August 20, 2019

AUTOMATIC PRINTER PERSISTENT CONNECTION MANAGEMENT BY MEANS OF MACHINE LEARNING

HP INC

Automatic Printer Persistent Connection Management by means of Machine Learning

We propose a hybrid, long-lived network connection mechanism that alternates between polling and persistent operation modes based on the activity profile of network-enabled printing devices. Our Machine Learning method is able to predict "printing peaks", proactively alternating the connection mode back and forth from polling (for those moments of low activity) to persistent model (for high demanding usage scenarios). This strategy combines the advantages of low consumption, low cost polling connections, and at the same time provides low response times offered by persistent connections. Our solution minimizes costs and maximizes network usage and performance, respecting the availability requirement thanks to its prediction capability.

Besides the fact that our solution was thought for printers, it can also be applied for other devices that require optimized connection management.

Glossary

Features: Data used by the Machine Learning models to learn.

Feature Extraction: The process of extracting features from raw data.

RNN: Acronym for Recurrent Neural Network.

LSTM: Acronym for "Long short-term memory", a modern Machine Learning model based on recurrent neural networks.

1 Introduction

Nowadays, almost every device is connected to the cloud, constantly exchanging information back and forth. Being connected to the cloud allows the printer to receive remote jobs, provided by user through devices such as smartphones or personal computers. When the user starts the job via his smartphone, for example, an application running in the device sends the job to the cloud. Then, the printer connects to the cloud via a persistent connection, e.g. a websocket, in order to get a list of jobs that it needs to print and starts processing them accordingly. One of the issues is that to keep a connection open all the time is very expensive, so in the current solution the printer polls the cloud asking for new jobs, and only when it has jobs to process, it opens a persistent connection to receive and process them.

Company recently determined a performance requirement that specifies that a printer must acknowledge a job and start printing no longer after a maximum elapsed time on N seconds. The problem with the current architecture is that the time to acknowledge the printer that a job is ready to print, open a dedicated connection and receive the job can take more than N giving polling interval configuration, device performance and environmental conditions. If we also consider the polling interval to check for new jobs, it can take even longer. This scenario makes availability requirement impossible to meet, unless the connection is kept alive which, as mentioned before, presents a higher cost.

To attack this problem, our solution proposes a method for managing the printer persistent connection automatically, by applying a Machine Learning method that can predict "printing peaks" and notify the printer about the best network operation mode at a given time, reducing costs and answering the aggressive performance requirements. Therefore, the printer can establish a persistent connection and be ready to receive and operate on job requests that are likely to be issued in advance, once it predicts when they will occur.

2 Description

The proposed method is strongly based on the temporal factor since jobs are processed in specific printers in specific moments of the day and with a given frequency. The cloud currently stores lots of information about jobs printed and the printers themselves, so all this information can be leveraged for the creation of a predictive model. An example is illustrated in Figure 1. Based on the history of jobs printed in a given printer along with the recent history, our model can predict a peak in the number of jobs that are going to be printed followed by a low demand period. Based on such prediction, the cloud can warn the printer (during one of the printer polls) to open a persistent connection for a given time and be ready to receive the jobs demand. On the other hand, if the model predicted a downtime, the cloud can warn the printer to close the connection for the next predicted period of time.

Machine Learning

Feature Extraction

In order to have a good predictive model, we need to extract the features that better represent what we want to achieve as well as perform pre-processing steps on it. For our scenario, we are going to use features that are available on the jobs logs that may be available in cloud and execute procedures to prepare this data. A plethora of information about the jobs is stored, such as printer model, color configuration, resolution, print quality, number of pages, sides (printed on both sides or just one), if a finisher was used, etc. We combine all this information into a single vector and apply data transformation to convert categorical into numerical values and normalize data between 0 and 1, for example, in order to handle all attributes in the same manner, aiming to reduce the "domination" of one attribute over the other. For the normalization, we used z-score technique.

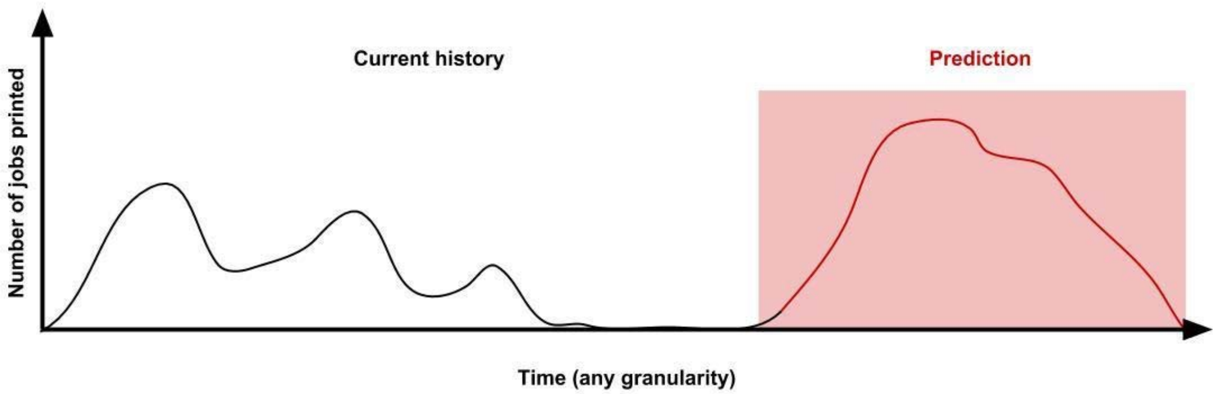

Figure 1. Example of our method. Based on the current history of printed jobs in a given printer for a given period of time, our model predicts a peak in the number of jobs, followed by a fall.

Predictive Model

We employed a state-of-the-art RNN (Recurrent Neural Network) implementation called LSTM (Long Short-Term Memory). After extracting the feature vectors from the raw data series (described in the Feature Extraction section), we feed these feature vectors into the LSTM model, which will output a prediction of the number of jobs the printer might be able to process in the next moment in time (which can be seconds, minutes, hours, etc., depending on the granularity need and based on the feature extraction process that came before). Based on such prediction, we define whether the printer should open or close the persistent connection.

We are not explicitly mentioning here in our data preprocessing pipeline, but an important aspect about time series is non-stationarity, which refers to a time series that varies its statistics (e.g. mean, standard deviation, and variance) over time. Neural networks usually struggle to handle this type of data, so techniques like decomposition of time series can be employed, where the time series is decomposed into different components such as seasonal, trend, and residual (noise). We take such techniques for granted and use them accordingly in our problem.

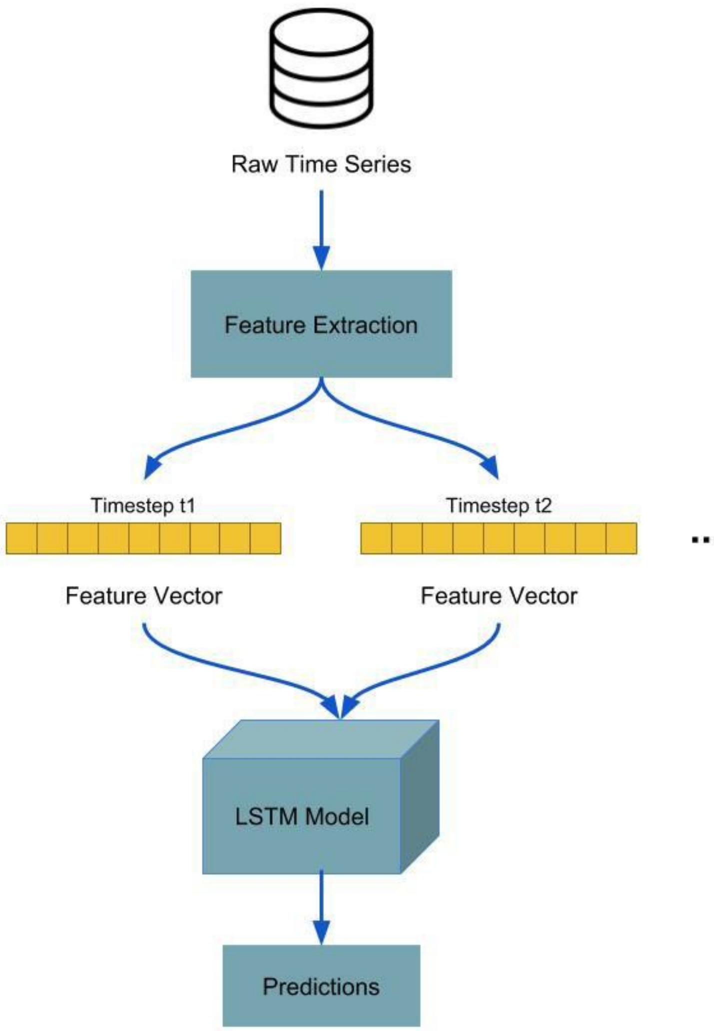

Figure 2. Prediction flow, where the raw time series data (printing history) is transformed into a representation (feature vector), which is fed into the trained LSTM model, which will predict the number of printing jobs in the next timesteps.

Multiple Predictive Models

There are two important factors to consider: different prediction granularity (minutes, hours, months, etc.) and different models (per printer model, per geographic location, etc.). We envision the possibility of having multiple models, where we can configure what decision will be made based on multiple models.

Since the granularity of the prediction can vary (minutes, hours, months, etc.), we foresee a scenario where we could have multiple predictive models, each of which with different granularity. In addition, we also conceive the existence of models available for a given printer family or a given geographic location. The balance between availability and cost is important, so having multiple models can help to reach a more refined decision. In addition, a mechanism with “many-to-many prediction” can be used, where a single model can have multiple outputs.

3 Final Remarks

This method presents many advantages, as follows:

- Smart management of printer resources through the proactive opening of persistent connections (only during peak activity prediction time);
- A better experience (less downtime) when consuming services, reducing the situations where the system could be unavailable;
- Making it possible to meet availability requirement, making the printer available when needed and, at the same time, reducing the “always on” cost;
- LSTM can capture long-term dependencies, which is crucial when dealing with time series data;
- Solution is based on state-of-the-art Machine Learning methods, which helps to achieve high accuracy;
- We can evaluate the prediction over the time and retrain the model with the new data, learning through experience;
- The invention can be extended to different categories of products and different scenarios, such as the connection of IoT devices or personal computers.

Disclosed by Thomas Paula, Juliano Vacaro, Wagston Staehler, and Deny Azzolin, HP Inc.