December 2021

JOB PARSING PREDICTION

HP INC

Job parsing prediction

Abstract

This disclosure is about enabling the print job parsing mechanism to have more intelligence in processing and rendering frequently printed patterns.

Problem statement

Currently the page-description-language parsing for any language doesn't make use of the history of parsing similar files. There are business usecases where same kind of data with minor modifications are printed again and again – such as cheques in banks, forms in educational institutions, invoices in a small-to-medium business etc., This results in a parser doing the same thing repeatedly throughout its lifetime but not improving the parsing and rendering performances.

Solution

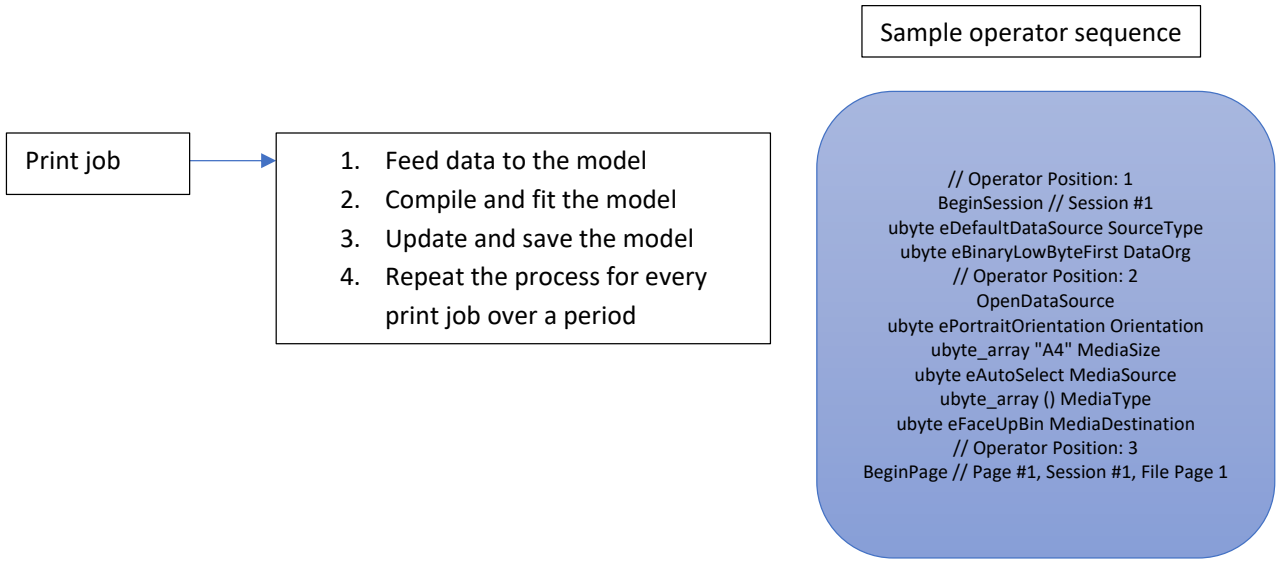
This disclosure proposes that we can use a recurrent neural network, which is trained with a corpus of test files that the parser has encountered within a period. With this, the parser can predict the most frequently used operators and the settings, hence reducing the parsing time significantly. The same can be extended to display list command generation where the rendering engine can auto-predict the next command while generating strips/swaths to be rendered.

Model building and activation

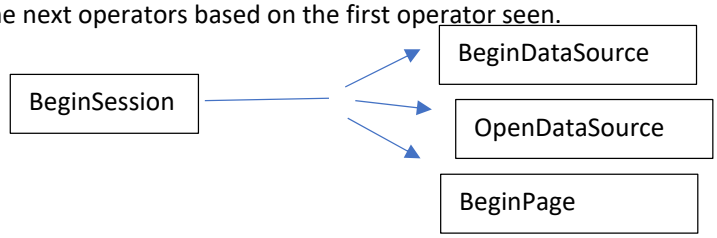
For the first few months of printer deployment, the incoming jobs are used for training the model dynamically. There is no storage of job stream involved in the proposed solution. Once the model is trained with good number of jobs, we can initiate the model activation. At this point, the model starts auto predicting the subsequent operators in the incoming file. With this it ensures the operator parsing is bypassed, appropriate rendering actions are taken, and the display list commands are generated to create the swaths(strips) to the printing engine.

In this solution we are proposing to use the Long-Short-Term-Memory recurrent neural network model for the next operator prediction. LSTM has feedback connections using which it works on inputs coming in a sequence. Using the forget gate, input gate and the output gates, LSTM decides what part of the previously seen input needs to be propagated to the output and what needs to be erased from the cell. LSTM is one of the most used models for next-word prediction (auto-typing, image captioning etc.,) applications.

Below is the basic workflow of the proposed solution



Once the model activation happens, as soon as the print job is received, the model starts predicting the next operators based on the first operator seen.



LSTM parameters:

hidden_nodes: Since the input to our model is a set of words which are formed by ASCII or UTF8 encoded characters, the learnable parameters are not as high as in case of an image or a hand-written character. We can use a standard 256 nodes generally used in word prediction applications.

timesteps: The number of words that generally would determine the next operator for a particular page description language. This can be made configurable based on the job stream.

input_dimension = This would be the dimension of the vector representation of the operator word. This can be made configurable based on the page description language as well. Each language will have unique kind of operators and lengths of these operator words.

Other hyper-parameters that can be tuned to achieve optimal accuracy are – learning_rate, dropout_value, decay, and momentum.

Prior Solutions

No prior solution exists for this problem statement

Advantage

1. Ensures not to waste the printer resources such as CPU, memory etc., on repeated tasks.
2. Ensures faster processing of incoming job stream.

Disclosed by Chaitra S, Debtanu Maji, Mayank Kakkar, HP Inc.