

Spark MLIB

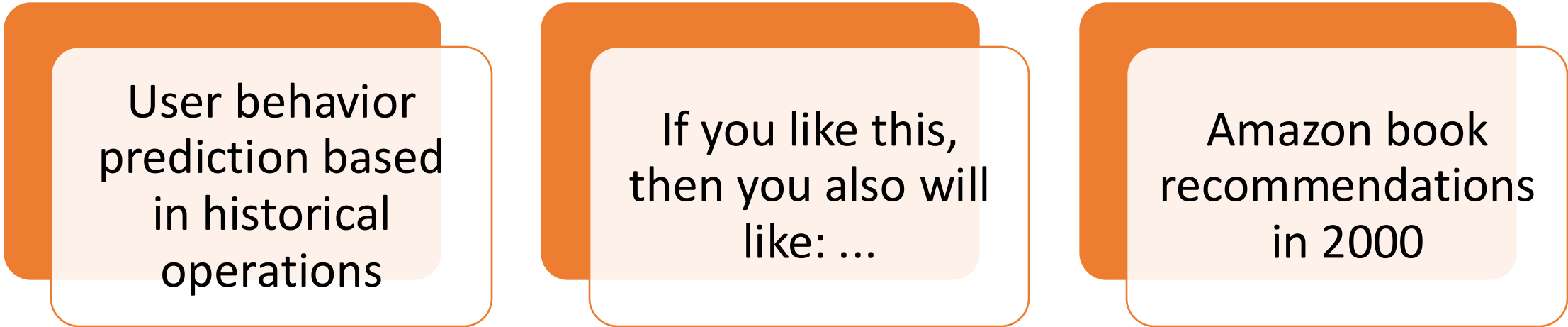
Recommender systems and ML pipelines



Scheduling of sessions – Spark MLIB

- 15/5 – Introduction to Spark MLIB
- 22/5 – [Spark MLIB lab 1](#)
- 26/5 – [Spark MLIB lab 2](#)
- 29/5 – [Spark MLIB lab 3](#)

Recommender systems



User behavior
prediction based
in historical
operations

If you like this,
then you also will
like: ...

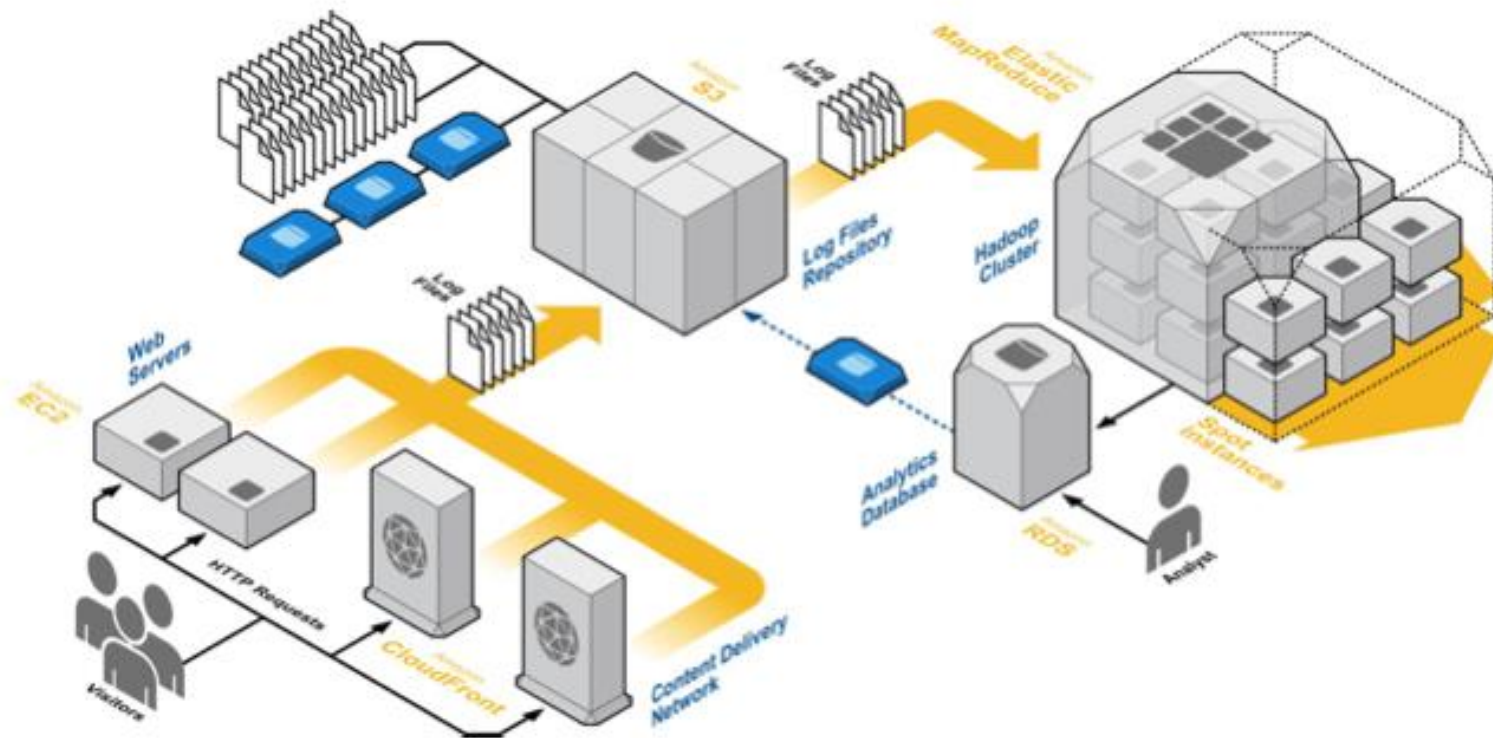
Amazon book
recommendations
in 2000

Data collected from user actions

- Amazon books is a trail of user visits to be correlated with user behavior
- Users are in a feedback loop in which they affect the products in use
- Companies: How to use tracking info effectively?



Recommending system cloud architecture



Search team at google
April 22nd 2001

“Carol Brady maiden name” at the top of the queries charts

Search team at google

April 22nd 2001

“Carol Brady” searches grouped in five peaks:

1. biggest
2. small
3. small
4. big
5. and finally, after a long wait, another small blip

Each peak started at 48 minutes after the hour



Why? What does the data mean?

Why would there be a new interest in a character from the 1970's sitcom "The Brady Bunch"?

Repeated five times during the day?


Why? What does the data mean?



Why would there be a new interest in a character from the 1970's sitcom "The Brady Bunch"?



You can't interpret it unless you know what else is going on in the world



What happened
April 21st 2001?





Who wants to be millionaire?

- That night the million-dollar question of TV show was:
 - "What was Carol Brady's maiden name?"
- Seconds after the show's host, posed the question, thousands made a Google search for the answer
- Google search spikes produced four spikes as the show was broadcast successively in each US time zone

Search team at google

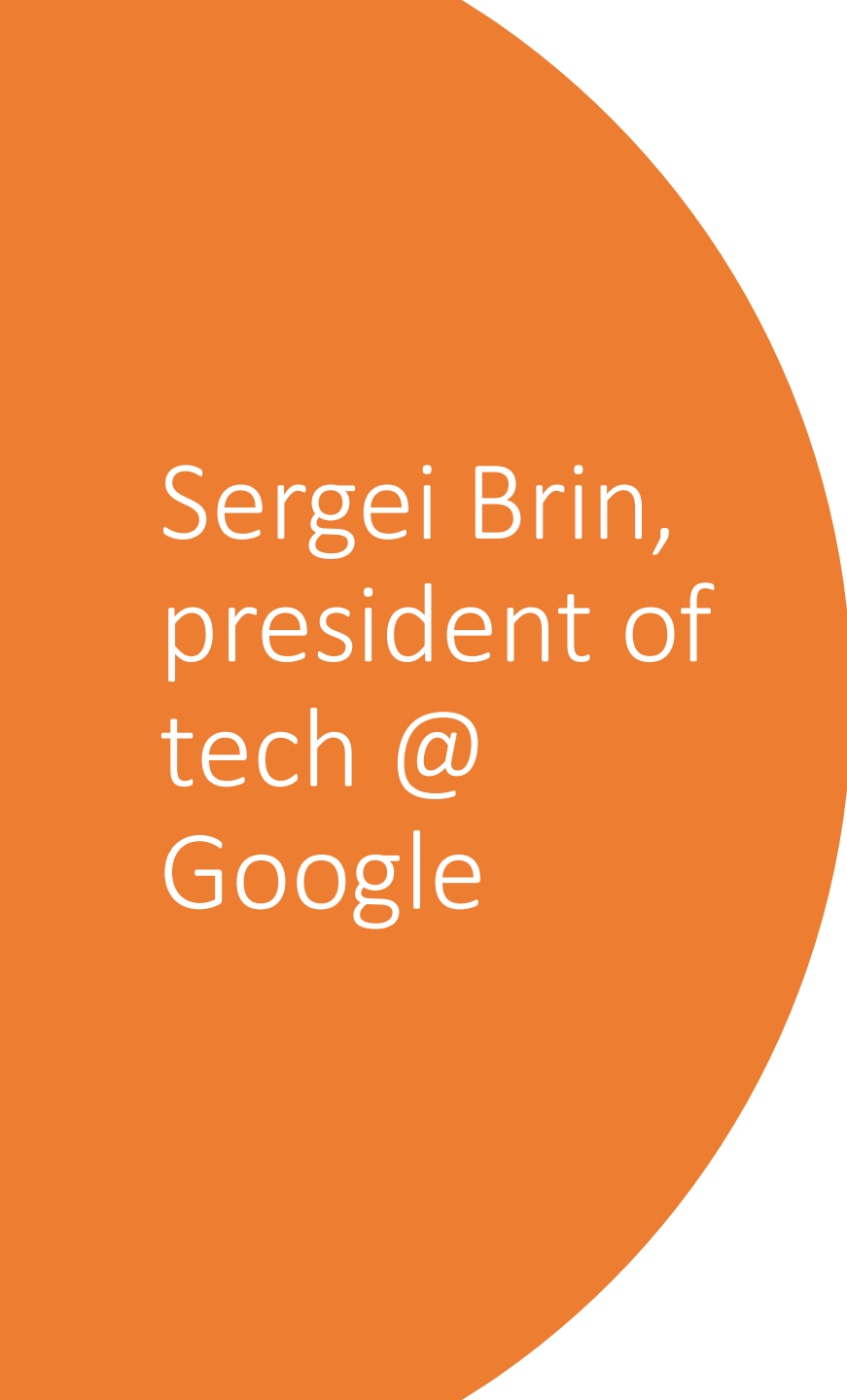
April 22nd 2001

“Carol Brady” searches grouped in five peaks:

1. biggest
2. small
3. small
4. big
5. and finally, after a long wait, small blip



Each peak started at 48 minutes after the hour

A large orange circle on the left side of the slide, partially cut off by the edge.

Sergei Brin,
president of
tech @
Google

- "It was like trying an electron microscope for the first time"
- "It was like a moment-by-moment barometer"
- **Google's strengths is its predictive power**, it can measure global trends before any other media know



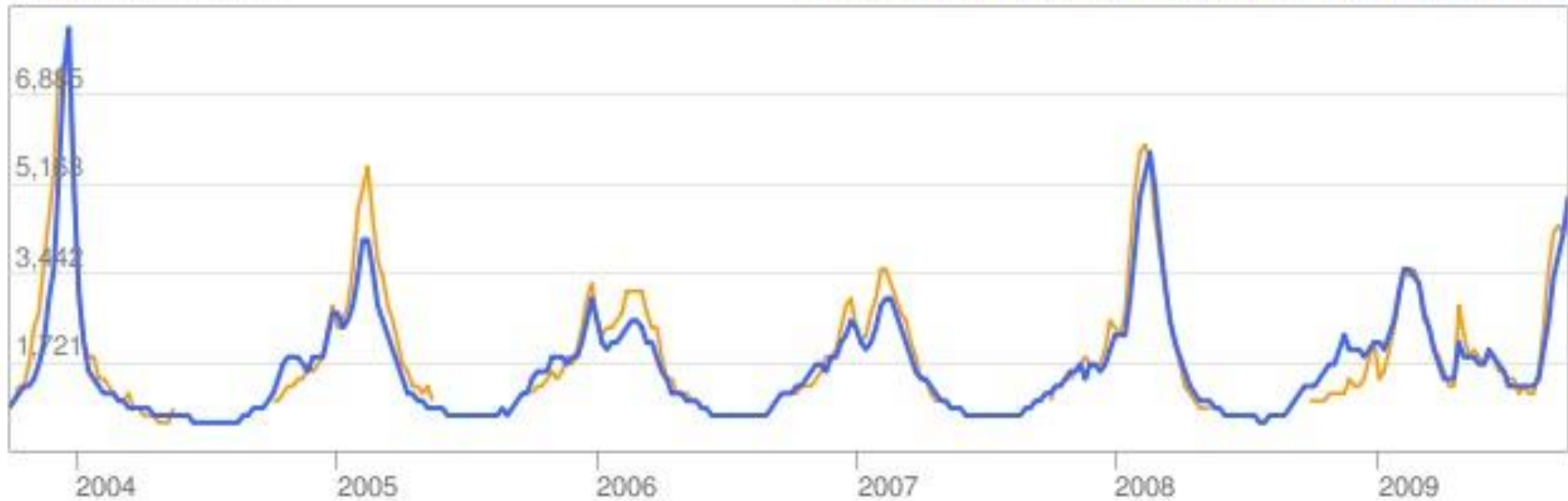
Google Flu prediction 2003-2008

Estimation of flu activity vs
Centre for disease control real
data

United States Flu Activity

Influenza estimate

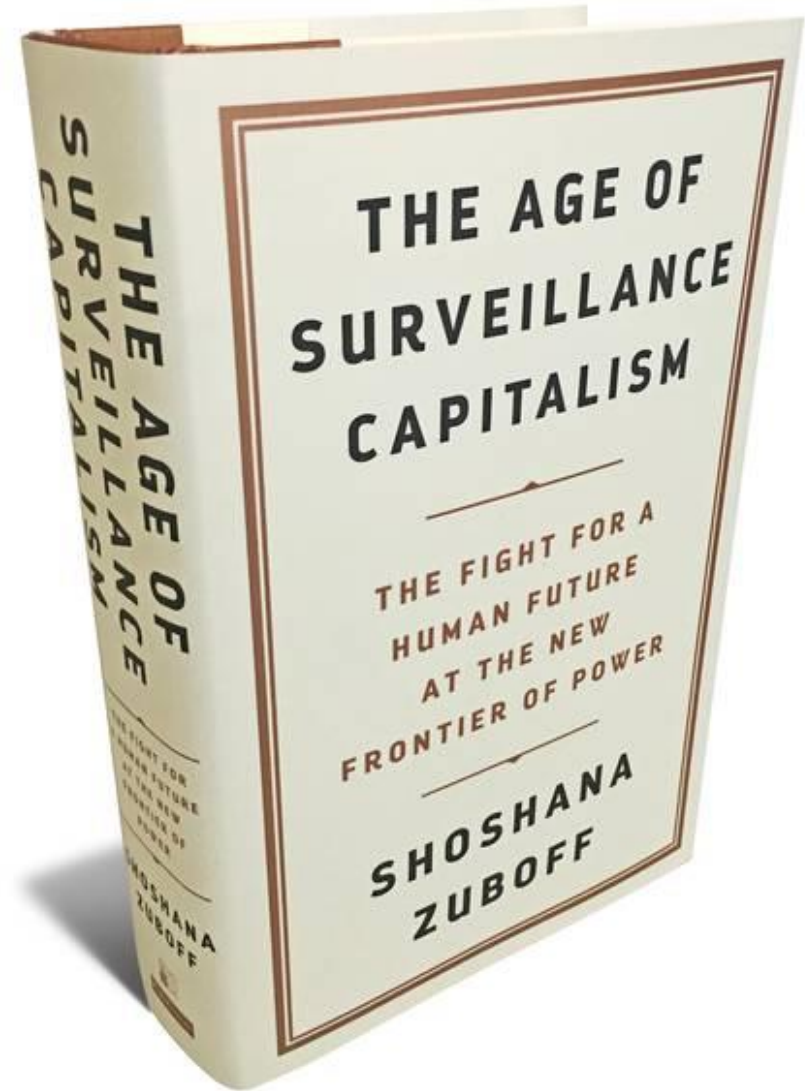
● Google Flu Trends estimate ● United States data



United States: Influenza-like illness (ILI) data provided publicly by the [U.S. Centers for Disease Control](#).

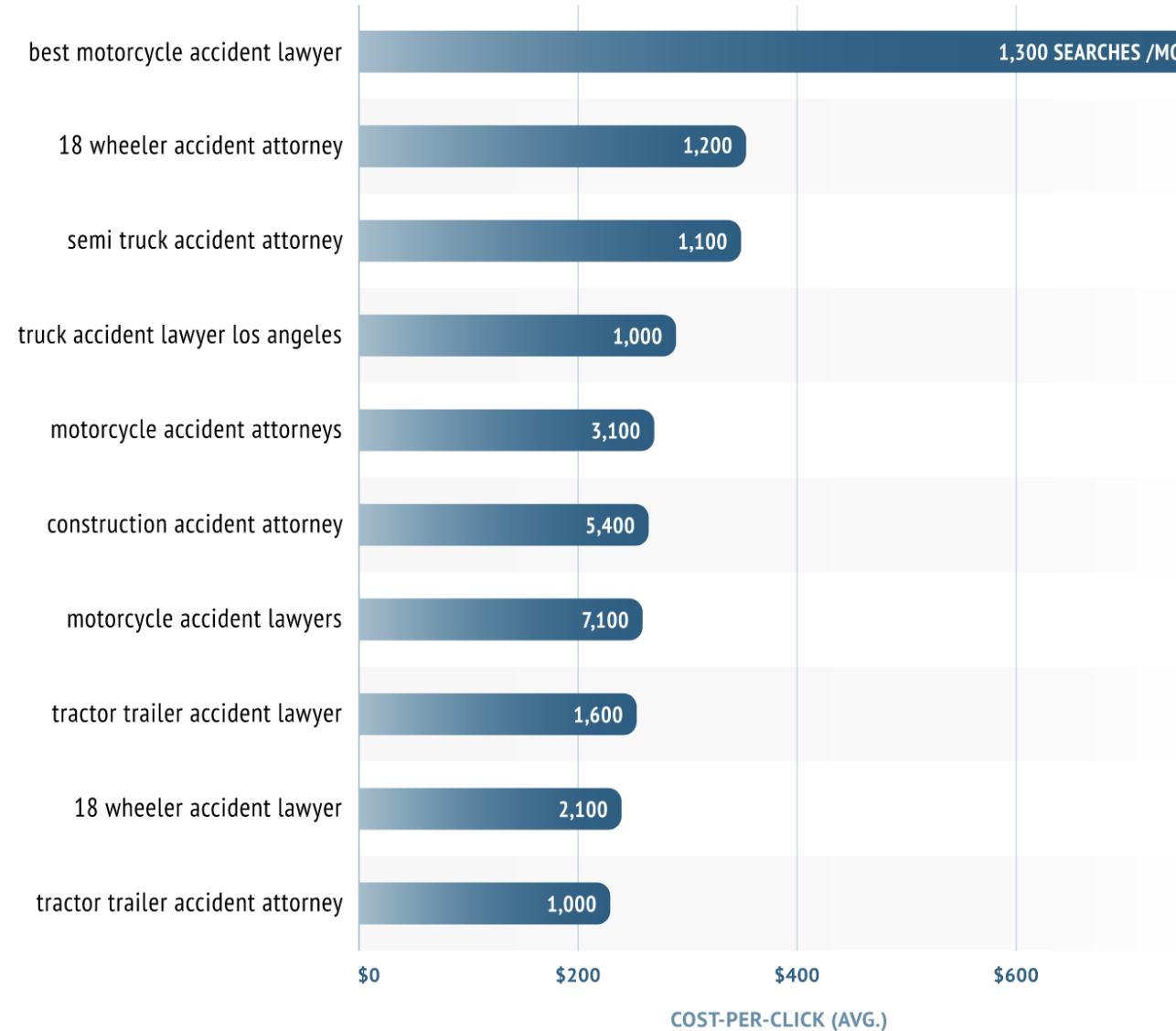
Digital events
predict(ed) real life

Search engines become
real-time word
auction marketplaces



SEARCH KEYWORDS ON GOOGLE

BASED ON AVERAGE COST-PER-CLICK PER KEYWORD
IN THE US (1,000 SEARCHES /MO. MIN.)



Is this true after GPT models?

Google

medieval illustration frog

Imágenes Shopping Videos Libros Más Herramientas

Sugerencia: Limita esta búsqueda a resultados en **español**. Más información sobre cómo filtrar por idioma

Imágenes : cute frog oil painting twitter openart toad illuminated manuscript

medieval painting of... OpenArt

weird medieval guys ... X.com

medieval painting of... OpenArt

Medieval Bestiary : Frog G... Pinterest

prompthunt: mediev... Prompt Hunt

Frog renaissance art portrait, m... Adobe Stock

Ca. 1197. Plague of frogs |... Pinterest

A medieval illustratio... OpenArt

weird medieval guys BOO... X.com

Weird Medieval Art ~... Facebook

Medieval painting of a ... Craiyon

Isolated cute frog with cr... Vecteezy

Sugerencias

Recommenders as data processes

Elements:

- Machine Learning algorithms
- ML models
- Results of applying model to a dataset

Process:

- data storage
- data processing
- data science
- data engineering

Common use of recommenders

- e-commerce: products and users
- Marketing campaigns
 - Empty basket analysis
 - E-mail campaigns
- Content: spotify, netflix, tiktok
- Prediction: cybersecurity

Frequently Bought Together

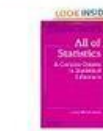


Price For All Three: **\$258.02**

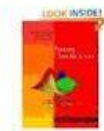
[Add all three to Cart](#)

- ☒ **This item:** [The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition \(Springer Series in Statistics\)](#) by Trevor Hastie
- ☒ [Pattern Recognition and Machine Learning \(Information Science and Statistics\)](#) by Christopher M. Bishop
- ☒ [Pattern Classification \(2nd Edition\)](#) by Richard O. Duda

Customers Who Bought This Item Also Bought



[All of Statistics: A Concise Course in Statist...](#) by Larry Wasserman
★★★★☆ (8) \$60.00



[Pattern Classification \(2nd Edition\)](#) by Richard O. Duda
★★★★☆ (27) \$117.25



[Data Mining: Practical Machine Learning Tools an...](#) by Ian H. Witten
★★★★☆ (29) \$41.55



[Bayesian Data Analysis, Second Edition \(Texts in...](#) by Andrew Gelman
★★★★☆ (10) \$56.20



[Data Analysis Using Regression and Multilevel /...](#) by Andrew Gelman
★★★★☆ (13) \$39.59

Commission opens formal proceedings against TikTok on election risks under the Digital Services Act

PAGE CONTENTS

Top

Quote(s)

Related topics

Print friendly pdf

Contacts for media

Commission opens formal proceedings against TikTok on election risks under the Digital Services Act

Today, the Commission has opened formal proceedings against TikTok for a suspected breach of the [Digital Services Act \(DSA\)](#) in relation to TikTok's obligation to properly assess and mitigate systemic risks linked to election integrity, notably in the context of the recent Romanian presidential elections on 24 November.

Commission President, Ursula **von der Leyen**, said: *“We must protect our democracies from any kind of foreign interference. Whenever we suspect such interference, especially during elections, we have to act swiftly and firmly. Following serious indications that foreign actors interfered in the Romanian presidential elections by using TikTok, we are now thoroughly investigating whether TikTok has violated the Digital Services Act by failing to tackle such risks. It should be crystal clear that in the EU, all online platforms, including TikTok, must be held accountable.”*

The proceedings will focus on management of risks to elections or civic discourse, linked to the following areas:

Recommenders objectives

Help users to discover information relevant to them



Must apply methods to input events:

item to sell

action to do

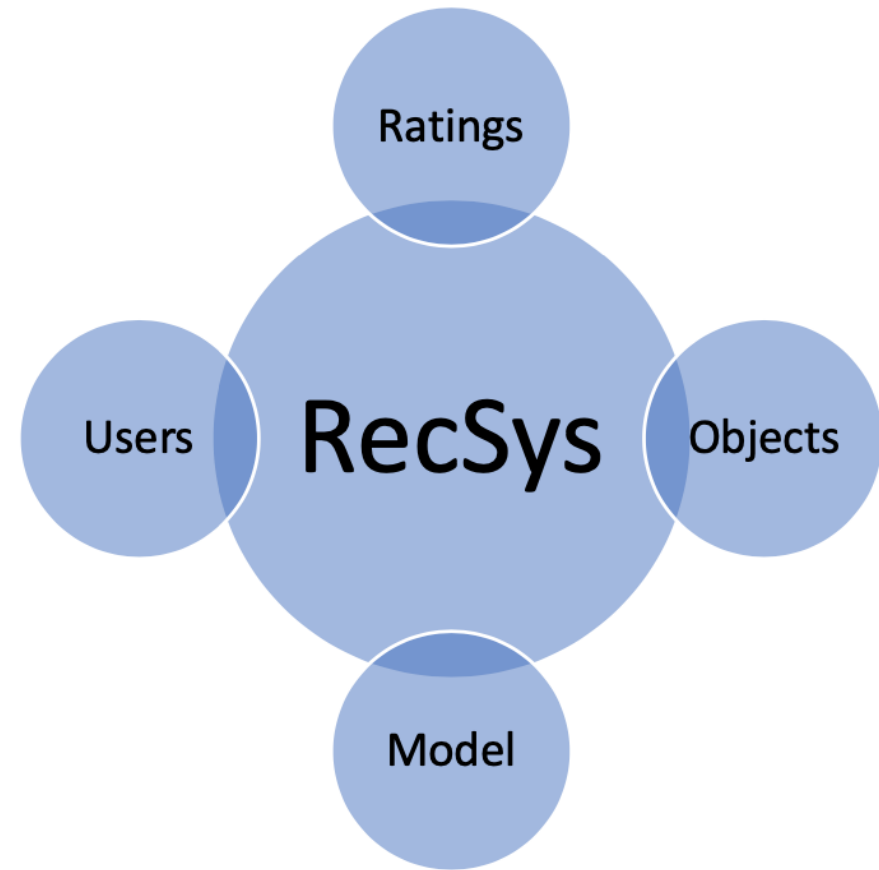
notification to read



Recommender systems architecture

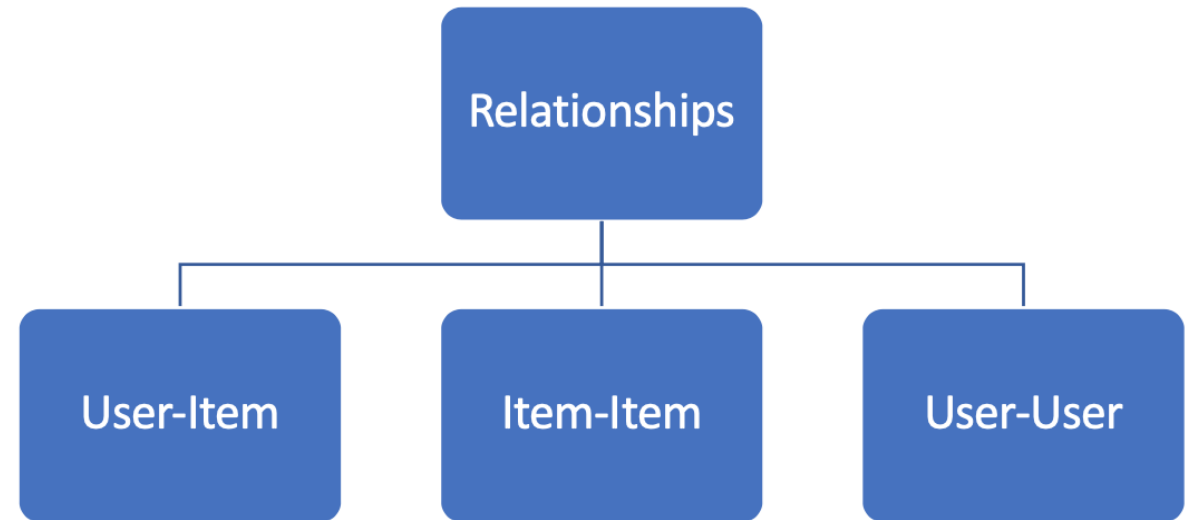
Recommender system architecture

- **Users:** information on user behavior: actions, preferences
- **Objects:** products, items, documents, transactions
- **Ratings:** quantification of user/object events
- **Model:** behavior prediction from unrated user/object events



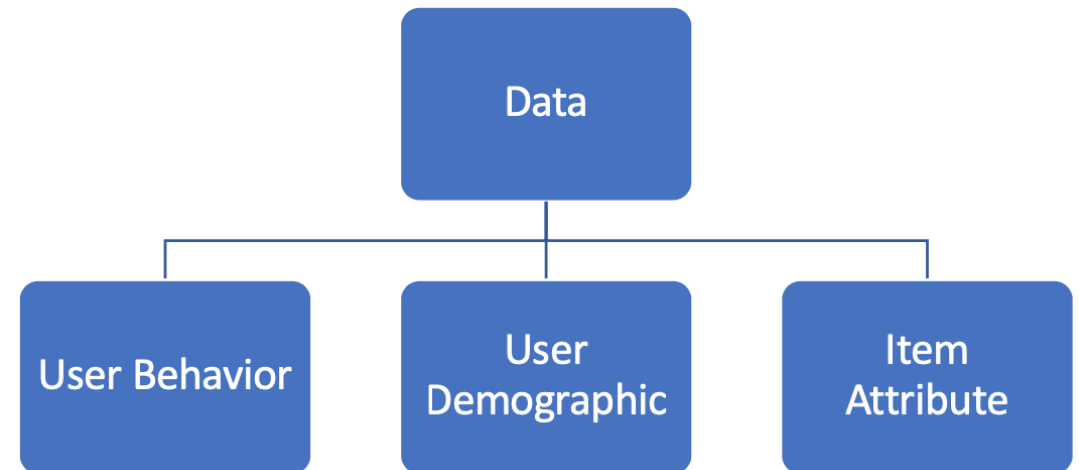
Algorithm types

- **User-item:** item affinity or preference
 - item purchased, song played
- **Item-item:** similarity of objects
 - Same-style song
 - Compatible ink cartridge
- **User-user:** personal similarity
 - Same language/country
 - Mutual connections in social media



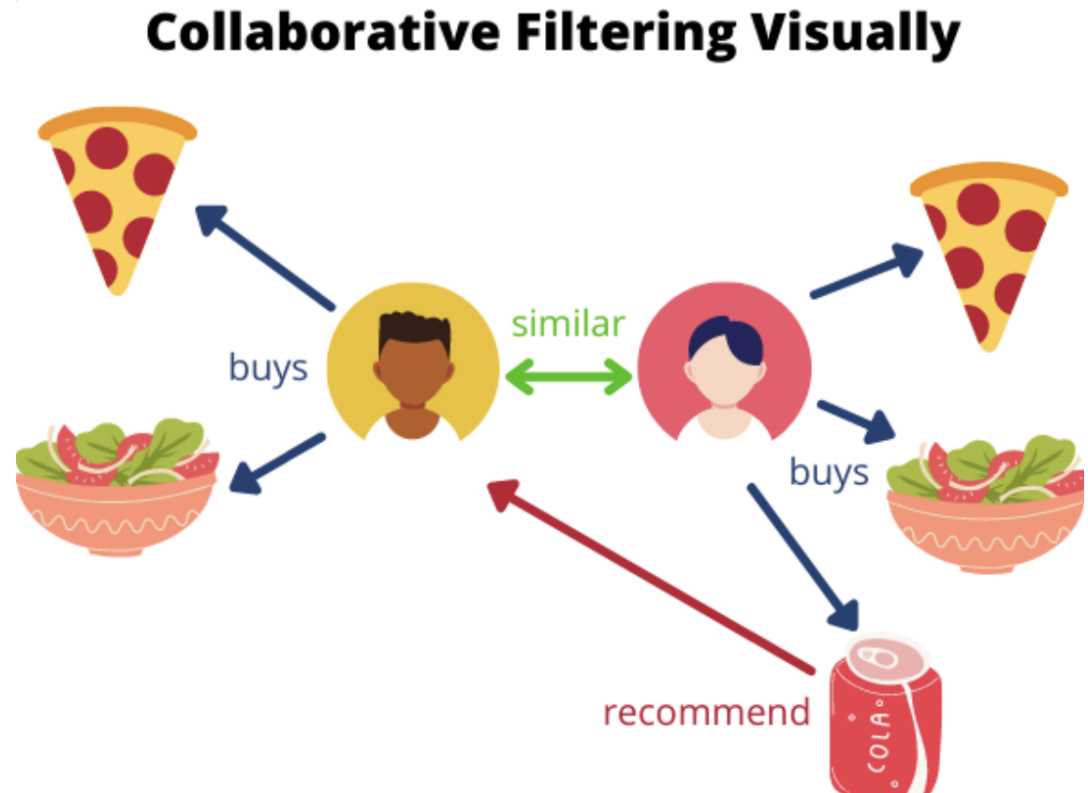
Information types for recommender systems

- **User behavior:**
 - user history described with objects
 - playlists, purchase history, cookies
- **User demographics:**
 - users personal information
 - Spotify top10 most popular today
- **Item attributes:**
 - Information on item
 - Artist name, genre



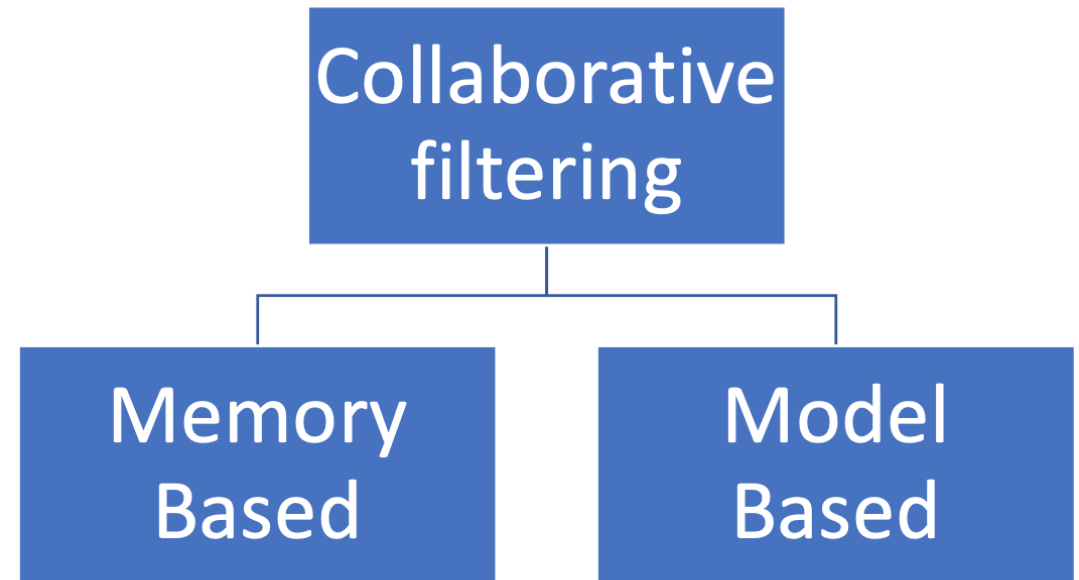
Most common techniques

- **Collaborative filtering:**
 - past interactions define future
 - user-item interaction matrix
- **Content-based filtering**
 - Products, documents, actions, ...



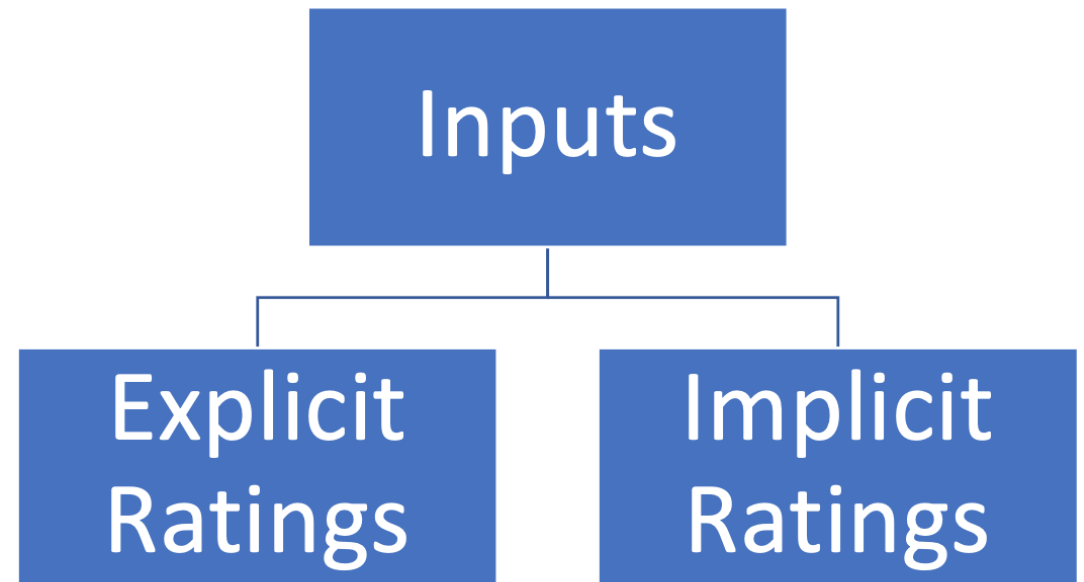
Recommender models

- **Memory:** based on past interactions
 - Assumes no model; usually nearest neighbors search
 - “Customers who viewed this also viewed those”
- **Model:** based on underlying generative model
 - Model explains user-item interactions



User/object rating systems

- **Explicit:** directly from the user
 - Star ratings, reviews, feedback, etc.
 - Netflix, Instagram, and Spotify
- **Implicit:** derived from interactions
 - Examples: clicks, view, and purchases



Challenges of recommenders

Explicit ratings can be
hard to process

Cold start problem:
how do you start
with few users?

Scalability

pandora (content-based) vs
spotify (collaborative filtering)

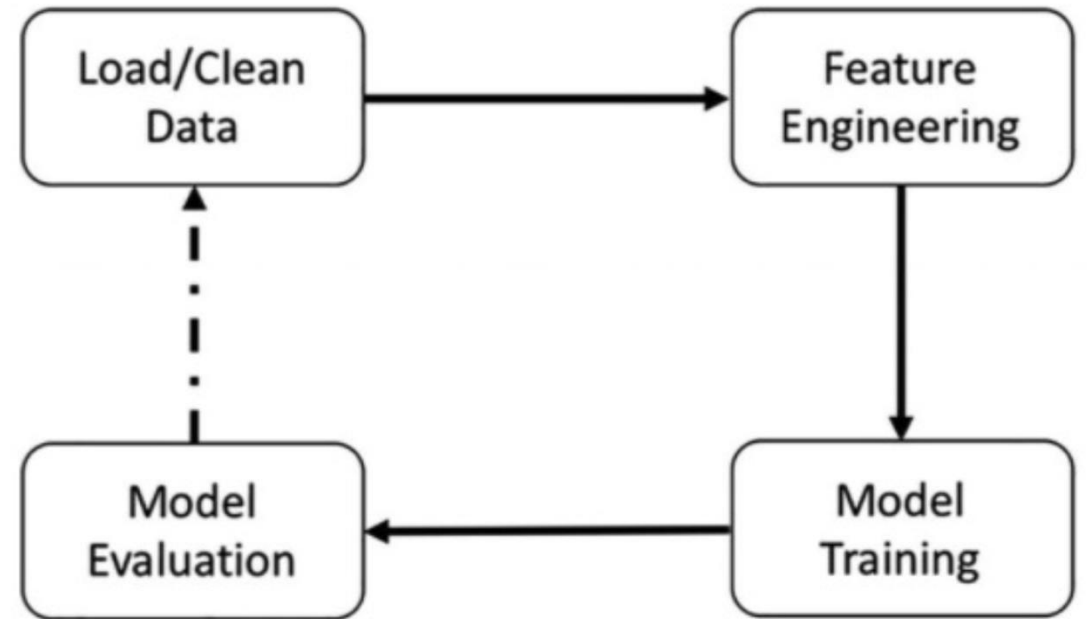


Machine Learning with Spark

Learning Spark, second edition, O'Reilly, 2020

ML data processing main steps

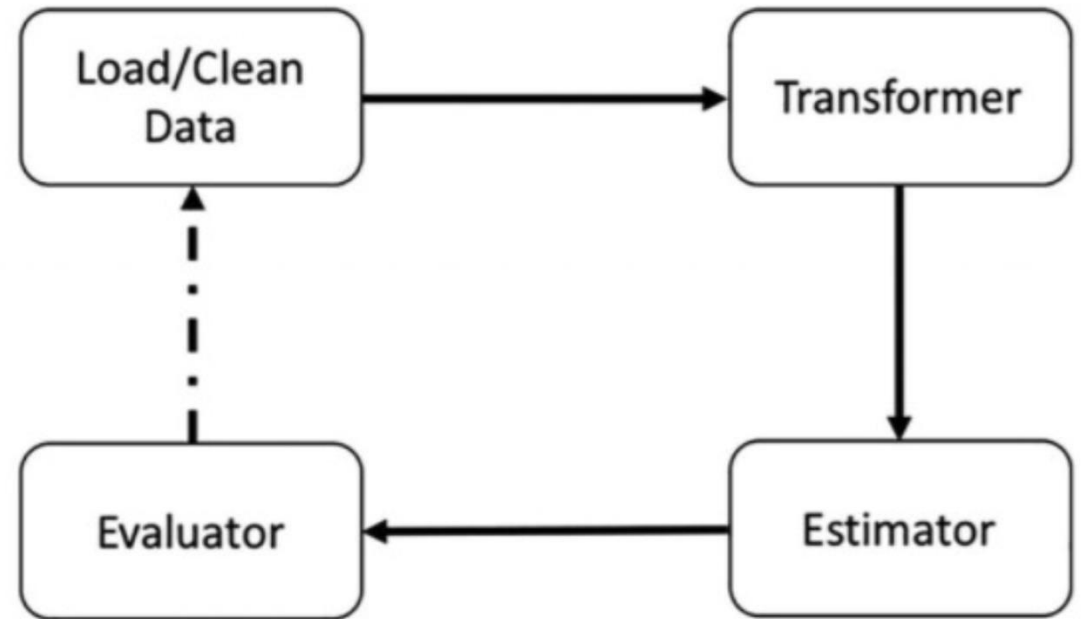
- Load data
- Clean data
- Feature engineering
- Model training
- Model evaluation



ML Main Steps

Apache Spark MLlib main concepts

- Load data
- Clean data
- Transformer
- Estimator
- Evaluator



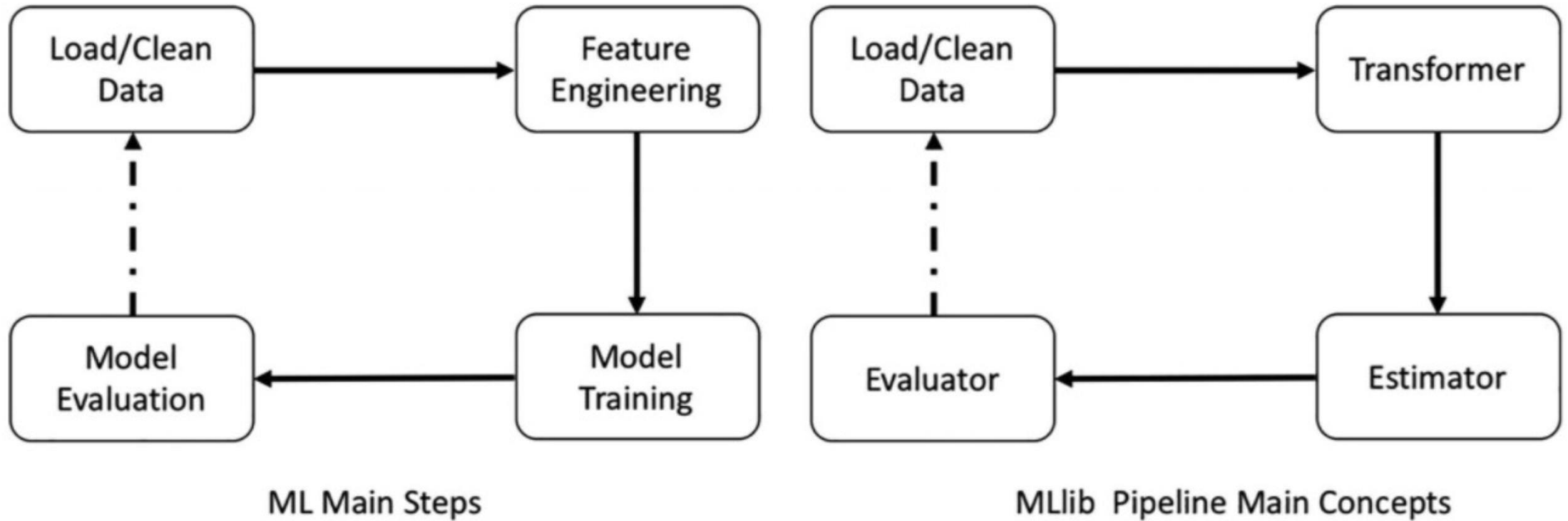
MLlib Pipeline Main Concepts

Apache Spark MLIB main concepts

- Load data
- Clean data
- Feature engineering
- Model training
- Model evaluation

- Load data
- Clean data
- Transformer
- Estimator
- Evaluator

ML vs Spark MLlib



Supervised learning

- Set of input records, each of which has associated labels
- Goal: predict the output label(s) given a new unlabeled input
- Output labels can either be discrete or continuous
- Two types: classification and regression



Supervised learning with Spark

- historical data with labels: dependent variables
- must train a model
- predict the values of those labels
- based on various features of the data points
- Example: predict a person's income based on age
 - Dependent variable: income
 - Feature: age

Training a ML model



Model **Training process** usually proceeds through an iterative optimization algorithm such as gradient descent



The training algorithm starts with a basic model and gradually improves it by adjusting various internal parameters (**coefficients**) during each training iteration



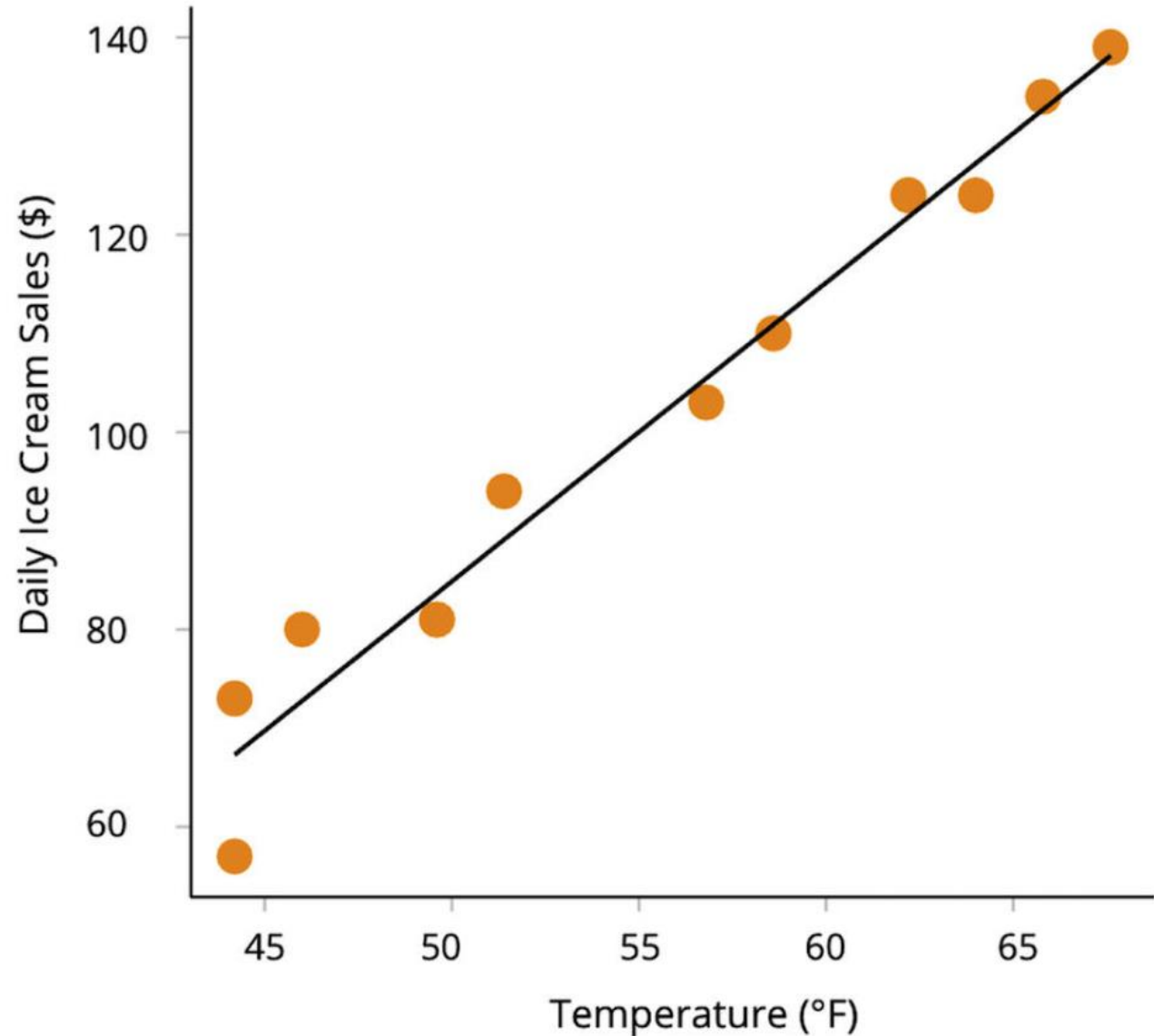
The result of this process is a **trained model** that you can use to make predictions on new data.



Measure the success of trained models: **metrics**

Regression problems

- The value to predict is a continuous number
- Predict values that your model hasn't seen during training



Classification

Training an algorithm to predict a dependent variable that is **categorical** (belonging to a discrete, finite set of values)

Classification model: will make a prediction that a given item belongs to one of two groups

Example: classifying email spam. Using a set of historical emails that are organized into groups of spam emails and not spam emails

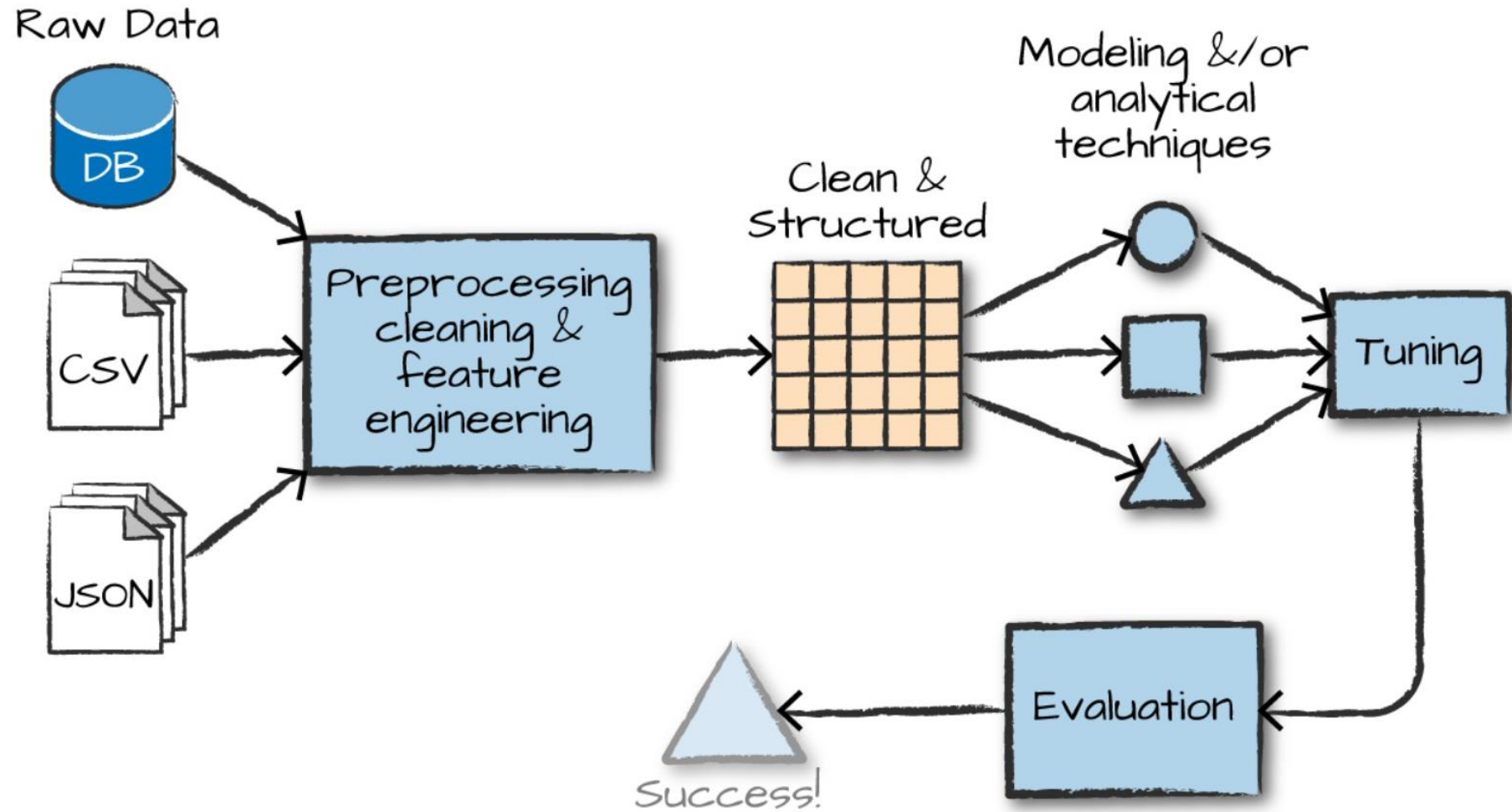
Spark ML available algorithms

ALGORITHM	USAGE
Linear regression	Regression
Logistic regression	Classification
Decision trees	Regression/Classification
Gradient boosted trees	Regression/Classification
Random forests	Regression/Classification
Naive Bayes	Classification
Support vector machines	Classification

Recommender systems

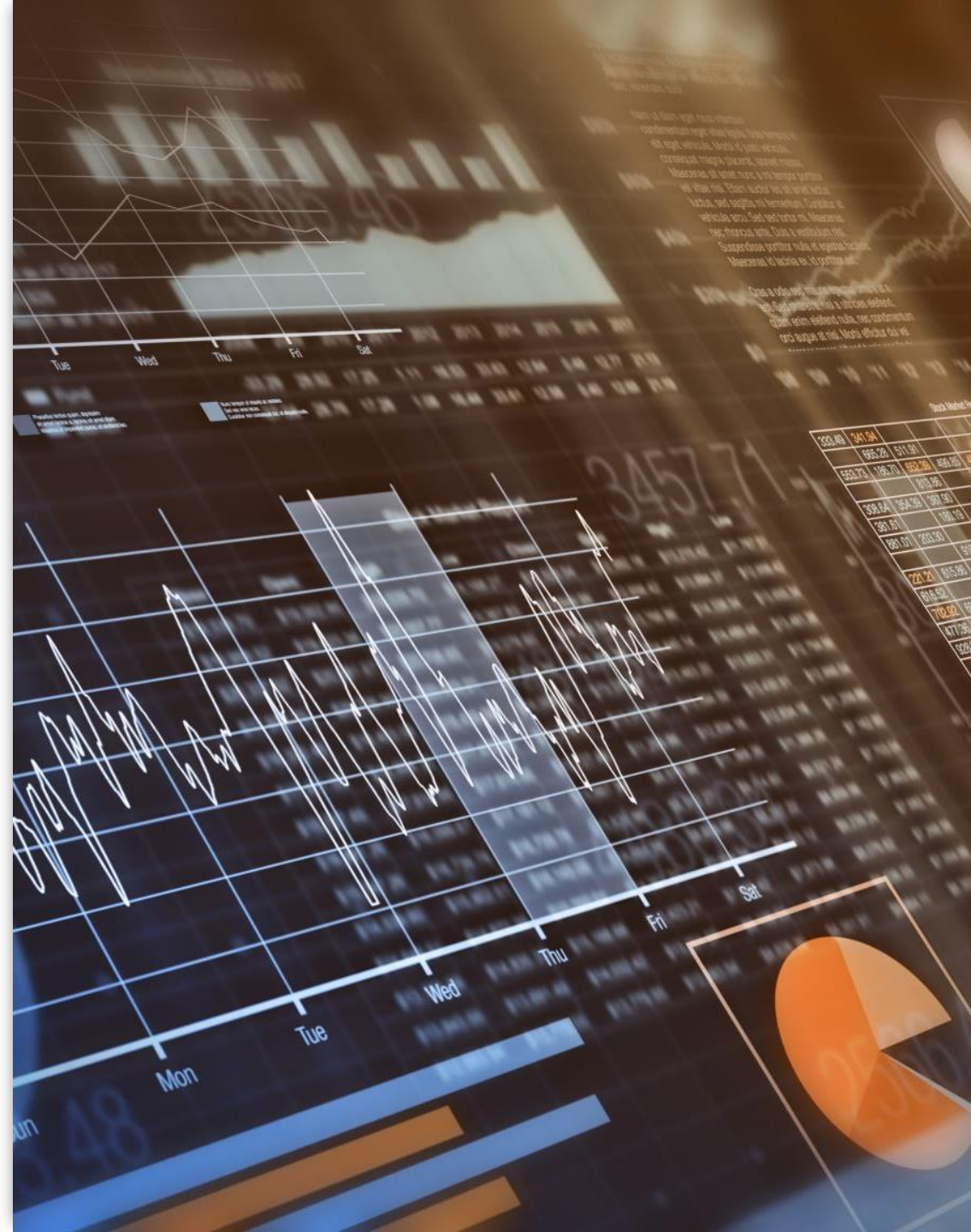
- **Input:** people's explicit or implicit **preferences** for various products or items
 - explicit: ratings
 - implicit: observed behaviour like clicks
- Algorithm makes recommendations on what a user may like by **finding similarities** between the users or items.
- **Output: recommendations** based on
 - what similar users liked
 - what other products resemble the ones the user already purchased

Machine learning workflow

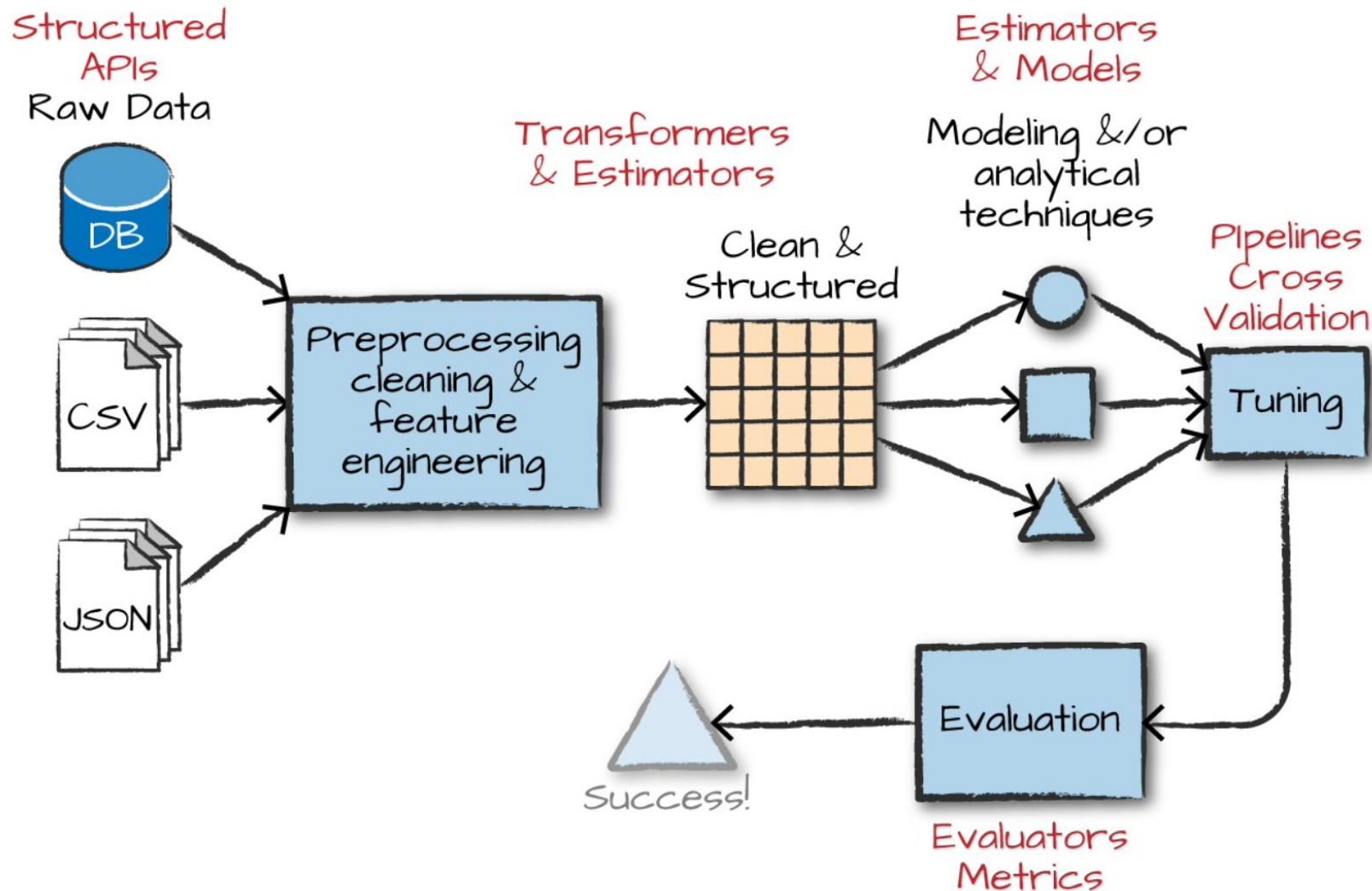


ML workflow steps

1. Gathering and collecting the relevant data for your task.
2. Cleaning and inspecting the data to better understand it.
3. Performing feature engineering to allow the algorithm to leverage the data in a suitable form (e.g., converting the data to numerical vectors).
4. Using a portion of this data as a training set to train one or more algorithms to generate some candidate models.
5. Evaluating and comparing models against your success criteria by objectively measuring results on a subset of the same data that was not used for training.
6. Use the model to make predictions, detect anomalies, or solve more general business challenges.



Spark ML workflow



MLlib structural types

End-to-end machine learning pipelines must define processing using these types:

- transformers
- estimators
- evaluators
- pipelines



Preparing data set: randomSplit



Divide our data set into two groups: train and test



Many data scientists use 80/20 as a standard train/test split.

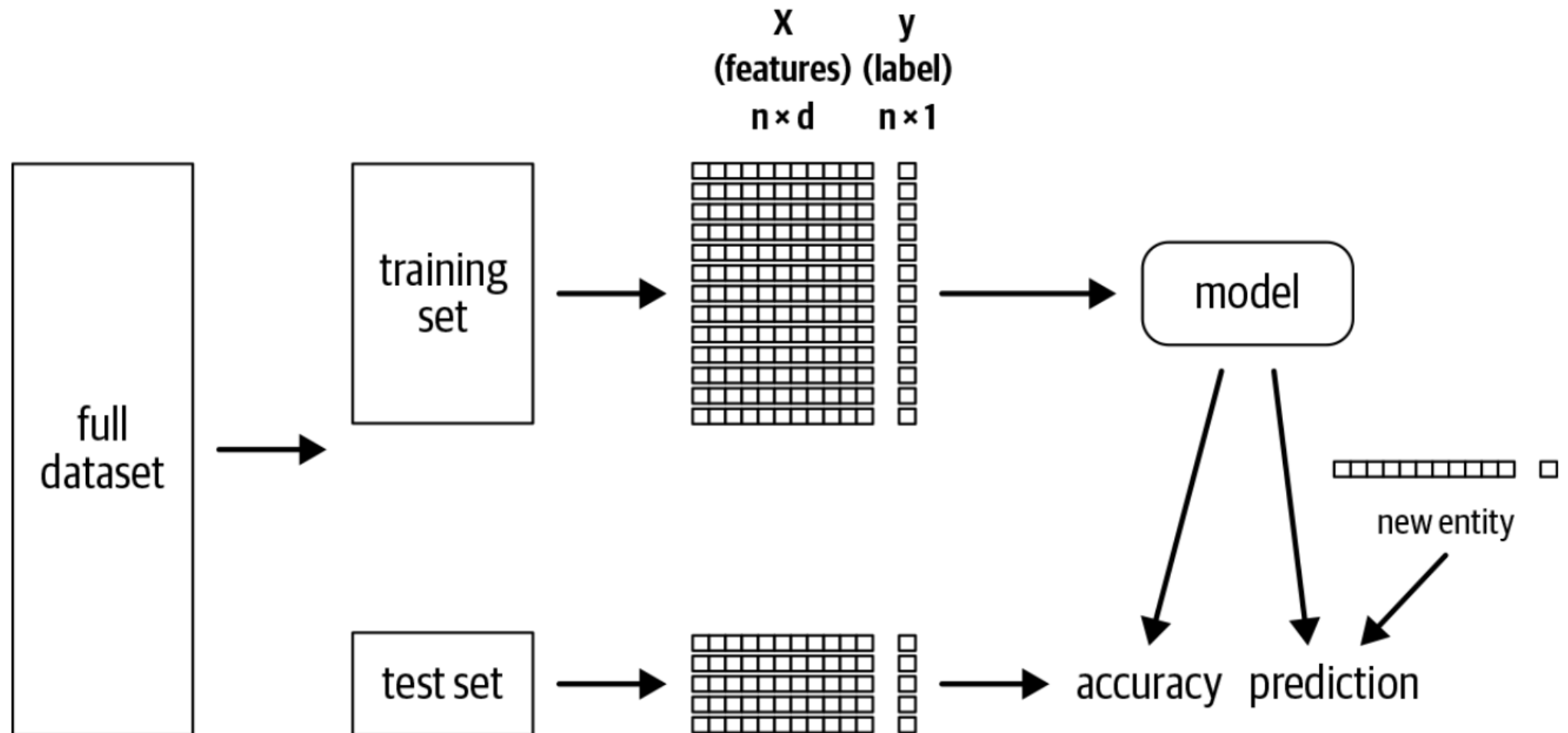


Why? if we built a model on the entire data set, it's possible that the model would memorize or "overfit" to the training data we provided



The model's performance on the test set is a proxy for how well it will perform on unseen data assuming that data follows similar distributions

Dataset to model



Transformers

- Transformers in Spark accept a DataFrame as input and return a new DataFrame with usually one added column.
- Transformers do not learn from your data, but apply rule-based transformations using the transform() method
- Used in preprocessing and feature engineering.
- Example: convert string categorical variables into numerical values

VectorAssembler transformer

concatenate all your features into one big vector

```
+-----+-----+-----+-----+
|int1|int2|int3|VectorAssembler_403ab93eacd5585ddd2d__output|
+-----+-----+-----+-----+
|    1|    2|    3|                                     [1.0,2.0,3.0]|
|    4|    5|    6|                                     [4.0,5.0,6.0]|
|    7|    8|    9|                                     [7.0,8.0,9.0]|
+-----+-----+-----+-----+
```

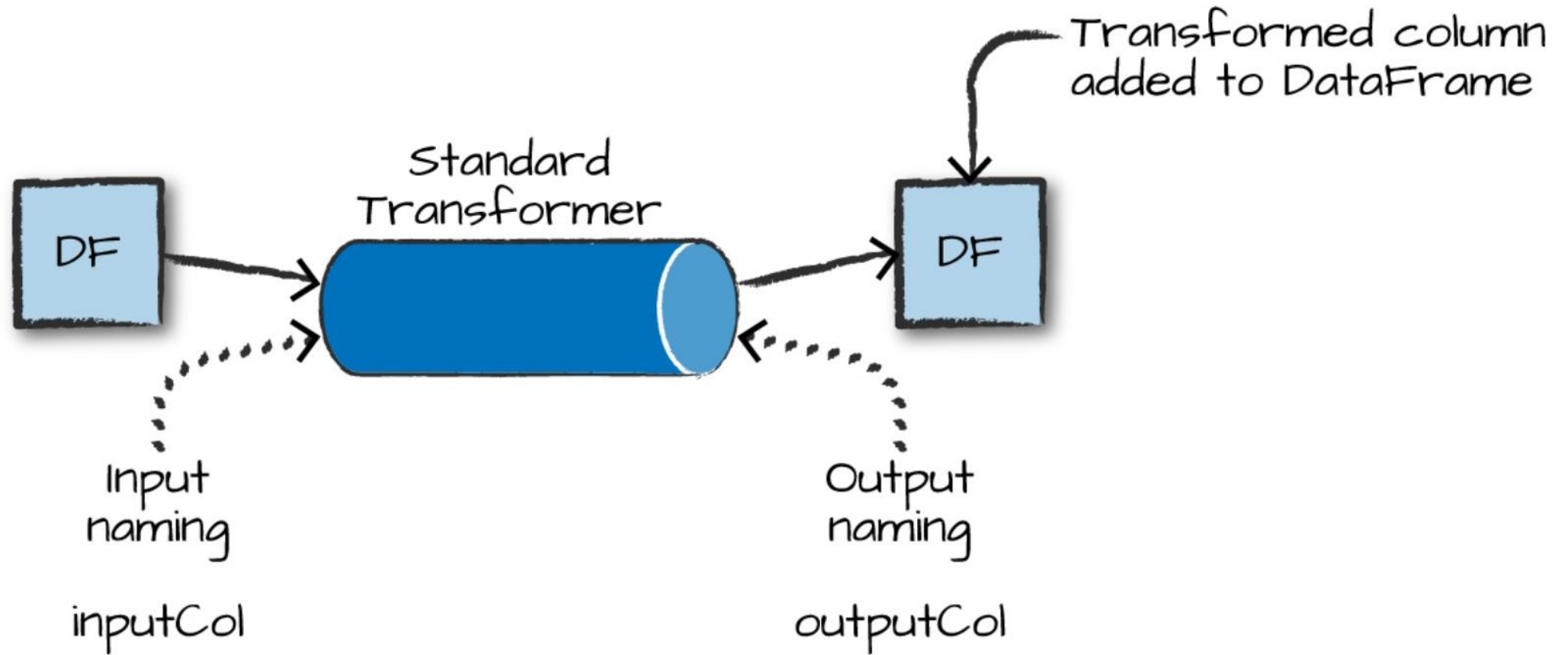
Transformer Python example

```
(trainDF, testDF) = indexedDF.randomSplit([0.8,0.2],seed=1)
```

```
vecAssembler= VectorAssembler(inputCols=trainDF.columns,  
                               outputCol = "features")
```

```
vecTrainDF = vecAssembler.transform(trainDF)
```

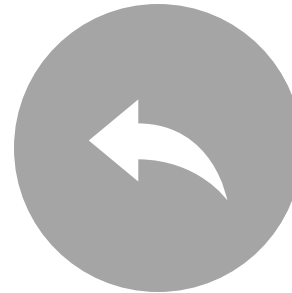
Transformer



Estimators



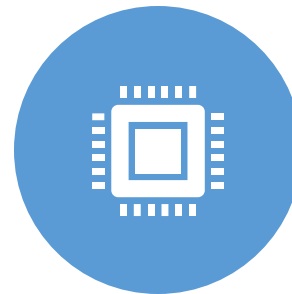
Estimators learn parameters from your data



Take in a DataFrame and return a **Model**



This requires two passes over our data—the initial pass generates the initialization values and the second actually applies the generated function over the data



In the Spark's nomenclature: algorithms that allow users to train a model from data are also referred to as estimators

Linear regression model estimator

```
lr = (  
    LinearRegression(  
        featuresCol="features",  
        labelCol="SalePrice",  
        maxIter=10,  
        regParam=0.8,  
        elasticNetParam=0.1,  
    )  
)  
lrModel = lr.fit(trainDF)  
predDF = lrModel.transform(testDF)
```

predicting with models

```
lrModel = lr.fit(trainDF)
```

```
predDF = lrModel.transform(testDF)
```

1. `lr.fit()` returns a transformer
2. we use this new transformer to apply the model parameters to new data points to generate predictions

Evaluators

- An evaluator allows us to see **how a given model performs** according to criteria we specify
- Use an evaluator to check model quality
- Then, decide to use that model to make predictions
- Typical evaluators: R^2 , RMSE (Root Mean Squared Error)

linear regression evaluator

```
lrEvaluator = (  
    RegressionEvaluator(  
        predictionCol="prediction",  
        labelCol="SalePrice",  
        metricName="r2",  
    )  
)  
r2=lrEvaluator.evaluate(testDF)
```

Pipelines

- We can specify each of the transformations, estimations, and evaluations one by one
- It is often easier to specify our steps as stages in a pipeline.
- This pipeline is similar to scikit-learn's pipeline concept.

SparkML Pipeline with two stages

```
pipeline=Pipeline(stages=[vecAssembler, lr])  
pipelineModel=pipeline.fit(trainDF)  
predDF=pipelineModel.transform(validationDF)
```

movielens dataset for recommendation

User ID	Movie ID	Title	Rating
42	281	River Wild, The (1994)	3
707	347	Wag the Dog (1997)	5
454	197	Graduate, The (1967)	4
943	23	Taxi Driver (1976)	4
199	539	Mouse Hunt (1997)	1
846	187	Godfather: Part II, The (1974)	4
406	132	Wizard of Oz, The (1939)	5
394	742	Ransom (1996)	5
306	1009	Stealing Beauty (1996)	4
312	8	Babe (1995)	5

Matrix with users opinions

	Movie 539	Movie 281	Movie 347
Jordi	1	?	4
Mariona	1	3	5

We wish to be able to predict how Jordi will rate Movie 281

Alternating least squares

	Movie 539	Movie 281	Movie 347
Jordi	1	?	4
Mariona	1	3	5

$$= \begin{bmatrix} U_1 \\ U_2 \end{bmatrix} * \begin{bmatrix} P_1 & P_2 & P_3 \end{bmatrix}$$

The goal of Alternating Least Squares is to find two matrices: U and P
 $U * P$ is approximately equal to the original matrix of users and products

we are able to predict what Jordi will think of movie 281 by multiplying row 1 of U with column 2 of P

Recommending items with Spark: build model

Alternating Least Squares (ALS) algorithm

ALS is a type of estimator that takes in a DataFrame and returns a Model

```
als = ALS(userCol="userId", itemCol="movieId", ratingCol="rating")  
model=als.fit(trainDF)
```

	movieId 539	movieId 281	movieId 347
userId 1	rating: 1		4
userId 2	1	3	5

Get predictions

Use validation DataFrame to obtain predictions for any user

```
1.predDF = model.transform(validationDF)
2.usersRec = model.recommendForAllUsers(5)
3.userPredictedRatings = usersRec.first().userId
```


use model to predict movie ratings

userId	movieId	prediction
471	281	3.0085273
471	2366	3.1742785

Enquesta curs 24/25

- <https://www.uab.cat/enquestes/>
- [http://sia.uab.es/servei/ALU ENWES PORT CAT.html](http://sia.uab.es/servei/ALU_ENWES_PORT_CAT.html)