Job problems buster

Sergey Padolski (BNL), Tatiana Korchuganova (MSU)

Jobs problems are dispatched manually

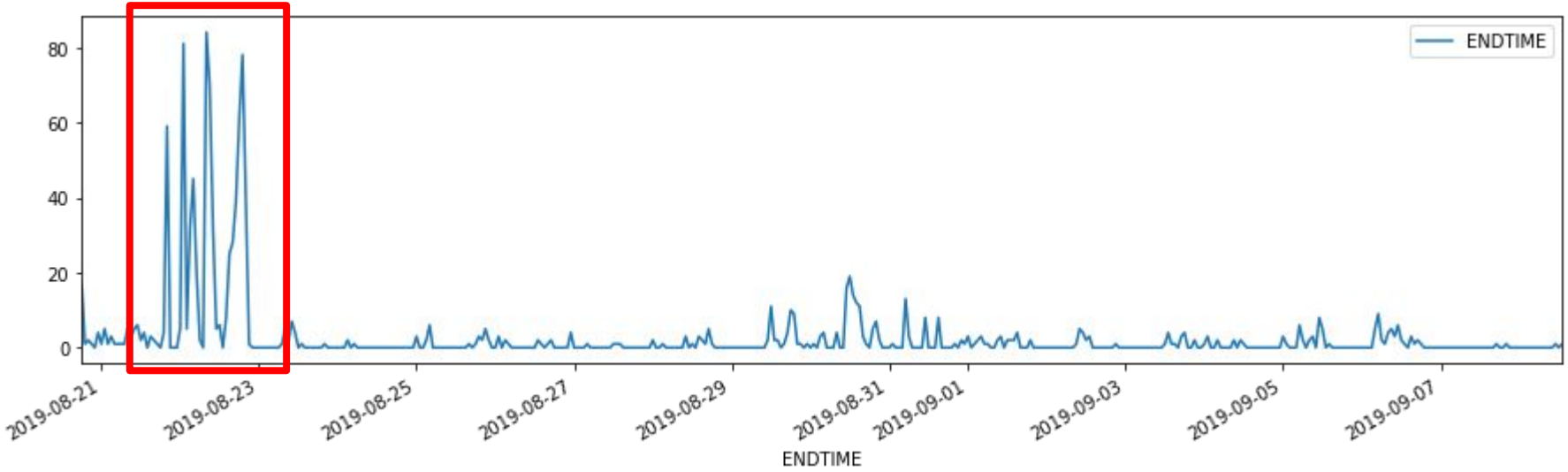
- Poorly scaling
- Time consuming

Vision of automation

- Not yet sharply defined, evolves
- Something about a machinery which spots GRID computing problems and helps people to understand them

Practical start

- A task, is an entity which unifies number of jobs. Some of them fails. There are sporadic failures and significant cases which should be spotted, dispatched and understood.



Jobs failures counts in a task

Initial inputs

- Blind clustering fails, because it would found clusters for processing sites, for jobs creating time, number of cores, etc. Zero interest
- Supervised clustering may also fail because job is described by tenths of features, some of them are categorical.

  Tenths of categorical features -> Hundreds of numerical

  Do we need clusters in O(100) parameter space?

  The point of interest is lay out in few dimensions which are really important and could be understood by human

A prototype approach

- We train a failure forecasting model for this task within time window
  - We don’t use this model to predict anything
  - We extract from this model factors influenced on jobs failures at particular circumstances
- We don’t use the historical data to train the model
- We do train on the fly and build a unique model suitable to understand a particular problem

Numbers

- We use 29 jobs features to build model
- 23 of them are categorical
- CatBoostClassifier forms decision trees which can classify successful and failed cases. Chosen due to many reasons, primary is simple handling the categorial features
- 50 iterations is enough to build quite accurate model (with accuracy of ~90%)
- 0.3s is the training/important analysis time (my work desktop)
- Results are compatible with what experts said in email threads about cases
- We build MVP

PILOTVERSION: 99.86791826943028
INPUTFILEBYTES: 0.13208173056971426
WORKINGGROUP: 0.0
TRANSFORMATION: 0.0
SPECIALHANDLING: 0.0
RESOURCE_TYPE: 0.0
PRODUSERNAME: 0.0
...

Minimal Viable Product

- A stand alone, REST application
- Fed by DB entries content supplied in JSON. Could be anonymized if needed. Logs and related stuff later
- Provides influencing factors and values to look at in JSON
- First adopter is the Atlas BigPanDA monitoring which will display a failures clusters on task page, accompany and then replace the eye catching jobs issues analysis
- Could be developed and brought to production in different fashion, could be a collaborative project
- We are open for suggestions from potential users/contributors