Popularity of Social Media Videos

Megan Jasek, Charles Kekeh,
Alejandro Rojas, Andrea Soto

Context

- Growth in online video consumption and creation
- Major social media networks are giving increasing relevance to video content
- Content and eyeballs are migrating to video
- Opportunity to develop tools that track, report and predict performance of online videos

Value Proposition

What video content to publish?

When to publish?

Where to publish?

Scope

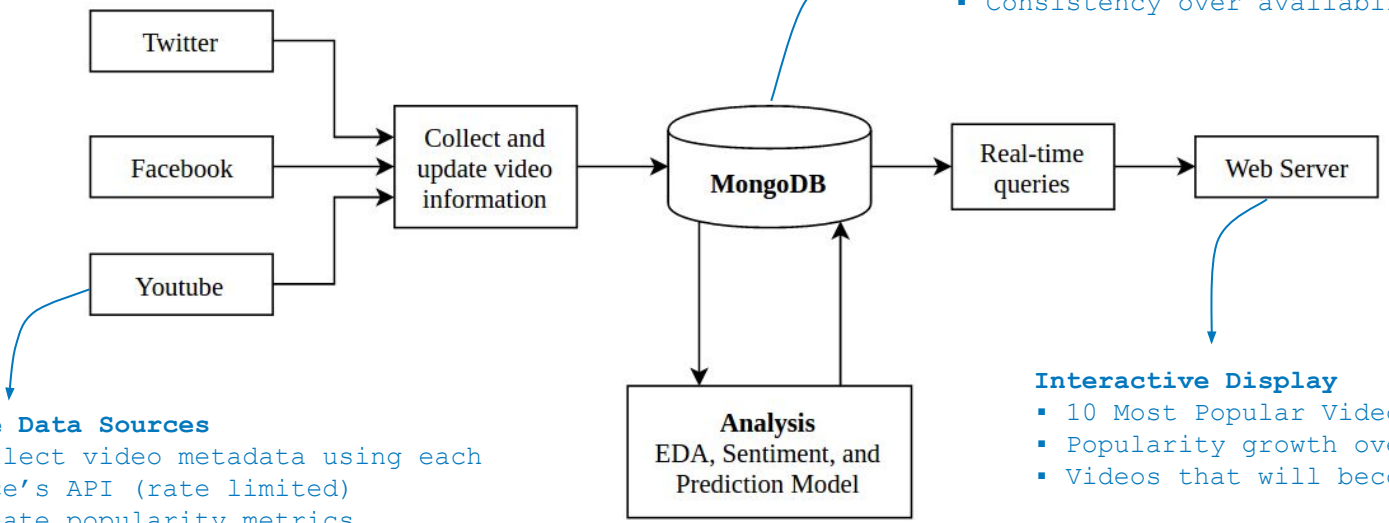
- Identify what videos are popular across three social media platforms: Twitter, Facebook, and Youtube.
- Ingest video metadata from the three sources and analyze the content to build a prediction model.

Live Demo of Prototype

Architecture

Toos: Softlayer,MongoDB,Robomongo, Python,Cron,Spark,HTML,CSS,Javascript

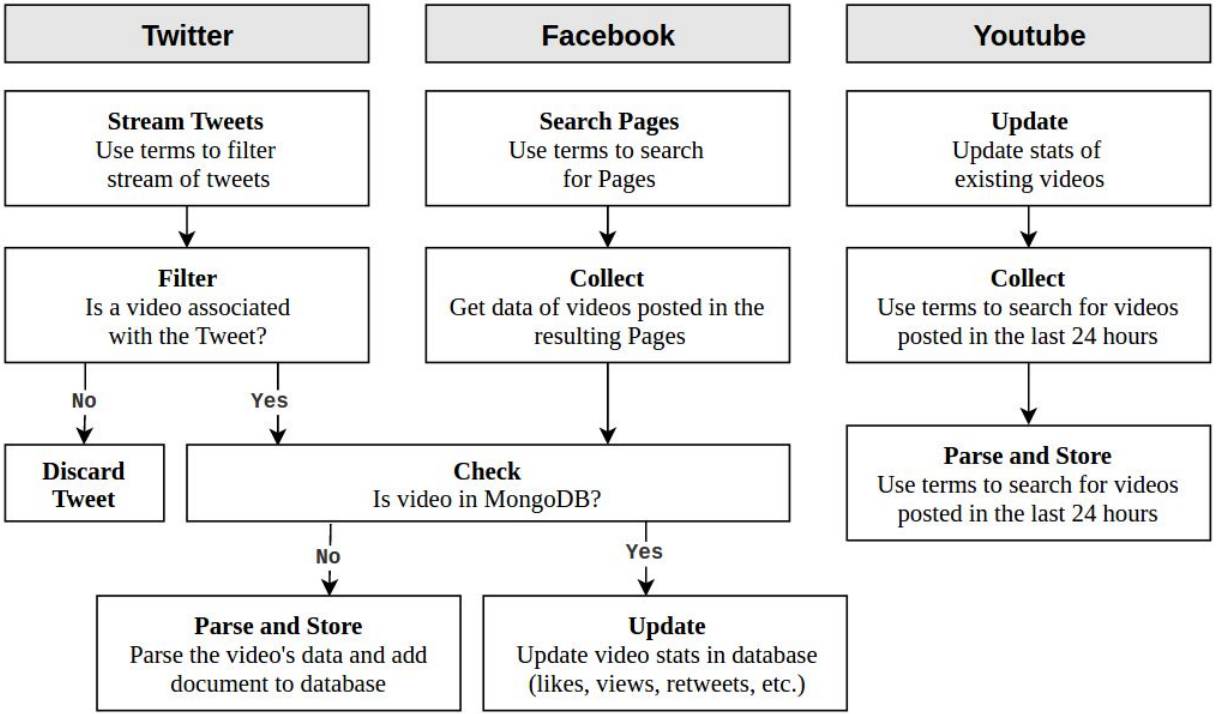
MongoDB Database
- Dynamic schema (JSON documents)
- Easy to implement and develop
- Easy javascript queries
- Consistency over availability



Three Data Sources
- Collect video metadata using each source's API (rate limited)
- Update popularity metrics
- Different information and schemas

Interactive Display
- 10 Most Popular Videos by Source
- Popularity growth over time
- Videos that will become popular

Data Ingestion Processes



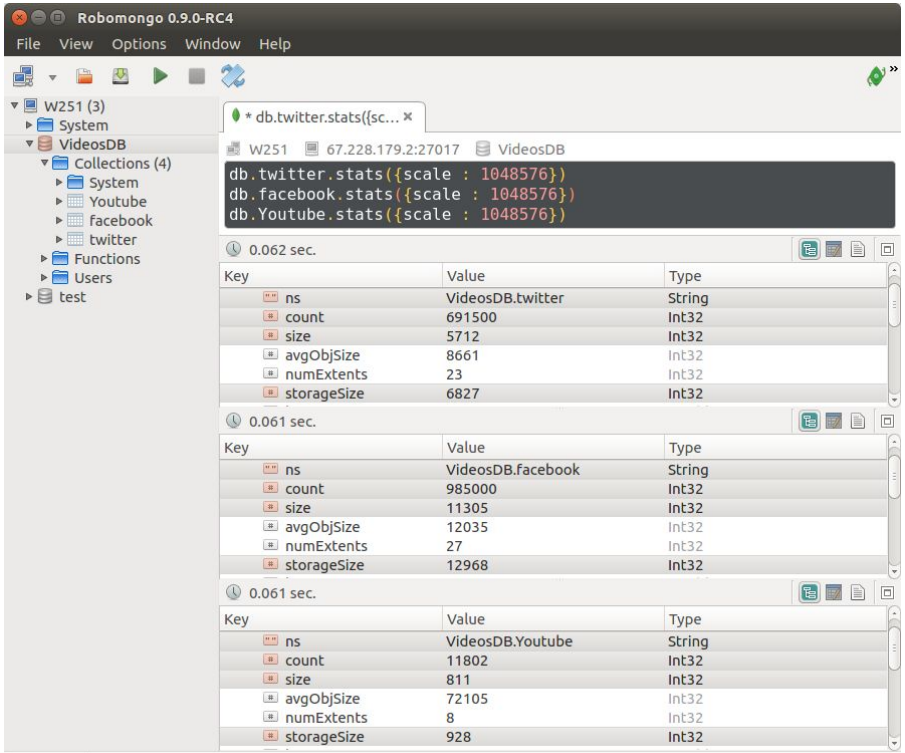
- **Topic Focused**
  Search/Filter API's using 74 terms related to entertainment, sports, music, and cooking

- **Ingest**
  Continuous for Twitter and Facebook. Scheduled every 24 hours for Youtube.

Dataset Overview

- ★ ~1,7 million videos
- ★ 18 GB raw size
- ★ 20 GB storage size

Twitter	691,250 videos 5.7 GB raw size 6.8 GB storage size
Facebook	985,000 videos 11.3 GB raw size 12.9 GB storage size
Youtube	11,802 videos 0.8 GB raw size 0.9 GB storage size



Data Analysis - Spark

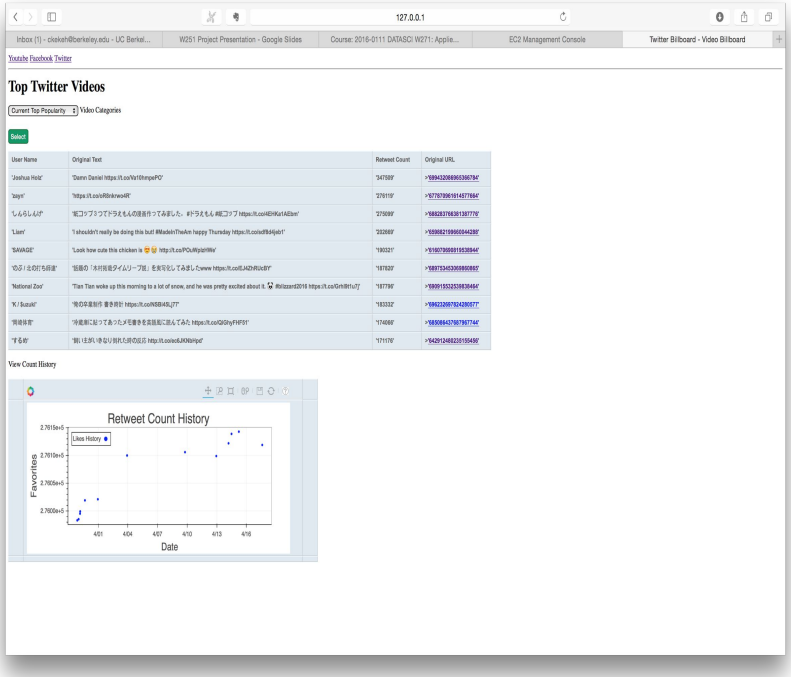
- Prediction - predicted video popularity using logistic regression
  - Data from all 3 sources: Twitter, YouTube, Facebook
  - Features: video_length_sec, popularity_count, other_count, growth_rate, sentiment, source
  - Model: weights=[-155.6, 16.1, 13.3, -0.48, -0.43, -7.37], intercept=0.0
- Spark - MLLib with python
  - LabeledPoint data type
  - LogisticRegressionWithSGD, LogisticRegressionModel
- Create Model
  - Process: 1) load data from JSON, 2) filter data, 3) create features, 4) split data, 5) create/save model, 6) evaluate model
  - Sentiment Analysis: vaderSentiment python library
- Predict from Model
  - Process: 1) use pymongo to load data from MongoDB, 2) make predictions using stored model, 3) write predictions to MongoDB

```
sc = SparkContext(appName="SparkCreateModel")
# LOAD data
twitter_data = load_data_from_file(sc, "file:///root/mongoData/twitter.json")
youtube_data = load_data_from_file(sc, "file:///root/mongoData/youtube.json")
facebook_data = load_data_from_file(sc, file:///root/mongoData/facebook.json")
# CREATE  MLLib LabeledPoints (LP =  LabeledPoint(DependentVar, [FeaturesList]))
twitter_LP = twitter_data.map(create_labeled_points_twitter)
youtube_LP = youtube_data.map(create_labeled_points_youtube)
facebook_LP = facebook_data.map(create_labeled_points_facebook)
# SPLIT DATA into training (80%) and test(20%) sets
train_twitter, test_twitter = twitter_LP.randomSplit([0.8, 0.2], seed=0)
train_youtube, test_youtube = youtube_LP.randomSplit([0.8, 0.2], seed=0)
train_facebook, test_facebook = facebook_LP.randomSplit([0.8, 0.2], seed=0)
# COMBINE all 3 datasets with the RDD.union command
train_LP = train_twitter.union(train_facebook).union(train_youtube)
test_LP = test_twitter.union(test_facebook).union(test_youtube)
# BUILD MODEL -  logistic regression
model_log = LogisticRegressionWithSGD.train(train_LP)
if store == True:
    model_log.save(sc, file_path)
# EVALUATE MODEL on test data
preds_test_log = test_LP.map(lambda p: (p.label, model_log.predict(p.features)))
```

User Interface

Two scenarios implemented:

Current most popular videos across ingestion sources and popularity history
Most popular videos predictions across ingestion sources and popularity history



- Python Web app using Flask and Bokeh
- Data is retrieved on access from documents in MongoDB

Path Forward

- Tracking tool:
  - Publishers compare content performance across social media platforms
  - Advertisers evaluate publishers performance to determine communication strategies
- Image Classification tool:
  - Analyze images to identify same video posted on multiple networks by multiple users
  - Classify videos according to its images
- Predictive tool:
  - Identify windows for publishing different type of videos on each network
  - Predict virality of videos
  - Propose video guidelines to improve video virality

The End

Questions?