Yandex

Yandex
CatBoost

Introducing CatBoost Spark

Andrei Khropov,
CatBoost team @ Yandex

Plan

- What is CatBoost?
- Why CatBoost?
- Why CatBoost Spark?
- CatBoost Spark and Competitors

What is CatBoost?

Yandex
CatBoost

- CatBoost is a machine learning algorithm that uses gradient boosting on decision trees (GBDT).
- It is available as an open source library.

Data at hand

Unstructured data

Music

DNA

Images

Text

End2End with Deep NN

Tabular (or structured) data

Well engineered features

Music track length	Year	Rating	Label
2	1990	3	1
3	1950	5	0
15	1970	4	1

GBDT

Tabular data? Decision trees!

Gradient boosted decision trees

- State-of-the-art quality on tabular data
- Easy to use, no sophisticated parameter tuning
- Works well with small data and scales for big data problems

Why CatBoost?

Yandex
CatBoost

- Sophisticated feature types support
- Training with pairs
- Good quality with default parameters
- Fast applier
- Extensive support of model formats
- Model analysis tools
- Stable, Production quality

Features

Numerical

Categorical

Text

Embeddings

Yandex
CatBoost

Features

Numerical

Categorical

Text

Embeddings

Length: 0.25
Weight: 57

Yandex
CatBoost

Features

Yandex CatBoost

Numerical

Categorical

Text

Embeddings

Gender: Male
Profession: Scientist
Country: Switzerland

Features

Yandex CatBoost

Numerical

Categorical

Text

Title: "CatBoost Spark"
Description:
"CatBoost is a fast, scalable, high performance gradient boosting on decision trees library."

Embeddings

Features

Yandex CatBoost

Numerical

Categorical

Text

Embeddings

Book name:	"War and peace"	"Lord of the Rings"
Embedding:	[0.1, 2.3, 0.0]	[0.3, 0.0, -4.2]

Model formats

Yandex CatBoost

- CatBoost Native
- CoreML
- ONNX
- PMML
- Raw code:
  - C++
  - Python

Model analysis

Yandex CatBoost

- Example with SHAP values (uses ‘shap’ python package)
- From https://github.com/catboost/tutorials/blob/master/model_analysis/shap_values_tutorial.ipynb

CatBoost in production

› **Yandex.Search**

Big datasets up to several Terabytes

Multi-host multi-GPU training

… many other users inside Yandex

CatBoost outside Yandex

- Recommendations at Netflix
- Hotel ranking in Aviasales
- Protection against bots in CloudFlare
- Particle classification in CERN
- Medical research at University of NSW Sydney
- Destination prediction in Careem taxi service
- ML competitions on Kaggle
- Join our Awesome Users List! ☺

Why CatBoost Spark?

Yandex CatBoost

APACHE Spark™

Why CatBoost Spark?

Yandex CatBoost

APACHE Spark™

- When data size is too big for a single machine

Why CatBoost Spark?

Yandex CatBoost

APACHE Spark™

- When data size is too big for a single machine
- Processing speed

Why CatBoost Spark?

Yandex CatBoost

APACHE Spark™

- When data size is too big for a single machine
- Processing speed
- When you need the flexibility of working with data on Spark

Why CatBoost Spark?

Yandex CatBoost

APACHE Spark™

- When data size is too big for a single machine
- Processing speed
- When you need the flexibility of working with data on Spark
- Low-level data processing in Spark Scala/Java API vs faster than Python API

Why CatBoost Spark?

Yandex CatBoost

APACHE Spark™

- When data size is too big for a single machine
- Processing speed
- When you need the flexibility of working with data on Spark
- Low-level data processing in Spark Scala/Java API vs faster than Python API
- CatBoost model training and analysis with JVM platform API (PySpark is also supported)

Why CatBoost Spark?

Yandex CatBoost

APACHE Spark™

- When data size is too big for a single machine
- Processing speed
- When you need the flexibility of working with data on Spark
- Low-level data processing in Spark Scala/Java API vs faster than Python API
- CatBoost model training and analysis with JVM platform API (PySpark is also supported)
- API is fully compatible with Spark ML library

API is fully compatible with Spark ML library

```
from pyspark.sql import SparkSession

sparkSession = (SparkSession.builder
    .master("local[*]")
    .config(
        "spark.jars.packages",
        "ai.catboost:catboost-spark_2.4_2.12:0.25-rc4"
    ).getOrCreate()
)

import catboost_spark

trainDf = sparkSession.read().load("/my_datasets/train")

classifier = catboost_spark.CatBoostClassifier(iterations=20)
classifier.write().save("/my_classifier")

model = classifier.fit(trainDf)
model.write().save("/my_model")

applyDf = sparkSession.read().load("/my_datasets/for_application")

dfWithPredictions = model.predict(applyDf)
```

CatBoost Spark vs Competitors

	CatBoost	LightGBM	dmlc XGBoost
GPU support	No (planned)	No	Yes
Categorical features	Yes, including CTR statistics	Yes	Only preprocessed as one hot
Pairs	Yes	No	No
Pre-quantization	Yes	No	No
PySpark support	Yes	Yes	No
SparkR support	No	Beta	No

Feature quantization

CatBoost Spark Performance vs Competitors

	Criteo derived 170 m samples 65 features		Epsilon 400 k samples 2000 features		Higgs 10.5 m samples 28 features	
	total	per iter	total	per iter	total	per iter
CatBoost	53 m 52 s	2.8 s	1 h 5 m	3.7 s	7 m 40 s	0.31 s
LightGBM	59 m	3.5 s	2 h 36 m	9.4 s	2 h 25 m	8.7 s
dmlc XGBoost	est. 8 h	28.9 s	17 m 25 s	1 s	10 m	0.6 s

- Cluster configuration – 16 nodes x 16 cores
- Time on 1000 iterations, total time includes preprocessing

The End

- CatBoost website: https://catboost.ai/
- CatBoost documentation: https://catboost.ai/docs
- CatBoost on GitHub: https://github.com/catboost
- CatBoost Spark home: https://github.com/catboost/catboost/tree/master/catboost/spark/catboost4j-spark