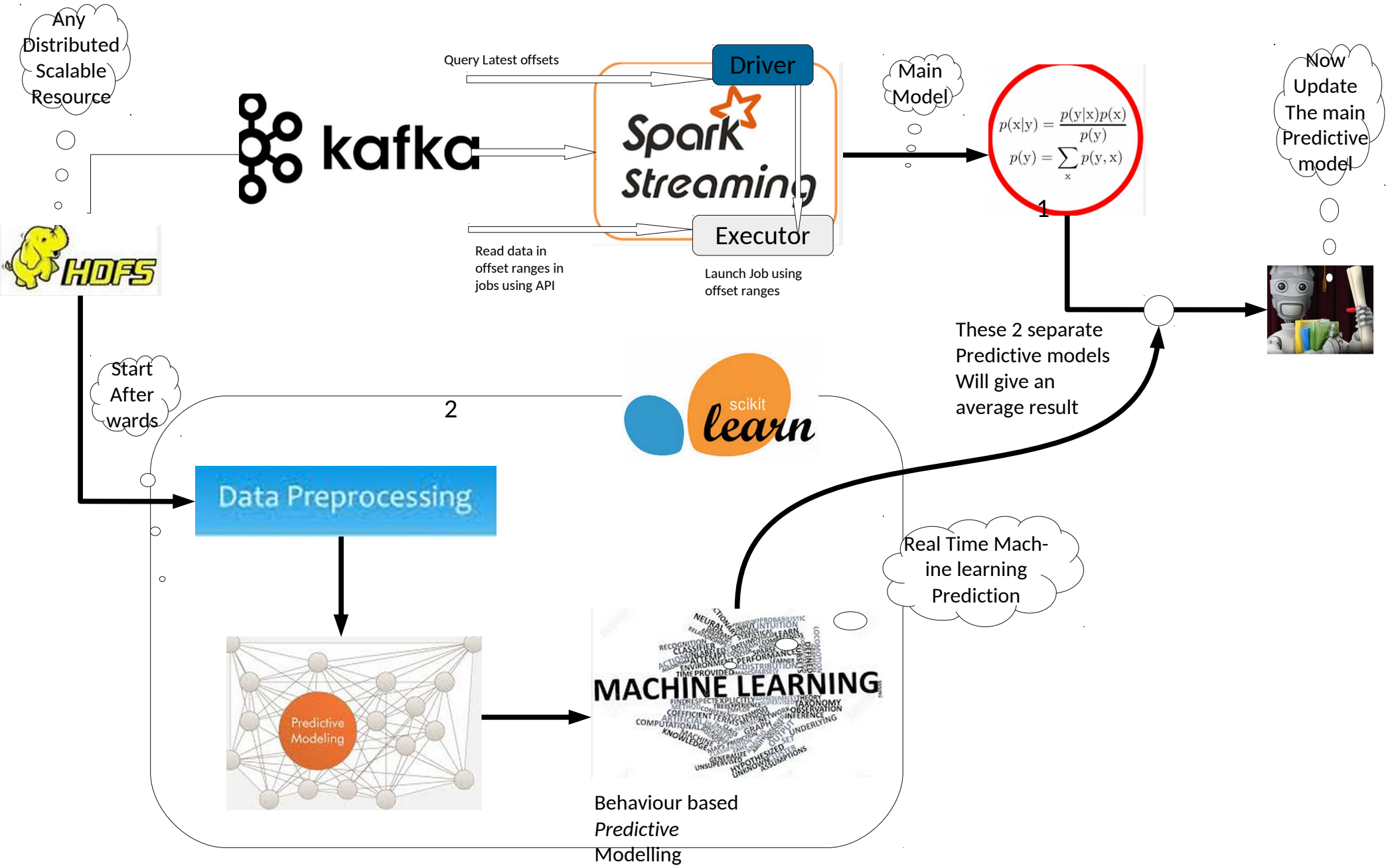
Smaato AdExchange Infrastructure

Presuming that Smaato's AdExchange (SOMA) can deliver detailed data on each auciton in any desired way, please describe how would you construct a component which calculates optimal floor prices. Expected is a list of which frameworks / software packages would you use and how these will interact.

Please describe, if you would use any AI / ML method for calculating the floor prizes or not. If yes, please describe, which. If no, please describe, why not.

Basic Infrastructure



Why Apache Kafka ?

- It is fault-tolerant, scales to enormous data sizes, and has a built in partitioning model
- Kafka is a high throughput, distributed log that can be used like a queue(actually its a circular buffer), takes less memory than Redis
- Real time analytic by consumer keeps track of data or ask for next time
- Very popular for high volume data processing

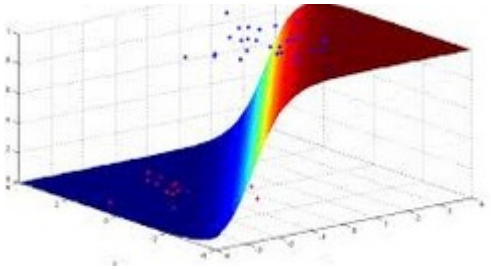
Direct Kafka

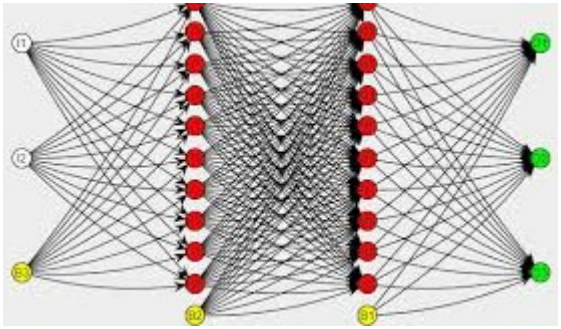
- Using Direct Kafka Approach because it has 1:1 partition ie Spark:Kafka partition , leads to cheap parallelism.
- So kafka partition with end:start offset leads to 1spark partition ie 1 RDD, so if executor dies just recreate RDD.
- No duplicate writes
- Offsets are access by spark streaming within its checkpoints
- Output operations can be Idempotent updates or Transaction updates

Why Apache Spark Streaming ?

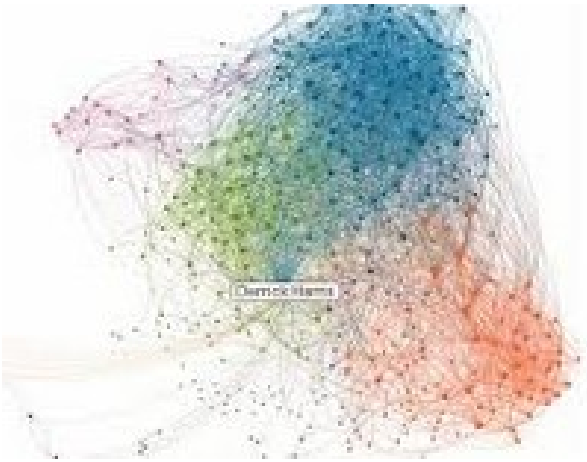
- Its beyond replication or upstream backup concept, but it based on DAG model named Discretized Stream
- Dstream is actually sequence of RDD.
- It structure streaming computation as series of stateless, deterministic batch computation on small interval and under the hood it uses RDD , so fault tolerance is brilliant
- Great Load Balancing with existing Spark infrastructure
- Advanced analytic and Machine learning library access
- Unification of batch, streaming and interactive analysis using spark sql and dataframe
- Gives windowing to aggregate all records from the sliding window of past time interval in to a particular RDD
- Provides some fundamental streaming APIs which comes handy like reduceByWindow, state Tracking etc.

MACHINE LEARNING TYPES





- Regression
- Decision Tree
- Neural Network
- Ensemble Models
- SVM
- Random Forest



Which Machine Learning Algorithm??

- *What is the size of Training data* ? - well Smaato has million of daily transactions so I believe an ETL and pre processing will give quite a good set of Dataset for training the Main model
- *Data Dimension/Features* – I predicted few like Add Type , users , time , max and min bid but there might be some feature which seems like independent but statistics shows its not, so depend on the Table Smaato builds
- *Feature Independent* ? - As above
- *Linear Dependency on Target*– Depends on the data
- *Is over-fitting expected*? - This is crucial, because result can be highly biased based on distribution of dataset and how each category of Adds collects or behave. Some adds might have less training data than other , or some add types are having huge number of bidders compared to others
- *Scalable Infrastructure* – What's Smaato infrastructure, depends on Nodes or scalability

Logistic Regression

- Logistic Regression is Easy
- Linear and linear separable
- Can change non linear feature to linear
- Robust to Noise
- Avoid over-fitting
- Can do feature selection using l1 or l2 regularization

SVM

- SVM uses different loss function(Hinge) from Regression
- If not linearly separable
- Not much Efficient
- Kernel is a big saviour because of Convex Hyperplane concept

Tree Ensembles

- No linear feature required
- Decision tree like flow chart to finalize the result, easy
- Can handle high dimensional state because boosting bagging technique

Deep Learning

- Take input , apply one model, get representation, then repeat the same on another model
- If we have huge data and complex then its good to use

Various Machine Learning Approach

Main Scalable Prediction Model

- This is Add type and other features based prediction
- This will be handling all the basic Required dimensions to build a prediction Model
- Since it won't run real-time , we can update the prediction Model in batch
- We can try ensemble or Neural Network approach
- Neural is like pipeline approach and we are very much particular about optimized result we may try
- This model can be retrained or more values can be added but not in real-time , so we can consider more features so we can assume complex and more time consuming formulas

Real Time Prediction

- This is behaviour based prediction
- While bidding happens, based on DSP involved , will fetch there previous trend and based on current entry , build a training prediction model in real time
- It won't run on massive data , so it will be quick
- Avg. the Main Prediction & Real time prediction to understand the squared error or log-loss
- Can use regression or Decision tree to build the prediction model
- Less features so quick analysis and prediction so regression even Random Forest can come handy

DSP Details

DSPID	**ADDTYPE**(any value from 0… n , presents what category of add , like game , social, finance)	**MIN BID**(in Amount)	**MAX BID**	**BIDDING FREQUENCY**(number of biddings made)	**TIME IN**(can find the active state of DSP)	**TIME OUT**(How long DSP tried bidding)	**RESULT**
1	1	2	10	5	10552	20555	1
1	0	1	20	30	9560	19560	0
1	1	1	5	25	15250	25658	0
1	1	2	6	20	16258	18965	0
1	0	2	10	10	15246	60598	1
1	2	5	15	5	1158	5896	0
1	1	1	15	10	11585	36989	0
1	2	1	18	5	989655	11100258	0
1	2	3	18	6	158934	256898	1
1	6	10	12	9	15872	99564	0

ADD Details

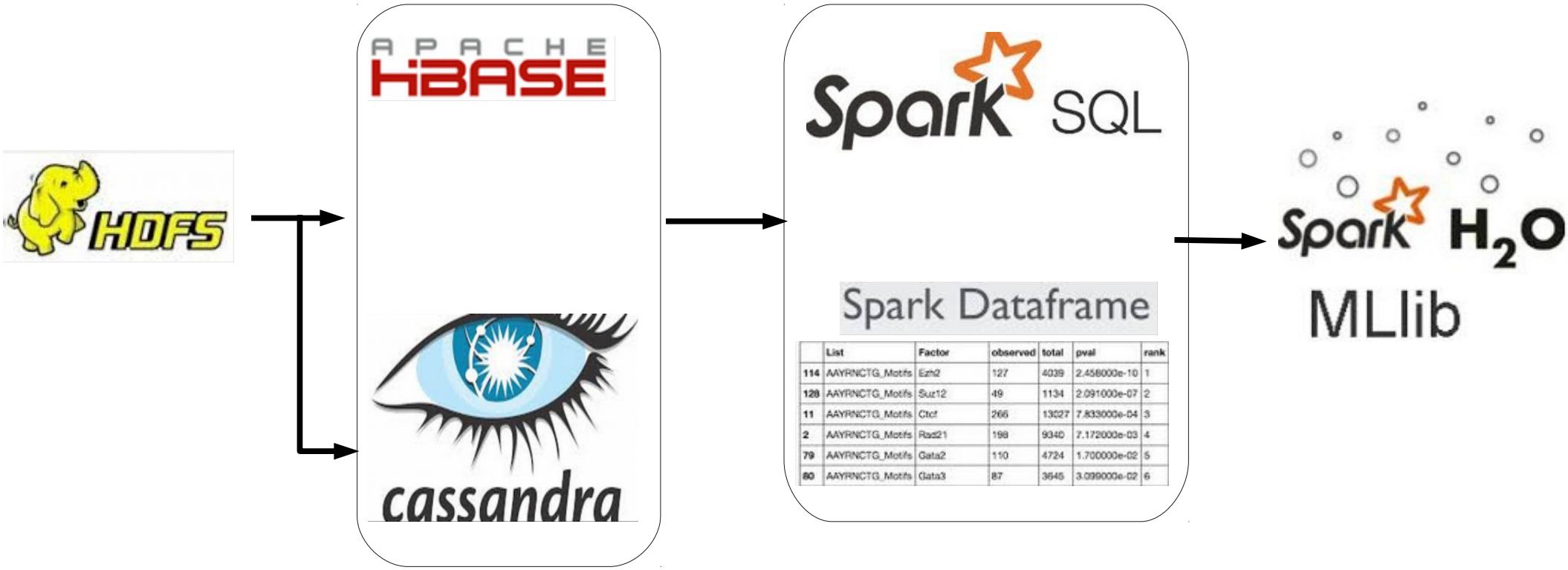
ADD ID	**ADD TYPE**	**Avg. MIN BID (**based on frequency)	**Avg. MAX BID**(winning)	**Avg. Bids**(in count)	**TIME**(in range)
1	1	1	15	10000	25565125
2	1	1	10	15245	2463656
3	2	1	12	65895	5745651
4	3	1	2	14000	57478585
5	2	1	4	25795	5475565
6	3	1	5	36549	9699856
7	4	1	6	24879	789222
8	4	1	12	35592	63697425
9	4	2	12	14736	6889523
10	5	3	10	36985	6584856

Basic Statistics

- Find Standard variation to determine the spread of data
- Find any outlier , any value which crosses the normal trend, that might be some spam or ...
- P(A & B) = P(A)P(B|A)

  probability of 1st event happening, then probability of 2nd event happening given 1st.
- P(A|B) = (P(A)P(B))/P(B)
- Build a Histogram or build exploratory Analysis
- Check the Distribution may be continuous, binary or normal
- Find Dimension and if required use PCA

Main Prediction Model(MPM)

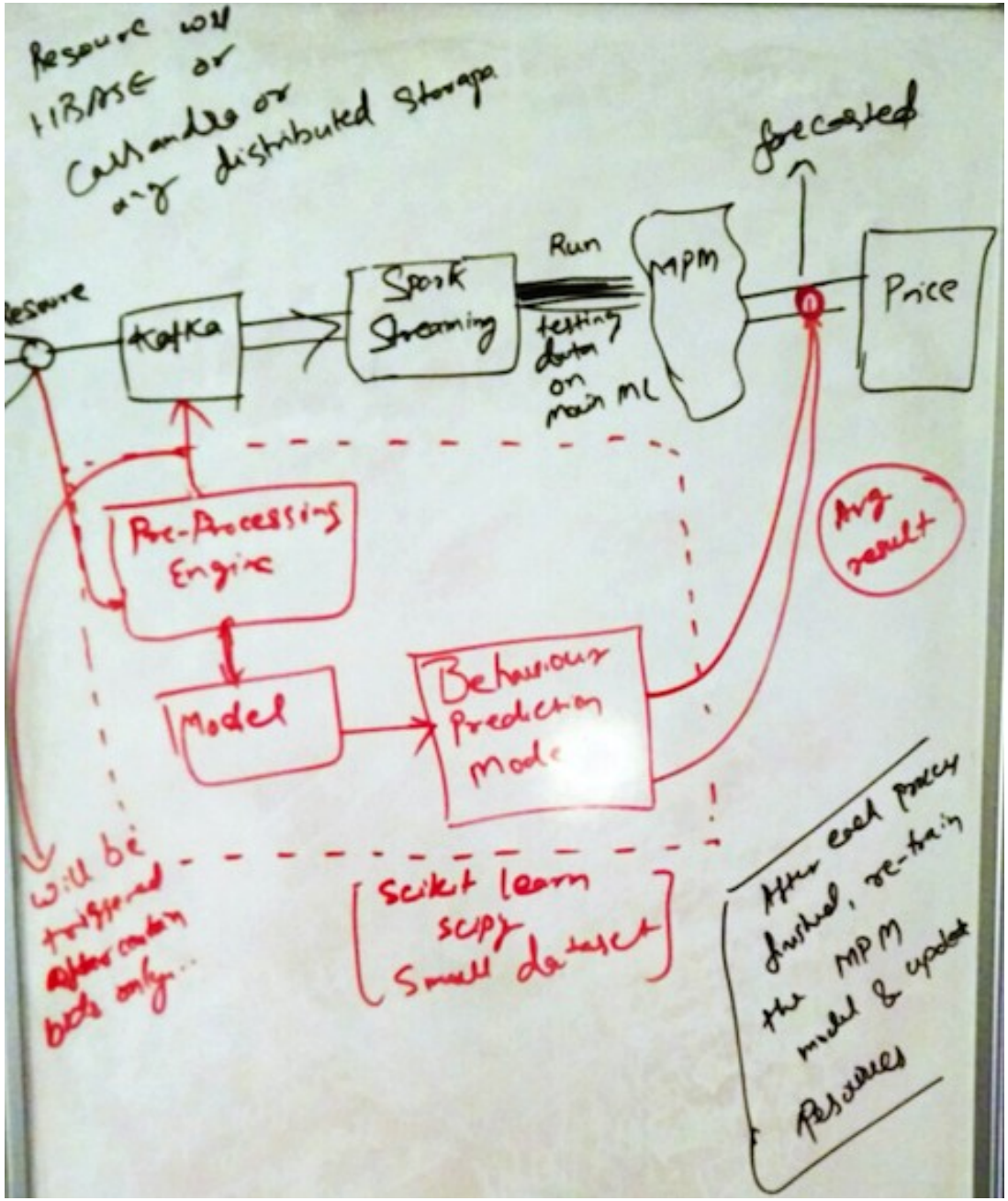


	List	Factor	observed	total	pval	rank
114	AAYRNCTG_Motifs	Ezh2	127	4039	2.458000e-10	1
128	AAYRNCTG_Motifs	Suz12	49	1134	2.091000e-07	2
11	AAYRNCTG_Motifs	Ctcf	266	13027	7.833000e-04	3
2	AAYRNCTG_Motifs	Rad21	198	9340	7.172000e-03	4
79	AAYRNCTG_Motifs	Gata2	110	4724	1.700000e-02	5
80	AAYRNCTG_Motifs	Gata3	87	3645	3.099000e-02	6

Any Distributed Resource depend on data tpe

For Data manipulation,

Machine Learning Model.
We will have 2 different ML running
One meant for floor Price prediction
other DSP behaviour prediction


Resource
Kafka
Spark Streaming
Run
testing data on main ML
MPM
Price
processed
Re-Processing Engine
Model
Behaviour Prediction Mode
Avg result
will be triggered after certain batch only...
Scikit learn
scipy
Small dataset
After each process finished, re-train the MPM model & update Resources

Real-Time Processing Pipeline