1. SO YOU THINK YOU KNOW ABOUT LINEAR REEGRESSION:

Bayesian Linear Regression , MCMC -> python lib pyMc available , Max Aposteriori (take log prior , log likelihood) Estimation, Multicollinearity , Ordinary Least Squares ,

Ridge Regression( penalizes coeefficients) , LASSO Regression , Informed Opinion , Cauchy Distribution

1. A STUDY IN CLASSIFICATION:

Distributed Crawling

Product Matching

Unsupervised Attraction

Harmonized System code

Scikit library

<https://prodi.gy>

Titus Winters , Cppcon 2017

Talks about software engg vs programming:

software engg is programming integrated over time

Programming is a task which finishes once done.

How you keep your programming working is engineering

https://www.youtube.com/watch?v=tISy7EJQPzI

beyonce rule-> if you like it put a test on it

1. Improving product discovery via Relevancy and Ranking optimization , Flipkart

They take into account clicks and purchase signals

Use the following techniques:

1. Collaborative filtering
2. Attribute similarity
3. Visual similarity

Focus on diversity in recommendations, ltr , ensuring product sanity (measuring accuracy)

*Big billion days:* adjust training data with sale days as a feature. As people log in to check just the sale prices and not purchase immediately

How to compute Quality feature? Ratings data/reviews. Higher quality -> higher conversion. Return Rate is also one factor

Historical Feature – use feedback loop

Presentation feature – how the results are displayed on the screen(Position bias-> result on right thumb gets selected more) Normalize clicks from android device and desktop

* Gets stale veery fast

1. Needle in a haystack, entity search on text and graph

Following options are available in the market

1. Knowledge graph-Wikipedia/Facebook?
2. Propreity- Linkedin
3. Satori-Microsoft

Why search engines don’t perform better in Entity extraction?

Not a direct match,

Doing it for large set specifically long tail query

Low coverage,

Lack of coverage

Missing entities

Approach to Entity Search Engine:

1. Use WebText with Knowledge Graph

Eg: film bahubali actor prabhu

( identify entities in text , annotaters) entity tagging

1. Query interpretation( Queery to KG map)

Three tokens-> entity , type , relation

Approach-> tag entities using tagger pin it on knowledge graph

Open source-Tag Me(Entity disambiguation)

1. Query to WebText Match
2. Direct Entity ranking
3. Use CNN to identify score (IDF score) of those words

**Weak supervision**: not exact match

**FAACC1 corpus**-CMU

**Clue web o9**

**Freebase Knowledge graph**

Web+KG performs better . Learning Joint Query Interpretation and Response Ranking

1. BUILDING ANALYTICS APP WITH STREAMING API IN AOACHE SOLR, LUCIDWORKS

Solr-> challenges building app in real time -> search and filter the data before performing analysis

->time consuming

Ans-> Solr cloud? Apache zookeeper? , Advance LTR

Parallel Computing Framework consists of following components:

1. Streaming aPI
2. Streaming Expression
3. Shuffling
4. Worker Collection
5. Parallel SQL

org.apache.solr.client.solrj.io.\*

tuple stream consist of three componeents:

1. Stream sources eg. Search facet
2. Stream Decorators eg. Row-wise hashJoin , innerJoin , merge
3. Stream Evaluators column wise eg. Div / multiply
4. SERVICIBILITY UNDER HIGH DEMAND , SWIGGY

Constraints:

1. Finite delivery fleet
2. Weatheer
3. Unpredictable demand
4. Restraunt prep time
5. Variation in delivery agent speed
6. Fleet is only partially on demand

How to address the problems:

1. Quantify load(orders)
2. Represent inflows and outflows as queuing model abstraction
3. Develop predictive models
4. Realtime strategies to reduce demand
5. Intelligent demand allocation

Compute **Stress Ratio: #undelivered orders/Delivery agent**

Queuing model

Delta d -> inflows of orders

Delta b -> outflows

Throttle the orders??

Demand Prediction

Preedict no of orders (delta 10 mins)

**Demand Shaping:**

Use customer segmentation , premium customers get priority in thee available slot(revenue loyalty taken into consideration)

Assignment algorithm -> prep time , restraint to customer travel time , batching – grouping customers