

Low Voltage Warning System for Stand-Alone Metering Station Using AI on the Edge

Krzysztof Turchan^(⊠)[™] and Krzysztof Piotrowski[™]

IHP - Leibniz-Institut für innovative Mikroelektronik, Im Technologiepark 25, 15236 Frankfurt (Oder), Germany {turchan,piotrowski}@ihp-microelectronics.com https://www.ihp-microelectronics.com

Abstract. Artificial intelligence is used in many different aspects of life these days. It is also increasingly used to create intelligent embedded devices. Their main task is to demonstrate intelligent features using limited hardware resources. This paper introduces the possibility of using AI on the Edge as a system to warn of low voltage on the battery in a standalone metering station powered by solar panels. The paper presents theoretical knowledge about artificial intelligence on the Edge, time series and algorithms in time series forecasting. In addition, a practical approach is presented in the Approach section. It also describes the stand-alone measurement station that was used, what type of data was collected and used, and what type of the time series forecasting algorithm was used. The main advantage of the approach is the practical use of Holt's Linear Trend method in battery voltage forecasting. Work on the project is still in progress therefore the results presented in the paper are generated during simulation.

Keywords: Artificial intelligence on the edge \cdot Time series forecasting \cdot Intelligent battery warning system

1 Introduction

1.1 Artificial Intelligence

It is not easy to clearly define what is artificial intelligence (AI). AI could be defined as a set of algorithms which operation is similar to intelligent human behaviour [1]. All current AI-based systems belong to the so-called Narrow intelligence [2]. Narrow intelligence is one of three defined types of AI (Narrow AI, General AI, Super AI). This type of AI is designed to perform a single task, which can not be automatically applied to other tasks in other applications. Narrow AI-based systems can be more or less complex, depending on the computing power of the device they run on. There are two approaches to implementing AI. The first approach concerns the implementation of increasingly complex artificial intelligence systems to achieve the so-called General intelligence. The second approach is to implement AI algorithms on embedded devices, which are highly specialized devices designed for one or very few specific purposes and are usually embedded or included as part of a larger system. This type of AI is called AI on the Edge [3].

AI on the Edge is mainly related to above mentioned embedded devices. The idea behind this solution is to use the AI algorithms directly on the embedded device, without having to send large amounts of data to a server [3]. This solution offers many possibilities such as creating smart sensors, intelligent cameras, etc., by using various artificial intelligence methods depending on the purpose.

Forecasting is one of the most popular artificial intelligence methods whose main task is to predict future values based on data collected in the past [4]. Nowadays, predictions are made in every field where historical data is available. This is due to the fact that it gives the possibility to estimate different values, which often brings different kinds of benefits. Examples of using artificial intelligence for data prediction purposes include:

- Stock price prediction
- House price forecasting
- Weather forecasting

All of these examples have in common the fact that predictions are based on historical time-stamped data. This data is the domain of time series.

1.2 Time Series Forecasting

Time series forecasting is the process of analysing time series data using modelling and statistics to make predictions. The accuracy of the prediction depends mainly on the available data. However, an important role is played by the proper selection of the algorithm for this data. There are many algorithms for predicting time series and they are selected according to several criteria.

- Amount of data at hand the more observations, the better problem understanding [4,5].
- **Time horizons and amount of variables** the shortest time horizon, the less chance for unpredictable variables and errors [4,5].
- Data quality the data must be completed, cleared and without duplication [4,5].

The most important thing is to collect data with the same interval. This allows finding trends, seasonality or cyclic behaviour. Data gaps can be filled by algorithms but this makes the final model not as accurate as it could be [5].

1.3 Forecasting Algorithms

It is not always possible that one forecasting algorithm will work equally well for different types of plots (historical data). Very often it is necessary to select an algorithm based on observations of the data. This section introduces several time series forecasting methods, along with their purpose [6].

Naive Method is a technique that assumes that the next expected point is equal to the previously observed point, due to the not very large variability of the data over time. This method is not dedicated to highly variable data sets

Simple Average is a technique that fits with data that changes over a small period of time, while the average over a longer period remains constant [6].

Moving Average. Using a simple moving average model, we predict the next value (or values) in a time series based on the average of a fixed finite number of previous values [6].

Holt's Linear Trend Method. Any dataset that follows a trend can use this method for forecasting. Thanks to the method it is possible to predict future value using a linear trend. [7]. Holt's method has three separate equations that work together to generate a final forecast. The first Eq. (1) is a smoothing equation. It adjusts the last smoothed value to the trend of the last period. The trend is updated over time using Eq. (2). It involves determining the difference between the last two smoothed values.

$$L_t = \alpha Y_t + (1 - \alpha)(L_{t-1} + T_{t-1}) \tag{1}$$

$$T_t = \beta (L_t - L_{t-1}) + (1 - \beta) T_{t-1}$$
(2)

$$Y_{t+1} = L_t + (h)T_t (3)$$

where:

 Y_{t+1} is value at time t+1, α and β are smoothing parameters, h is forecast horizon.

2 Motivation

Applying the techniques and methods presented in the previous section, the idea was to implement a low battery warning system for a stand-alone measurement station.

As is well known, such measurement stations are often powered by solar panels because they could be located in places far from conventional power sources. This solution could generate problems because the panels that are connected to the measurement station could not be able to keep up with charging the batteries. Of course, it is possible to install panels with a much higher power than expected, but this generates additional costs. Instead of that a low battery warning system has been designed. The exact purpose of this system is to predict the voltage of the battery a few days ahead and to warn when the voltage level may fall below the required one. Thanks to this the user responsible for the maintenance of the station is able to react to a warning at the appropriate time.

The second problem of stand-alone measurement stations is the possibility of losing access to the network. In this case, data that could not be sent must be stored in the embedded device's memory, which is often very limited. For this reason, it is important to send as little data as possible. The most important data that must be sent is information about the continuous operation of the station and warnings about possible battery discharge. Using the forecasting algorithm on the microcontroller makes it unnecessary to send all the electrical measurement data.

3 Approach

This section provides an approach to solving the problem presented in the previous section. Furthermore, it provides information on what type of data is collected on the microcontroller and an algorithm that is responsible for forecasting. It also gives ways to optimize algorithms to save as much microcontroller processing power as possible.

3.1 Measurement Station

The warning system covered in this paper is part of a larger measurement station as shown in Fig. 1 that collects data from the environment. The whole station is powered by a battery that is charged by solar panels. The station power supply section uses a Three-Channel (3-Ch) DC meter that records voltage and current measurements data for three independent channels. 3-Ch meter is a special purpose board designed at Institute for High Performance IHP, based on MSP430F6779 microcontroller.

The first channel that is shown in Fig. 1 measures voltage and current generated on the solar panel. This allows inferring the daily insolation level. Then, the second channel is used to measure battery voltage and current. With this data, it is possible to observe the charging and discharging cycles of the battery. The last channel serves to measure the current consumption of the environment measurement system. In addition, there is a sensor for collecting the temperature data. The schematic shows also an arduino MKR1010 WiFi board that serves as an network gate for the data.



Fig. 1. Measurement station schematic

3.2 Dataset and Algorithm

This entire dataset can be used to train an advanced artificial intelligence model such as Artificial neural network (such a case will be an object of future research). However, the application must run on an embedded device with relatively low computing power and it is important to use as simple algorithm as possible. The Fig. 2 below shows an example of the data collected on the 3-Ch DC meter. As can be seen in the figure (see Fig. 2), the battery voltage series shows a trend. The Holt's method described previously is ideally suited to predict battery voltage values. This gives the possibility to generate warnings if the predicted value is below the threshold value.

The microcontroller's task is to predict the battery voltage value for the next day based on the data from the five preceding days. In order to do this, the data must be prepared in advance. Each measurement is taken every 10 min. The predicted values will be daily values so in order to standardize the data with those that will be predicted it needs to be averaged beforehand. The data is averaged every 12 h which gives 2 values per day. This data is then stored in an array with the size of 10 elements. When this array is full, the first forecast of two 12 h values is made using the Holt's Linear Trend method. The value from the second forecast value is then checked. If this value is below the battery threshold then a low battery warning is given. Each day, the two oldest elements are removed from the array to make space for the two new values. The data in the array is shifted and the two new average values, calculated from the current day are written to the array. This is followed by a recalculation of the two forecast values for the following day. Then the algorithm is repeated over and over again. This gives a daily voltage forecast for the next day.



Fig. 2. Measurements collected on 3-Ch meter

The Fig. 3 shows a simulated forecast based on data that shows a downward trend. The data reflects a situation where the battery is continuously discharged by the load connected to it. It also represents the fact that battery voltage can be predicted using Holt's Linear Trend Method. The first version of the application was implemented in java programming language and executed on a PC. This was to verify the correctness of the prediction before implementing the algorithm on an embedded device. The plot shows a situation where the predicted voltage is close to 10 V. This would result in a low battery voltage warning. The prediction was made using data from the measuring station that was equipped with an under-sized solar panel on a cloudy day.



Fig. 3. Low voltage warning simulation

4 Conclusions

The whole idea of designing a low battery warning system based on artificial intelligence was to optimize the number of solar panels used in a stand-alone measuring station. Calculating the power demand and selecting the number of solar panels is nothing special, but here comes the aspect of random events such as a few days of bad weather. A warning system provides the user with sufficient time to react and secures the operation of the measuring station. As the implementation process is still underway, there are no calculated mean square errors of the forecast yet. Checking the correctness of the solution is the next task that will be performed. Unfortunately, it takes time to perform the test under natural conditions. Work will be continued directly on the microcontroller, which will give more accurate data on the speed of the proposed algorithm, needed RAM and processing power. After this, the next step will be to implement an algorithm that will use all the data described in the dataset and algorithm section. This will give the possibility to determine the prediction quality of the described solution.

Artificial intelligence is not just about complex systems for object recognition, it's also about simple algorithms implemented on devices that serve simple everyday tasks like collecting data from sensors.

Acknowledgments. This work was supported by the European Regional Development Fund within the BB-PL INTERREG V A 2014-2020 Programme, "reducing barriers - using the common strengths", by project SmartRiver, grant number 85029892 and by project SpaceRegion, grant number 85038043 and by the German government under grant 01LC1903M (AMMOD project). The funding institutions had no role in

the design of the study, the collection, analyses, or interpretation of data, the writing of the manuscript, or the decision to publish the results.

References

- 1. Hurbans, R.: Artificial Intelligence Algorithms, 1st edn. Manning Publications Co., New York (2021)
- Naudé, W., Dimitri, N.: The race for an artificial general intelligence: implications for public policy. AI Soc. 35(2), 367–379 (2019). https://doi.org/10.1007/s00146-019-00887-x
- Huh, J., Seo, Y.: Understanding edge computing: engineering evolution with artificial intelligence. IEEE Access 7, 164229–164245 (2019). https://doi.org/10.1109/ ACCESS.2019.2945338. https://ieeexplore.ieee.org/abstract/document/8861030
- 4. Chatfield, C.: Time-Series Forecasting, 1st edn. Chapman & Hall/CRC, Bath (2000)
- 11 Classical Time Series Forecasting Methods in Python. https:// machinelearningmastery.com/time-series-forecasting-methods-in-python-cheatsheet/. Accessed 10 Sept 2021
- Armstrong, J.S.: Selecting forecasting methods. In: Armstrong, J.S. (ed.) Principles of Forecasting. International Series in Operations Research and Management Science, vol. 30. Springer, Boston (2001). https://doi.org/10.1007/978-0-306-47630-3_16
- Forecasting with a Time Series Model using Python: Part Two. https://www. bounteous.com/insights/2020/09/15/forecasting-time-series-model-using-pythonpart-two/. Accessed 10 Sept 2021

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

