CRAFTML, an Efficient Clustering-based Random Forest for Extreme Multi-label Learning

Bidya Sarkar(18111011), Nirjhar Roy(18111409), Avijit Roy(18111404), Manish Mazumder(18111038)

Indian Institute of Technology, Kanpur

November 26, 2018

Outline

Overview

- Extreme Multi-label Learning (XML) is designed for the large data sets with huge number of labels that can exceed one million.

Overview

- Extreme Multi-label Learning (XML) is designed for the large data sets with huge number of labels that can exceed one million.
- Parallel approach with trees, which reduces the problem into small subproblems.

Overview

- Extreme Multi-label Learning (XML) is designed for the large data sets with huge number of labels that can exceed one million.
- Parallel approach with trees, which reduces the problem into small subproblems.
- Inclusion of randomization in tree based approach, Uses Random forest approach.

Prior Work

- multi-label k-nearest neighbors (ML-kNN) and of the multi-label random forest (Suffers from scaling issues)

Prior Work

- multi-label k-nearest neighbors (ML-kNN) and of the multi-label random forest (Suffers from scaling issues)
- **3 Popular Techniques: To resolve Scaling Issues**
  - Optimization Tricks and Parallelization: PDSparse, PPDSparse, DISMEC
  - Dimensionality Reduction:WSABIE, LEML, SLEEC, AnnexML
  - Tree-based Methods: LPSR, FastXML, PFastReXML

Prior Work

- multi-label k-nearest neighbors (ML-kNN) and of the multi-label random forest (Suffers from scaling issues)
- **3 Popular Techniques: To resolve Scaling Issues**
  - Optimization Tricks and Parallelization: PDSparse, PPDSparse, DISMEC
  - Dimensionality Reduction:WSABIE, LEML, SLEEC, AnnexML
  - Tree-based Methods: LPSR, FastXML, PFastReXML
- Several Tree Based approach also exists in XML literature.(e.g.,RF-PCT, HOMER, LPSR, FastXML).

Why CRAFTML?

- There is still room for improvement in XML

Why CRAFTML?

- There is still room for improvement in XML
- Exploring two directions: Using very fast partitioning strategies and exploiting tree feature/label randomization.

Why CRAFTML?

- There is still room for improvement in XML
- Exploring two directions: Using very fast partitioning strategies and exploiting tree feature/label randomization.
- Lower complexity computation in each node.

Why CRAFTML?

- There is still room for improvement in XML
- Exploring two directions: Using very fast partitioning strategies and exploiting tree feature/label randomization.
- Lower complexity computation in each node.
- A projection is able to preserve more information than a selection for a same

Why CRAFTML?

- There is still room for improvement in XML
- Exploring two directions: Using very fast partitioning strategies and exploiting tree feature/label randomization.
- Lower complexity computation in each node.
- A projection is able to preserve more information than a selection for a same
- joint random projection of features and labels is more promising to deal with the extreme number of labels ratio of compression

Now Comes the CRAFTML...

Let the number of training examples = n

$$x \in \mathbb{R}^{d_x} \;\; and \;\; y \in \{0,1\}^{d_y}$$

Implementation Contd...

- Algorithm uses K ary decision tree forest.

Implementation Contd...

- Algorithm uses K ary decision tree forest.
- stopping criteria : No. of data points in leaf node < 200

Implementation Contd...

- Algorithm uses K ary decision tree forest.
- stopping criteria : No. of data points in leaf node < 200
- first feature set is sampled using sparserandomprojection, then number of datapoints is sampled.

Implementation Contd...

- Algorithm uses K ary decision tree forest.
- stopping criteria : No. of data points in leaf node < 200
- first feature set is sampled using sparserandomprojection, then number of datapoints is sampled.
- for decision tree forest: forest of 4 trees , each tree having its own k value, row sampled value and feature_sampled value

Implementation Contd...

- Algorithm uses K ary decision tree forest.
- stopping criteria : No. of data points in leaf node < 200
- first feature set is sampled using sparserandomprojection, then number of datapoints is sampled.
- for decision tree forest: forest of 4 trees , each tree having its own k value, row sampled value and feature_sampled value
- in each node k means will run and will return the centroid, based on that data points will be clustered and each cluster will be a child node.

Implementation Contd...

- Algorithm uses K ary decision tree forest.
- stopping criteria : No. of data points in leaf node < 200
- first feature set is sampled using sparserandomprojection, then number of datapoints is sampled.
- for decision tree forest: forest of 4 trees , each tree having its own k value, row sampled value and feature_sampled value
- in each node k means will run and will return the centroid, based on that data points will be clustered and each cluster will be a child node.
- train data split is 9:1 ratio for train and validation datasets

Implementation Contd...

- Algorithm uses K ary decision tree forest.
- stopping criteria : No. of data points in leaf node < 200
- first feature set is sampled using sparserandomprojection, then number of datapoints is sampled.
- for decision tree forest: forest of 4 trees , each tree having its own k value, row sampled value and feature_sampled value
- in each node k means will run and will return the centroid, based on that data points will be clustered and each cluster will be a child node.
- train data split is 9:1 ratio for train and validation datasets
- each node in decision tree is implemented as object and each node having train_tree and train_node_classifier methods.

Results

validation data set used for hyperparameter value:

mediamill dataset:

p@value	Actual Accuracy(%)	TEST data achived accuracy(%)
p@1	85.86	81.22
p@2	69.01	64.84
p@3	54.65	50.74

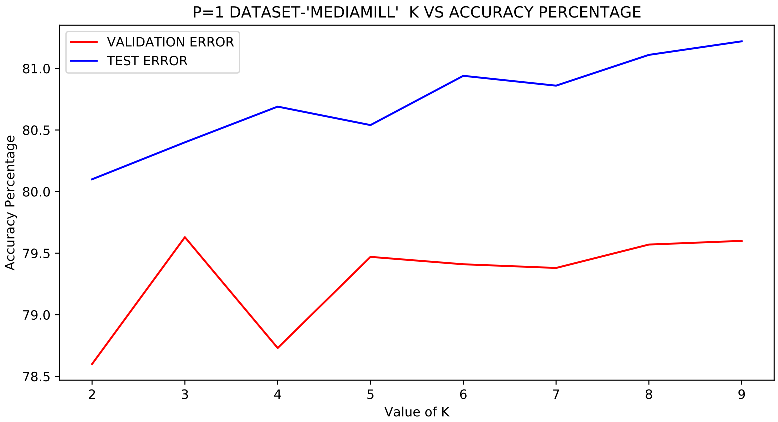
bibtex dataset:

p@value	Actual Accuracy(%)	TEST data achived accuracy(%)
p@1	65.15	61.83
p@3	39.83	27.49
p@3	28.99	20.58

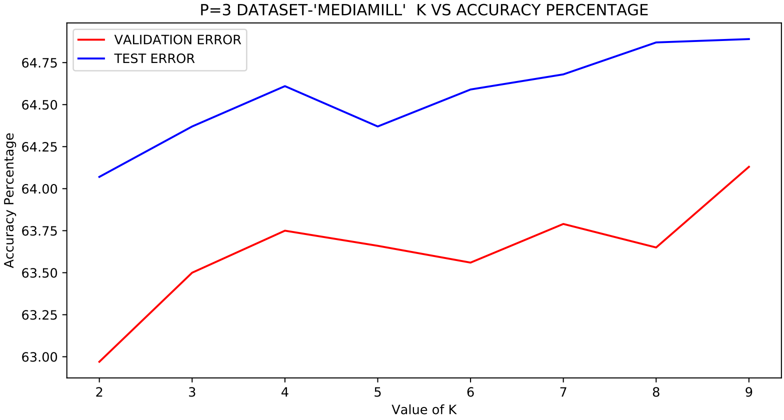
delicious dataset

p@value	Actual Accuracy(%)	TEST data achived accuracy(%)
p@1	70.26	62.85
p@2	63.98	56.43
p@3	59.00	52.17

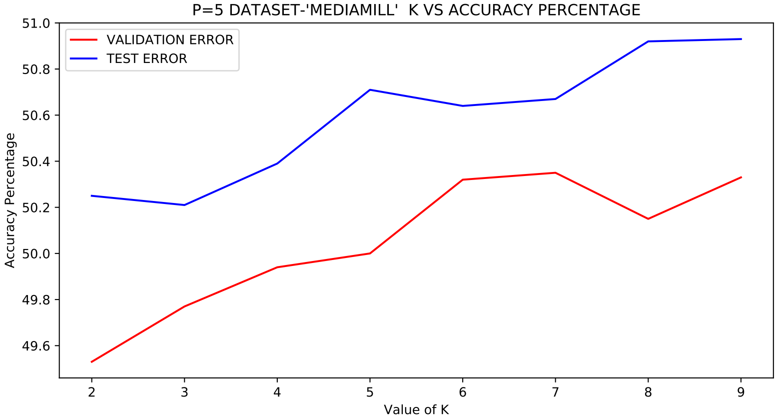
MEDIAMILL Dataset for P = 1



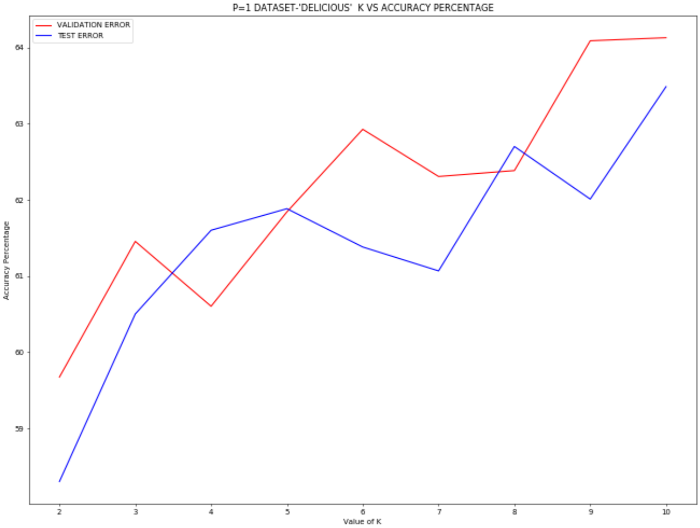
MEDIAMILL Dataset for P = 3



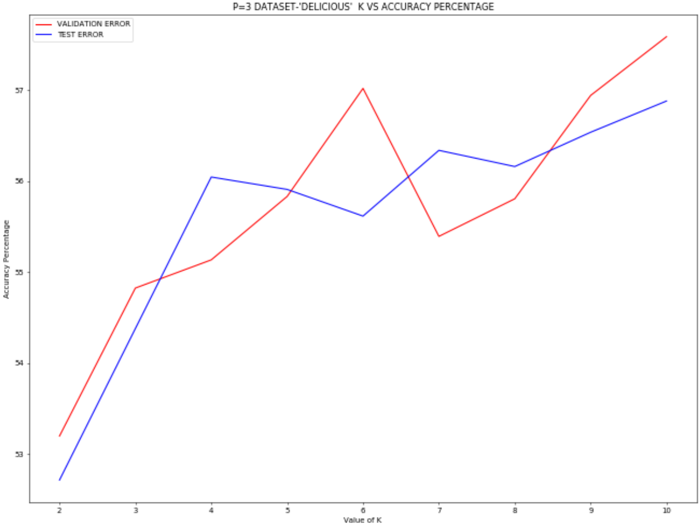
MEDIAMILL Dataset for $P = 5$



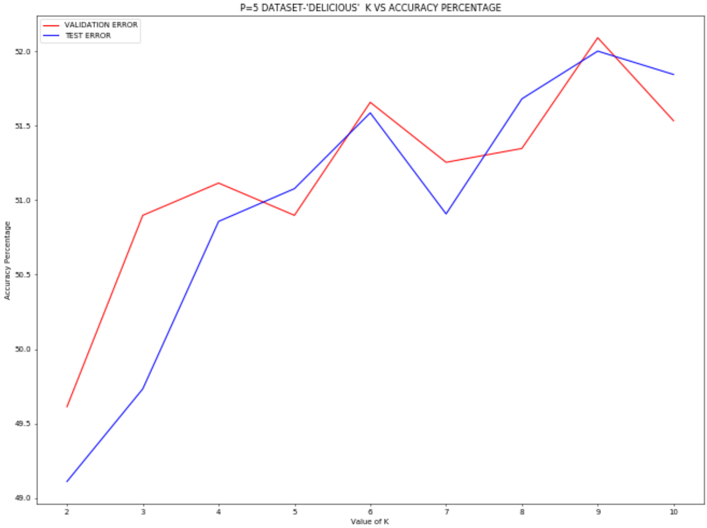
DELICIOUS Dataset for P = 1



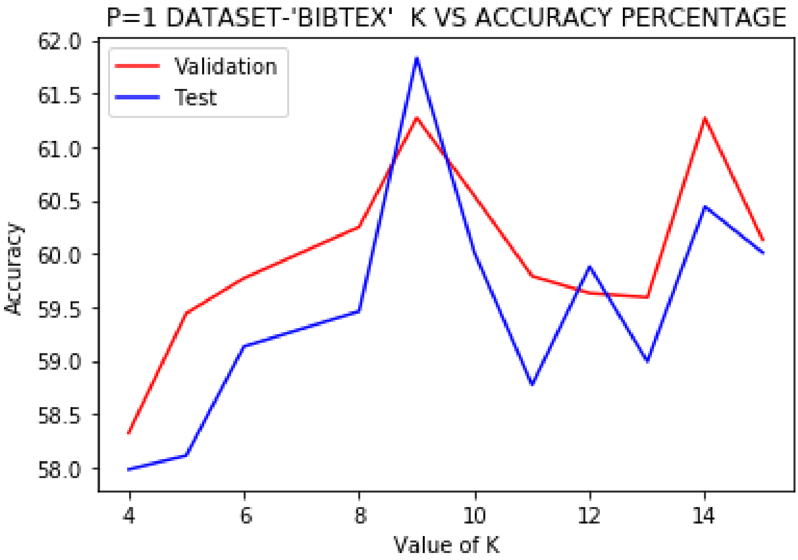
DELICIOUS Dataset for P = 3



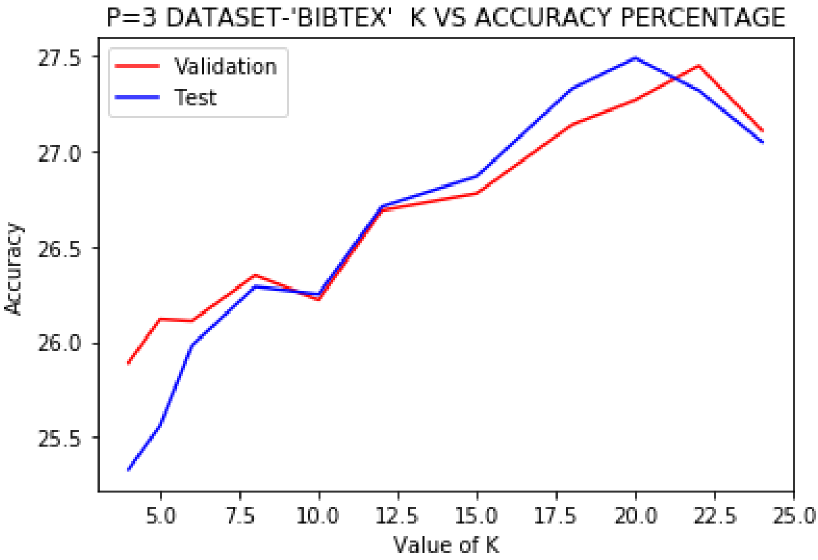
DELICIOUS Dataset for P = 5



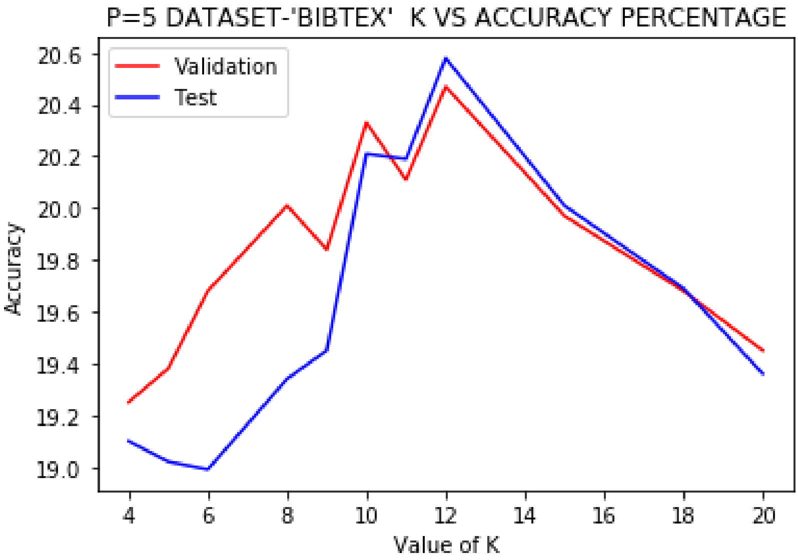
BIBTEX Dataset for P = 1



BIBTEX Dataset for P = 3



BIBTEX Dataset for P = 5



Conclusion

- We have performed validation with different set of values of hyper-parameters like number of clusters, number of row sampled, number of feature sampled and get the accuracy values.

Conclusion

- We have performed validation with different set of values of hyper-parameters like number of clusters, number of row sampled, number of feature sampled and get the accuracy values.
- Studied the behaviour of model with different values.

Conclusion

- We have performed validation with different set of values of hyper-parameters like number of clusters, number of row sampled, number of feature sampled and get the accuracy values.
- Studied the behaviour of model with different values.
- Also perform the testing on test data and get the accuracy of the mode.

Conclusion

- We have performed validation with different set of values of hyper-parameters like number of clusters, number of row sampled, number of feature sampled and get the accuracy values.
- Studied the behaviour of model with different values.
- Also perform the testing on test data and get the accuracy of the mode.
- We have seen that, with closer cluster number for all trees in decision forest keeping all parameters constant, increases the accuracy.

Thank You!