Big data project
Twitter cluster analysis

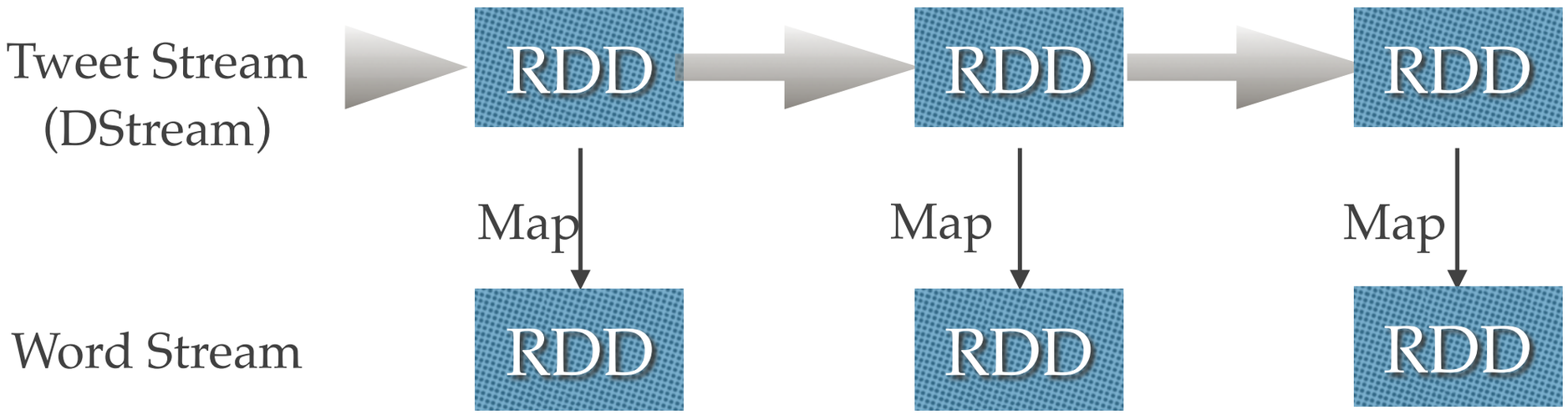
Mengchen Ding & Yuanzhi Yao

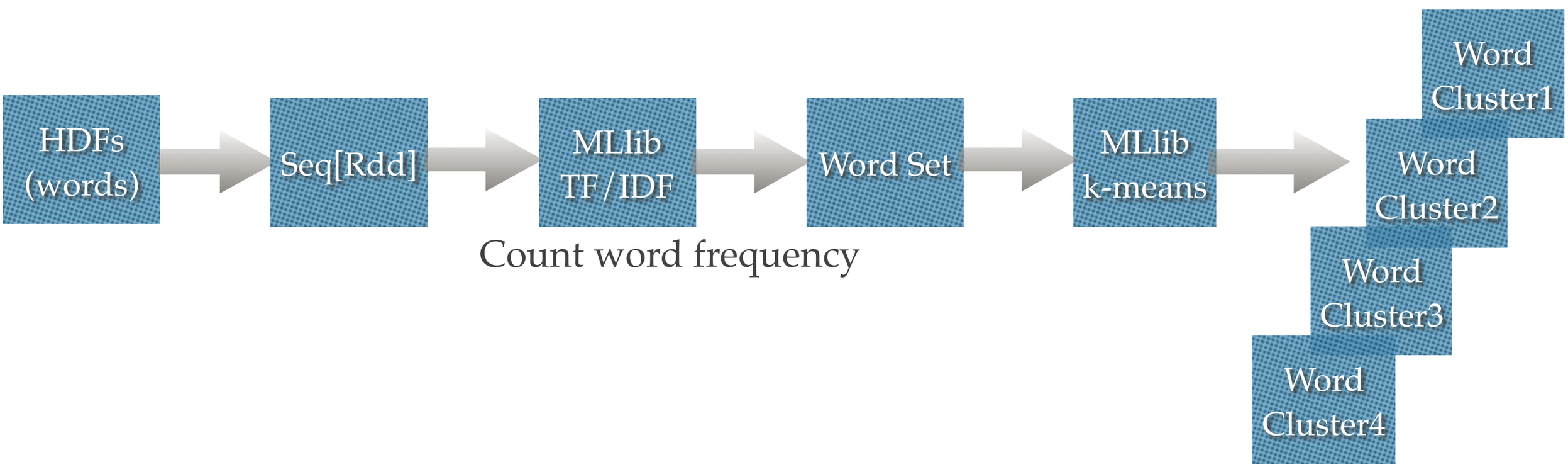
Project Outline

- Ingest tweets stream
- Parser and compute functional word frequency
- Data Analysis by clustering same topic data
- Algorithm: TfIdf, KMeans
- Tool: Scala & Spark

Methodology

- Spark Streaming+Twitter API
- RDD to store data
- Use Tokenizer to filter and process data, use Lazy Collection and Map transformation
- Algorithm: Tf/Idf & KMeans
- Tool: Spark MLlib


Tweet Stream
(DStream)
RDD
RDD
RDD
Map
Map
Map
Word Stream
RDD
RDD
RDD


HDFs
(words)
Seq[Rdd]
MLlib
TF/IDF
Word Set
MLlib
k-means
Word
Cluster1
Word
Cluster2
Word
Cluster3
Word
Cluster4
Count word frequency

Tokenizer and Parser

- Regular expression
- Add tag to tokenizer
- Normalize and filter data
- Parse tweet DStream

Tf-Idf & Clustering

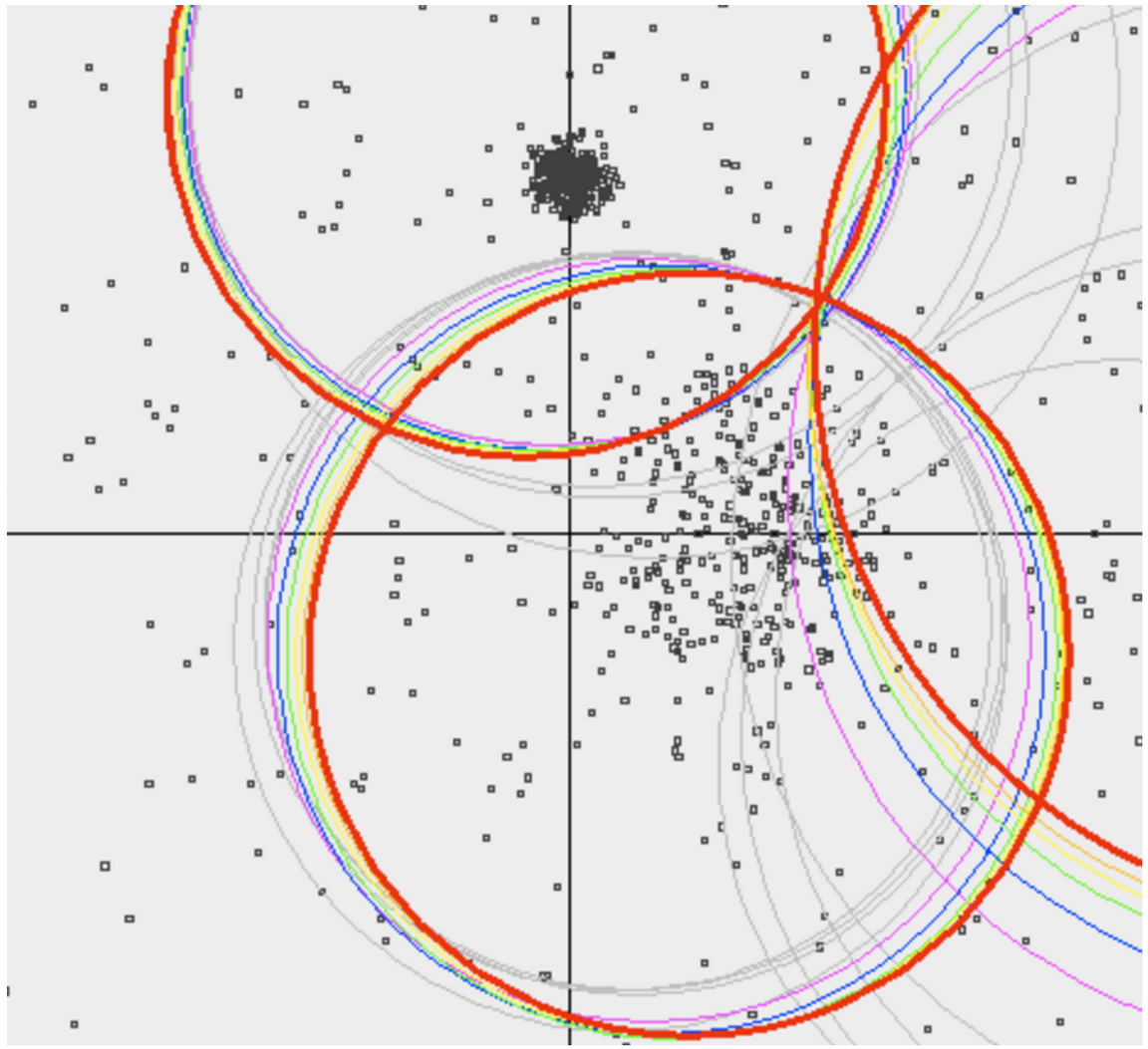
- Tf= word frequency in a tweet/ frequency of most frequent word
- Idf=Log(Number of Doc in a file/Number of file containing the word+1)
- Tf-Idf= Tf* Idf
- K-means: MLlib, train() method

Clustering Effect

Use computeCost() method to evaluate the clustering effect.

The smaller computeCost value, the better clustering.

Our result



Running Result

Application

- Capture topics in heat
- Analysis topic trends
- Serve for analysis models for predication

- Thank you!