Anomaly Detection Using Deep Learning

UNIVERSITÉ DE LORRAINE — Supervised Project

Institut des sciences du Digital Management & Cognition

Esteban Marquer
Prerak Srivastava

Loria Laboratoire lorrain de recherche en informatique et ses applications

Supervised by:
Christophe Cerisara
Samuel Cruz-Lara

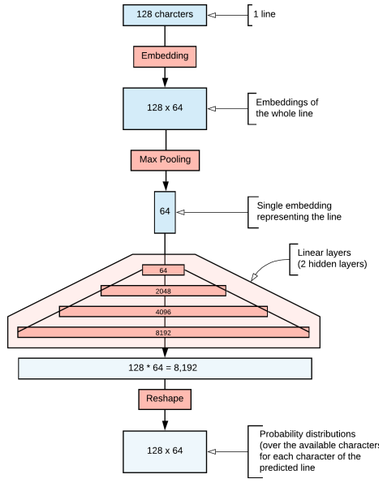
Objective

The general objective of the project is to detect anomalies in log lines.

To do so, we could train a model to detect the anomalies. But the data contains almost no anomalies to train the model.

We need to use an alternative method: predicting the "normal" log lines, and detect anomalies from this "normal" behaviour.

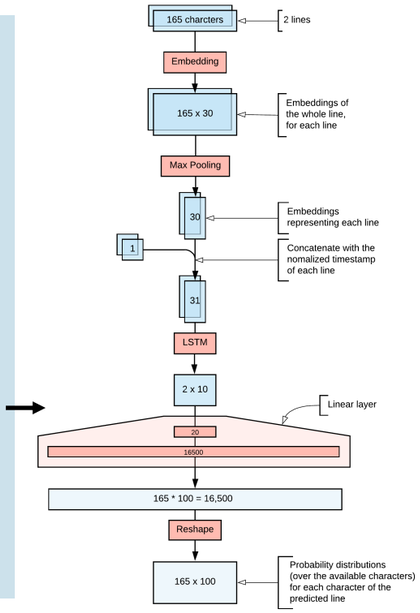
Different Models ...



Two deep learning models were used.

One of them (on the left) is based exclusively on the **Deep Averaging Network (DAN)**. It uses **1 line** to predict the following one.

The other one (on the right) is based on the **DAN** and includes a **Long- and Short-Term Memory network (LSTM)**. It uses **2 lines** to predict the following one.



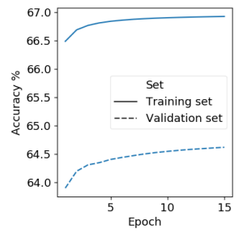
... To Predict The Next Log Line ...

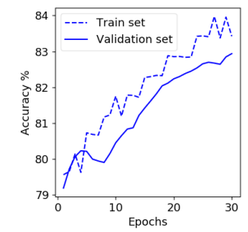
The two models were tested on two dataset, and we evaluated their performance using the **accuracy** of their prediction.

Accuracy: % of correctly predicted characters

DAN using 1 line, trained on the **LANL dataset.**

DAN-LSTM using 2 lines, trained on the **BAREM dataset.**





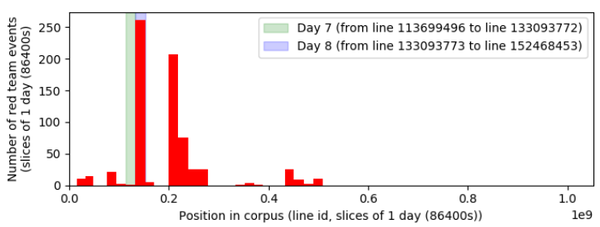
... On Different Datasets ...

LANL dataset

Los Almos National Laboratory cyber-security dataset

A **public dataset** of 58 days of log lines, with **labled anomalies**.

We use the *day 7* to train the models, and use *day 8* to evaluate the detection ability (with AUC ROC).



BAREM dataset

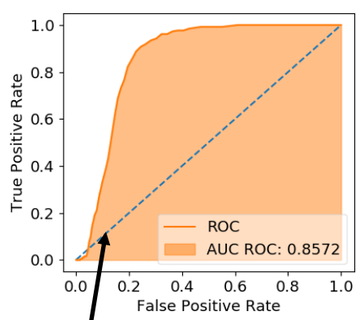
An **industrial dataset** of more than 331,000 log lines, grouped in around 15,000 sessions.

The dataset is undocumented, posess **no labeled anomalies**, and require **heavy preprocessing** before being usable.

Exemple of the content of a log line (fields used are in **green**):

Date	`2017-09-08`
Timestamps (in ms)	`02:28:00,017`
Server name	`CC74.Administration`
Session ID	`wGjQrps5TRpcOnQhdXTnBlzE`
Workflow ID	`14ae7ddb06697d9578dadb77f8b76842`
Port address	`0x7A26D608`
Tag	`START_ACT__`
Executed task	`Home activity`
Stack of the previous tasks	`String of all the previous task`
Entry or exit (*entrée* or *sortie*)	`entree`

... And Detect Anomalies



If the **distance** between the predicted and the true log line is above a **threshold**, the line is an anomaly.

We use the **Area Under the Receiver Operator Characteristic Curve (AUC ROC)** to evaluate the detection ability of the model for a range of detection thresholds.

Worst case (diagonal): the model can't differentiate between anomaly and non-anomaly

TPR (or *probability of detection)*: probability to classify an anomaly as such

FPR (or *false alarm ratio)*: probability to classify non-anomaly as an anomaly