Analyzing High-velocity Real-time data
Jinglun Li, Xiang Lin, Bina Ramamurthy, Ph.D.
Computer Science and Engineering Department, the State University of New York at Buffalo NY, 14260

Purpose and Goal
Big-data analysis is getting more attention with explosive growth in Internet and mobile applications. Big-data is defined by four V's (high) Volume, Velocity, Variety and Veracity. We focus on the data generated at high velocity. An example of this kind of data is the tweets sent out by twitter users. Our goal is to analyze the real-time twitter data to find whether the words that from people’s discussion can help us to predict the result of future events and the popular things in the society.

Introduction to MapReduce

1) Iterate over a large data set
2) Mapper: generated key-value pairs for all input
3) Shuffle and sort intermediate results
4) Reducer: All values with the same key are reduced together
5) Output final result

Implementation Detail

1) Collect twitter data by Python Twitter API.
2) Filter data by Stopwords list.
3) Run WordCount in MapReduce and sort output data by frequency of words.
4) Discover interesting words of persons or objects to analyze.
5) Visualize the result.

Issues and Challenges

• The raw data we get back will be encoded in JSON (JavaScript Object Notation) which enclosing complicated information in a platform-independent way. We need to re-process the JSON data to precisely extract what users post in each tweet.
• The words that twitter users post in each tweet contain many unnecessary stop words that need to be filtered out.
• Time complexity time is high because there are a lot of word pairs to sort and shuffle around. We need to improve the algorithm to reduce the complexity.

Words Co-occurrence

• Set data as co-occurrence context
• Term of co-occurrence matrix
  – M is square n x n matrix
  – n x V (the vocabulary size)
  – M[j]: number of times word w co-occurs with word wj in a specific context, such as sentence, paragraph,
• Purpose: Infer the interesting points about the events
  – “Pairs” and “Strips”

Java Application for Co-occurrence Output

1) Collect data by twitter API
2) Filter data by Stopwords list
3) Run MapReduce co-occurrence by using “Pairs”/“Strips” algorithm.
4) Output frequency of keyword co-occur with others words
5) Analyze relation among words that are of interest.
6) Run output on Java App

Conclusion

• Through implementing words count and co-occurrence in MapReduce with Hadoop, we are able to find the relation between words which will improve the precision of the prediction in future.
• Understand the sentiments and trends in the society by analyzing the real-time twitter data.

Acknowledgment and Reference

• Partially funded by the College of Science and Technology Entry Program (CSTEP) UNIVERSITY AT BUFFALO http://cads.buffalo.edu/cstep
• Partially funded by National Science Foundation Grant NSF-DUE-CCLI-0920335 http://www.coe.buffalo.edu/ ~bina/cse487/spring2013
• Center for Computational Research (CCR) UNIVERSITY AT BUFFALO http://ccr.buffalo.edu
• Four dimensions of Big Data: http://www-01.ibm.com/developerworks/data/bigdata/
• Twitter Application Programming Interface (API): https://dev.twitter.com/whatsnew
• Apache™ Hadoop: http://hadoop.apache.org