definitions tend to have significance in the near field data (this study was limited to the distances between 0 km and up to 200 km). The effect of local site conditions is also included in the attenuation models studied through the inclusion of the shear wave velocity. Furthermore, this study implemented the site classification by the Turkish code (ZA for Vs30>1500 m/s, ZB for Vs30 between 760 and 1500, ZC for values between 360 and 760, and ZD for values between 180 and 360, and ZE for values less than 180). The data used in this investigation with the distribution across various distances, various magnitudes, and different site conditions is shown in figure.2. finally, although distinguishing between the various fault mechanisms generally is considered important in this study, it was not included explicitly in the input. For the output peak response parameters (PGA, PGV, PGD) the geometric mean of the two horizontal components was estimated and used.

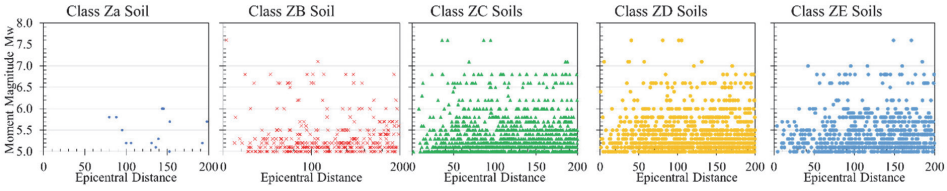

Figure 2. Data distribution for soil classes ZA and ZB, ZC, ZD, and ZE, respectively

4 The Automated Data Monitoring System (ADMS)

Due to the highly expanding nature of today's catalog, the urge for informed models in real-time became inevitable. Subsequently, this investigation relayed first on real-time retrieval of the seismic records as they are introduced in the seismic catalogs. The data cycle is illustrated in this section from the monitoring of the seismic catalog to the final retrieval and safekeeping of the acquired records.

4.1 Data Monitoring Unit (DMU)

To provide automated real-time updates for the intelligent ground motion prediction model, the software was written in Python language to deliver periodic and daily-request to AFAD's database to query for new records. However, to minimize the load on the AFAD database, the software request interval was maintained minimal. The software triggers every 24 hours at 3:00 AM at Istanbul local time (1:00 AM GMT) and sends a request to the Turkey accelerometric database and analysis system (TADAS) for new records. The query cycle network performance counters are given in figure.3. The query process starts with Local Processing for the new query inputs which are handled in an input string containing the magnitudes bounds, the epicentral distance bounds, and geographical and time constraints. This is then handled to a resolved IP in the form of an HTTP request through an established TCP connection. The query process ends with receiving the database server's response containing the number of newly available records (if any). During testing and subsequent operations execution time of this cycle ranged from 5 seconds up to 14 seconds depending on the number of found records with a fixed initial connection establishment of 4 seconds.

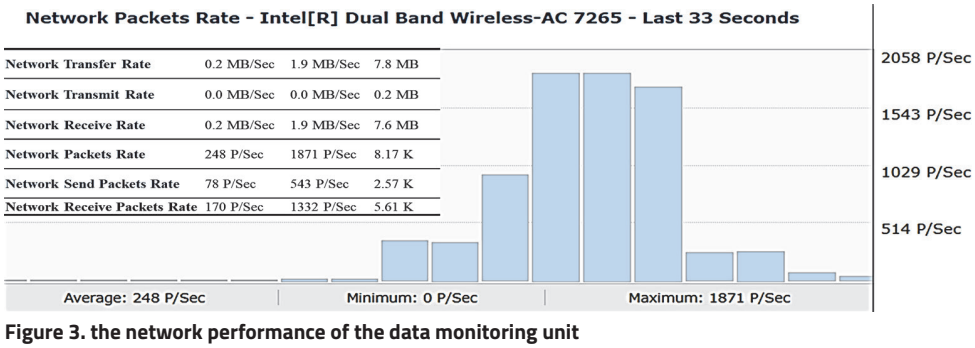

Figure 3. the network performance of the data monitoring unit

Once the data monitoring unit (triggered by a time scheduler) receives the database response, it triggers the data indexing unit (DIU).

4.2 Data Indexing Unit (DIU)

Exploiting the same connection established by the data monitoring unit DMU, the data indexing unit scraps the database response collecting only the three essential identifiers (event, station code, and magnitude) for each retrieved record. These identifiers are then examined by the DIU in terms of both the number of new records and the magnitudes of these records. This is to allow for the conditional triggering of the bulk records retrieval process by the data retrieval unit and to maintain a minimal interaction time with the local database.
The execution time of the DIU was optimized by exploiting the pre-established connection with the TADAS server. Therefore, the execution time ranges between 0.0003 milliseconds up to 0.864 milliseconds depending on the number of new records to index. The output of the DIU indexing is stored and handled in JSON format for efficient retrieval in the data retrieval stage.

4.3 Data Retrieval Unit (DRU)

The triggering criteria for the data retrieval unit DRU are flexible and can be set to any value with an update versus efficiency trad-off. Herein, for the sake of the present study, the DRU triggering criteria are listed in the table.1.

Table 1. Triggering criteria adopted for the present investigation to trigger the data retrieval unit (DRU)

Unit	Criterion	Threshold
DMU	Time	Daily at 3:00 AM at IST local time (1:00 AM GMT)
DRU	Moment Magnitude	Urgent retrieval if occur
	Number of new events	Exceptional retrieval if number of new events3
	Number of new records	Typical retrieval if number of new records40

Once the threshold criteria are met in terms of either an urgent event's magnitude (or the number of new events and record (with) the data retrieval unit DRU starts processing for the retrieval of JSON formatted records identifiers indexed by the DIU. However, due to the high demand of this stage and the network delays present when interacting with the web-based database, the single processing core and single-threaded operations were deemed inefficient. Therefore, a multi-threaded multiprocessing scheme was examined for the enhanced speed of the data retrieval.

5 Results and discussion

5.1 Optimizing data retrieval with parallelism

In this investigation the Python programming language was implemented for the monitoring, scraping, and processing of the data after storage. Inherently, however, Python is a linear language that was not designed considering more than one core. therefore, python includes a global interpreter lock (GIL) to ensure thread-safety and to globally enforce lock when accessing a Python object. However, various libraries were developed to bypass this limitation (eg, Dask, Joblib, and multiprocessing), and thus, facilitating the use of both multiprocessing and multithreading in this study[8,9].

Nevertheless, the libraries just mentioned do not coordinate the spawning and pooling of threads, which might cause over-subscription (the case where more threads are active than the hardware available resources can handle, leading to frequent context switches and sub-optimal performance. Therefore, the practice of determining the optimal number of threads per processor core is widely adapted to tune the performance with a specific CPU's scheduler to prevent the processor overhead with multiprocessing and threading. It is also crucial for benchmarking whether an input/output (I/O) bound or a processor overhead is dominating the execution time in web scraping[9].

In this investigation, a typical 4 physical cores Intel(R) Core (TM) i7-8550U CPU @ 1.80GHz processor was used with 12 GB ram. Moreover, GPU processing capabilities were not employed in this investigation. The optimization was done by the incremental decrease of the records per thread number and by the incremental deployment of further CPU cores into the processing pool. Figure.4 presents the results of the optimization for the retrieval of 20 records with threads between 1 and 20 for a single processor (records per threads number between 20 and 1, respectively). Thereafter, an optimization of the processing core number was conducted with (1, 2, 3) cores deployed (the 4th core was preserved for system operations).

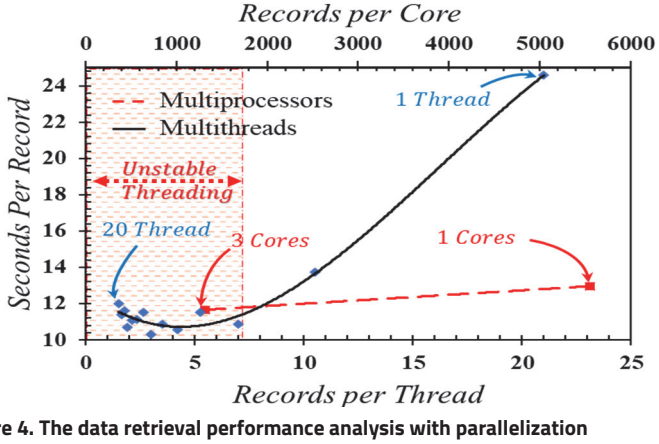

Figure 4. The data retrieval performance analysis with parallelization

Following the optimization analysis as per figure.4 it was found that the process is mainly I/O bonded; where I/O (Input/Output) bounds are due to the idle time in which the processing thread/core is waiting for the server response. The efficiency of incorporating a second thread per processor core results in up to 50 % execution time reduction. Moreover, further spawning of a thread increases the accuracy up to 56 %. However, at this level of computational demand and beyond the threads started finishing at the same time resulting in overhead for the processing where the schedular switches repeatedly between various active threads which halts the operations. Table

Counter	Average	Maximum	Total
Network Transfer Rate	0.8 MB/Sec	1.9 MB/Sec	104.0 MB
Network Receive Rate	0.8 MB/Sec	1.9 MB/Sec	101.8 MB
Network Packets Rate	801 P/Sec	2137 P/Sec	106.54 K
Network Send Packets Rate	245 P/Sec	816 P/Sec	32.59 K
Network Receive P. Rate	556 P/Sec	1364 P/Sec	73.95 K

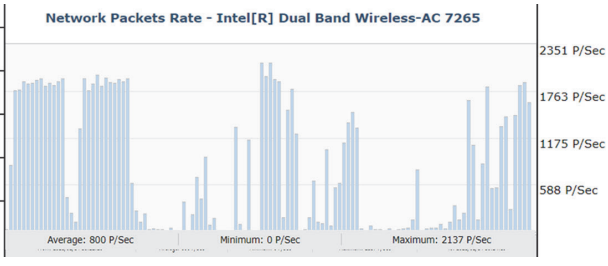

Figure 5. Network performance during the simultaneous retrieval of six records by three CPU cores and two threads each

With the multiprocessing case, a total of 5600 records were split among (1, 2, 3) processors with 2 threads per processor to overcome the waiting for the server response. Moreover, for the multiprocessors investigation, various deposits were assigned to each processor to prevent data overlaps and a potential corruption of the local database. Furthermore, separate loggers were assigned to maintain a record of potential issues and trace the responsible processor. Finally, it was found that 3 processors with two threads each yield relatively higher performance and helps to exploit the processing unit more efficiently. Fig.5 next depicts the network performance profiling for six records from the TADAS database by the optimized number of CPUs and Threads. The farthest left side depicts a single core with two successive record downloads while the second core is sending two consecutive requests in the middle portion. This is followed by the third core directly sending two threads in the farthest right with requests. This scheduling between CPUs use of the network is done internally by the schedular.

Table 2. The processing performance for the monitoring and retrieval units (computation, memory, and data processing rate)

Counter	DRU		DMU	
	Average	Maximum	Average	Maximum
Process CPU Usage	0.6 %	6.8 %	0.7 %	4.2 %
Process Memory Used	1.52 GB	1.56 GB	890.8 MB	900.0 MB
Process Thread Count	86	90	86	88
Process Handle Count	1755	1811	1951	1973
Process Data Rate	0.0 MB/Sec	1.1 MB/Sec	0.0 MB/Sec	0.9 MB/Sec

5.2 Validation and sample training session

Following the development of the ADMS system, a sample neural network training session was attached to it using Keras and tensor flow. The trained network was compared to the model developed by Ozbey,2003[7] to investigate its accuracy at different time frames (thus, if it is accurate today, will it be accurate before 20 years or after 20 years?). Therefore, the selection of this model is mainly done to investigate the prediction's durability with time and different degrees of noise in the data. It was found that the model was accurate satisfactorily compared to Ozbey's model with a slope in linear regression between 0.85 and 0.91. Therefore, the results of the model training using the entire TADAS records up-to 2003 were compared to Ozbey's findings. The two models showed comparable accuracy. On the other hand, the prediction considering all data up to 2021 with Ozbey's reveals the decrease in its accuracy. This is associated with the enormous number of records that were implemented in the proposed model (3844 records) compared to the records available in 2003 for Ozbey's model (195 records). The model along with the linear regression result is shown in Fig.6.

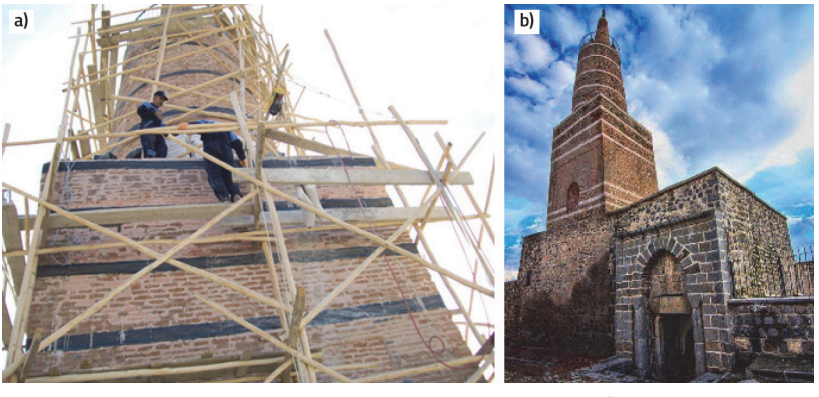

Figure 6. Proposed model prediction against Ozbey's 2003 model for ground motion prediction

6 Conclusions

In this investigation, it was found that the implementation of automation to the ground motion records retrieval, monitoring, and processing up to the training can be achieved efficiently even with a typical computational resource. In this study, an automated system was optimized through multiprocessing and multithreading to optimize its computational demand. Thereafter, the accuracy of the prediction of the proposed system was examined in comparison to a model for the ground motion prediction in Turkey. The model accuracy was comparable to the regression-based model for the same data. However, for the increased amount of data present with time the proposed model showed superior accuracy.

The system introduced in this study would be further advanced with optimization to the machine learning training computational cost, and the accuracy of prediction shall be enhanced further. Furthermore, it is planned to expand this system to PEER and IRIS DMC databases to increase its geographical coverage beyond Turkey.

Acknowledgments

The authors are in debt to the general directorate of disaster management (AFAD), Turkey, and to the Turkey accelerometric database and analysis system (TADAS) for providing access to their database. The authors utilized the open-source projects CProfileV 1.0.7 and SysGauge v.7.6.38 for the inter-software and across the operating system performance profiling and optimization, respectively. The selenium 3.141.0 project was likewise implemented for web-testing and scraping in python.

References

[1] Baltay, A.S., Hanks, T.C. (2014): Understanding the magnitude dependence of PGA and PGV in NGA-West 2 data. Bull. Seismol. Soc. Am. 104(6): 2851.

[2] Hassani, B., Atkinson, G.M. (2018): Adjustable generic ground-motion prediction equation based on equivalent point-source simulations: Accounting for kappa effects. Bull. Seismol. Soc. Am. 108(2): 913.

[3] Derras, B., Bard, P.Y., Cotton, F., Bekkouche, A. (2012): Adapting the neural network approach to PGA prediction: An example based on the KiK-net data. Bull. Seismol. Soc. Am. 102(4): 1446.

[4] Al Atik, L., Youngs, R.R. (2014): Epistemic uncertainty for NGA-West2 models. Earthq. Spectra. 30(3): 1301.

[5] GURBOGA S. NEO- AND SEISMO-TECTONIC CHARACTERISTICS OF THE YENĠGEDĠZ (KÜTAHYA) AREA. Middle East Technical University.

[6] Ulusay, R., Tuncay, E., Sonmez, H., Gokceoglu, C. (2004): An attenuation relationship based on Turkish strong motion data and iso-acceleration map of Turkey. Eng. Geol. 74(3–4): 265.

[7] Özbey, C., Sari, A., Manuel, L., Erdik, M., Fahjan, Y. (2004): An empirical attenuation relationship for Northwestern Turkey ground motion using a random effects approach. Soil Dyn. Earthq. Eng. 24(2): 115.

[8] Salvatier, J., Wiecki, T.V., Fonnesbeck, C. (2016): Probabilistic programming in Python using PyMC3. PeerJ Comput. Sci. 2016(4): 1.

[9] Vasilev, I., Slater, D., Spacagna, G., Roelants, P., Zocca, V. (2019): Payton Deep Learning. P Dhandre, ed. Packt Publishing, Birmingham, UK.