cobec CONSULTING

QUANTIFYING THE FUTURE

Growing Maintenance Costs: Understanding How Weather Impacts Maintenance

Bryan Anderson

May 14, 2019

Agenda

- Introduction
- Motivation
- Data Sources
- Analysis
  - Objectives
  - Methodology
  - Results
- Project Management
- Conclusion

Introduction

- Bryan "Fargo" Anderson
- Consultant with Cobec Consulting
  - Federal Aviation Administration (FAA)
- B.A. in Economics and Mathematics from Augsburg College - Minneapolis
- M.S. in Industrial & Systems Engineering from the College of Science & Engineering University of Minnesota – Twin Cities
- 8 years of Industry Experience
  - Industrial Engineering with Supervalu Inc

Motivation

- Cost estimation is often concerned with the amount of effort attributed to maintenance of its assets in the field
- FAA records all maintenance activities in a challenging central repository.
- New analysis methods are available to leverage this data source
- We want to know what activities are happening in order to leverage that in our estimation analysis

Remote Monitoring & Logging System

- Remote Monitoring & Logging System (RMLS) is the FAA's maintenance logging system
- Field technicians are alerted and tasked maintenance activities
- Technicians log a description of work performed
  - Free formed text
  - Validated: Code category, Facility Identification, System Type

Natural Language Processing

- There is abundance of data for analyses, but the problem is it is not in a useable format yet
- New technology and techniques are becoming more robust that enable the use of the previously un-useable datasets
- One of these new techniques and technology is Natural Language Processing (NLP)
- NLP is a field of science focused with translating unstructured text into a structured format for analysis

RMLS – Summary Examples

- [redacted] RWY 28 DME OTS for PM. 30 Min recall. ATSS/LSS rpts RTS, ZZZ ATC/NM.
- [redacted] PCL OTM for RRCS PMs. RWY 27 MALS/PAPI & & RWY 09 MALSR/PAPI will be placed in the setting that ATC prefers for duration of maint.
- MASS indicating soft alarm. LUID 282C Link Mer. Cleared.

Integrated Terminal Weather System

- Integrated Terminal Weather System (ITWS) is a FAA System Wide Information Management (SWIM) Java Messaging Service (JMS) product
- This report focused on 5 nautical mile precipitation data
- This data provides a grid snapshot of time of a geographical region
- Each grid gives an NWS level of precipitation
- Grids are averaged in a day to represent the amount of rainfall

Analysis Objectives

- The goal of this project is to:
  - Determine if weather and maintenance actives are related
  - Investigate word usage for re-occurring or common activities
- Two analyses are conducted:
  - Time series of average precipitation and log summary uniqueness
  - Topic modeling of RMLS logs

Time Series Analysis

- Time Series Analysis consists of RMLS and ITWS data sources.
- RMLS Data Processing
  1. Inputs: airport, start date, end date, and day-bin size
  2. The ratio of in-memory size to its zipped in-memory size of log summaries in the bin window is calculated
- ITWS Data Processing
  1. Inputs: airport, date
  2. Grid snapshot of each the sensor precipitation grid is averaged for each date

Time Series MSP

I I I

I x 3

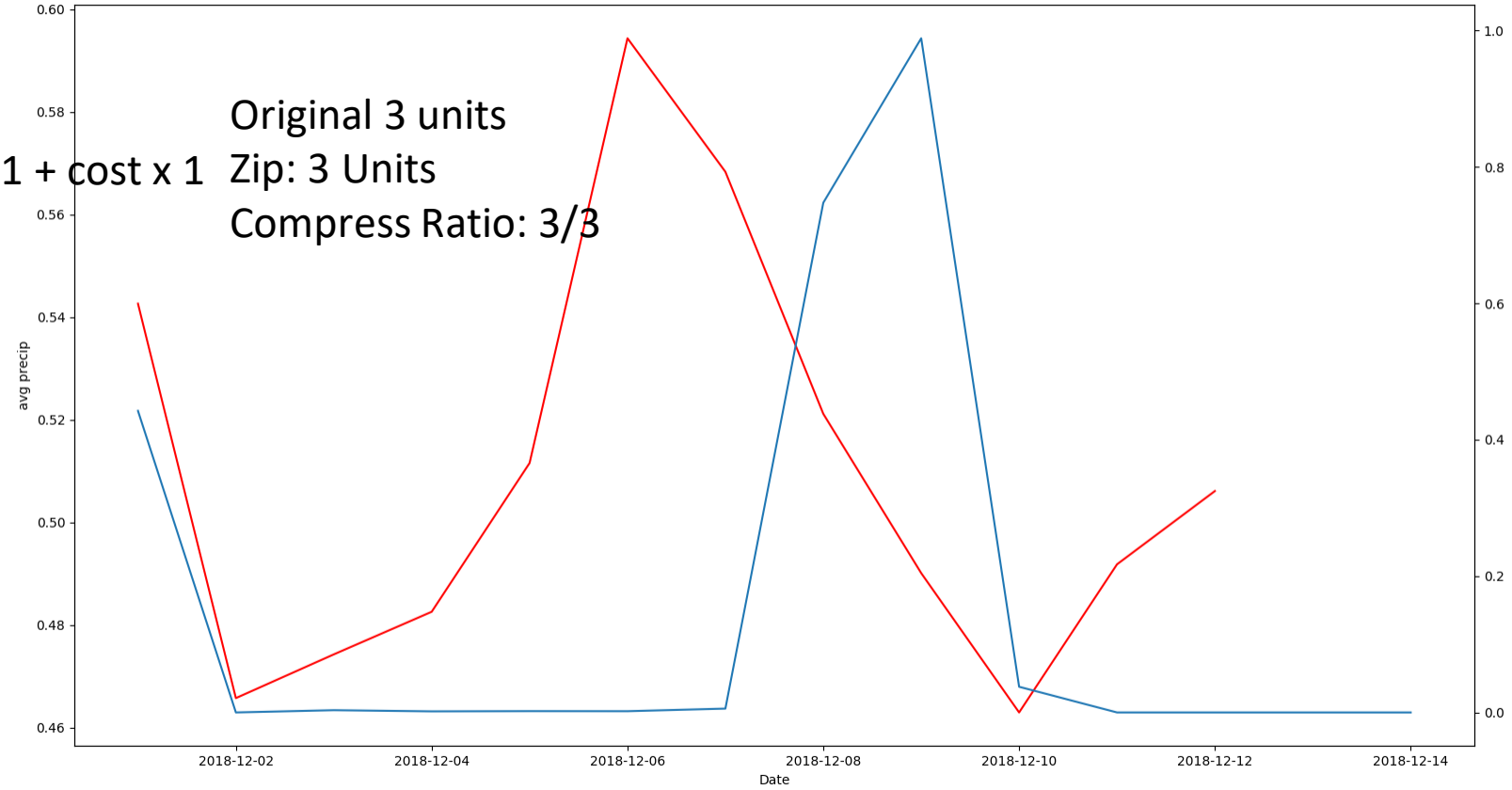
Original 3 units
Zip: 1 Unit
Compress Ratio: 1/3

Assume it costs 1 unit to keep a unique word (and We'll know where it goes and don't care how many).

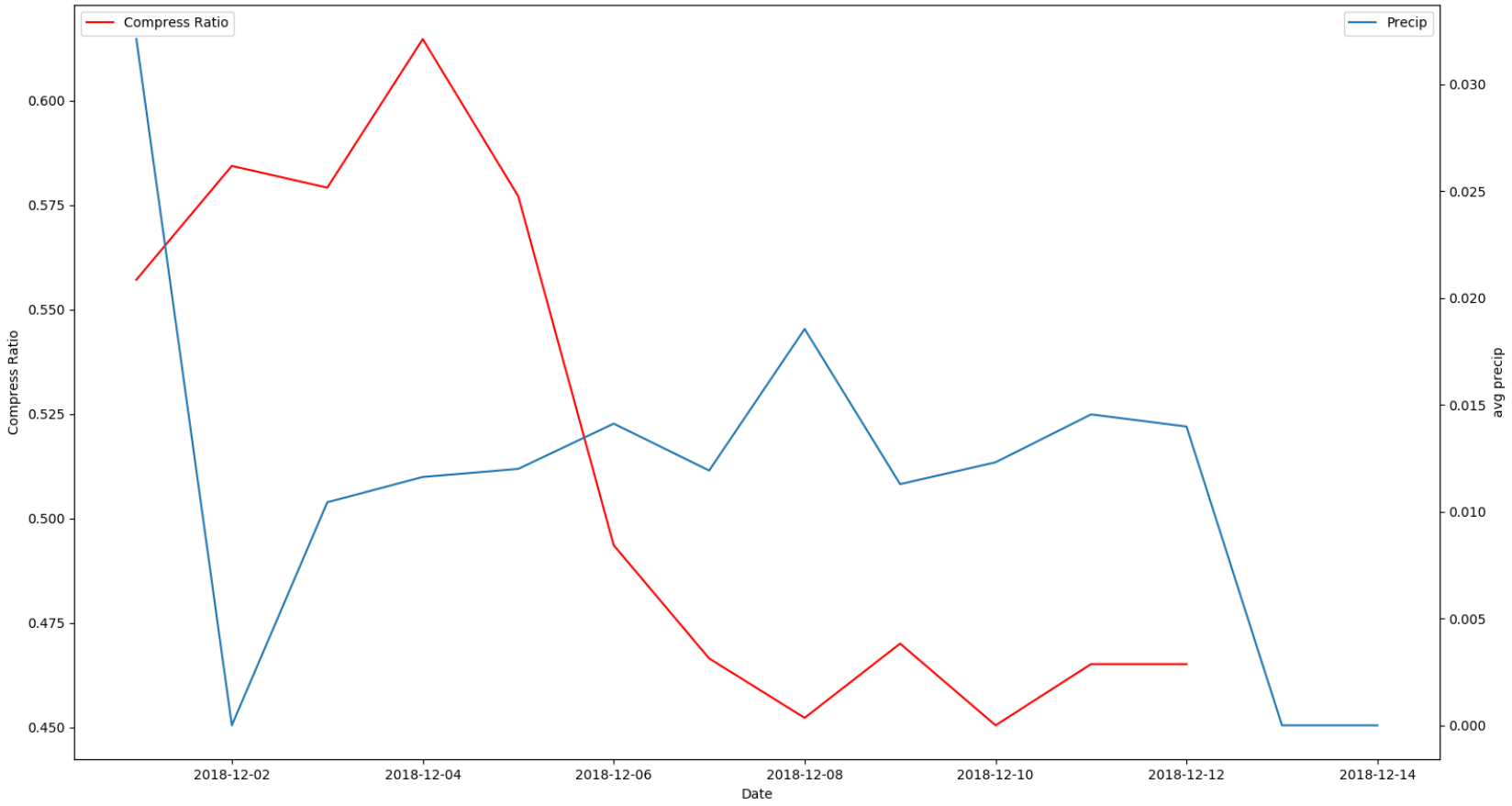
I like cost.

I x 1 + like x 1 + cost x 1

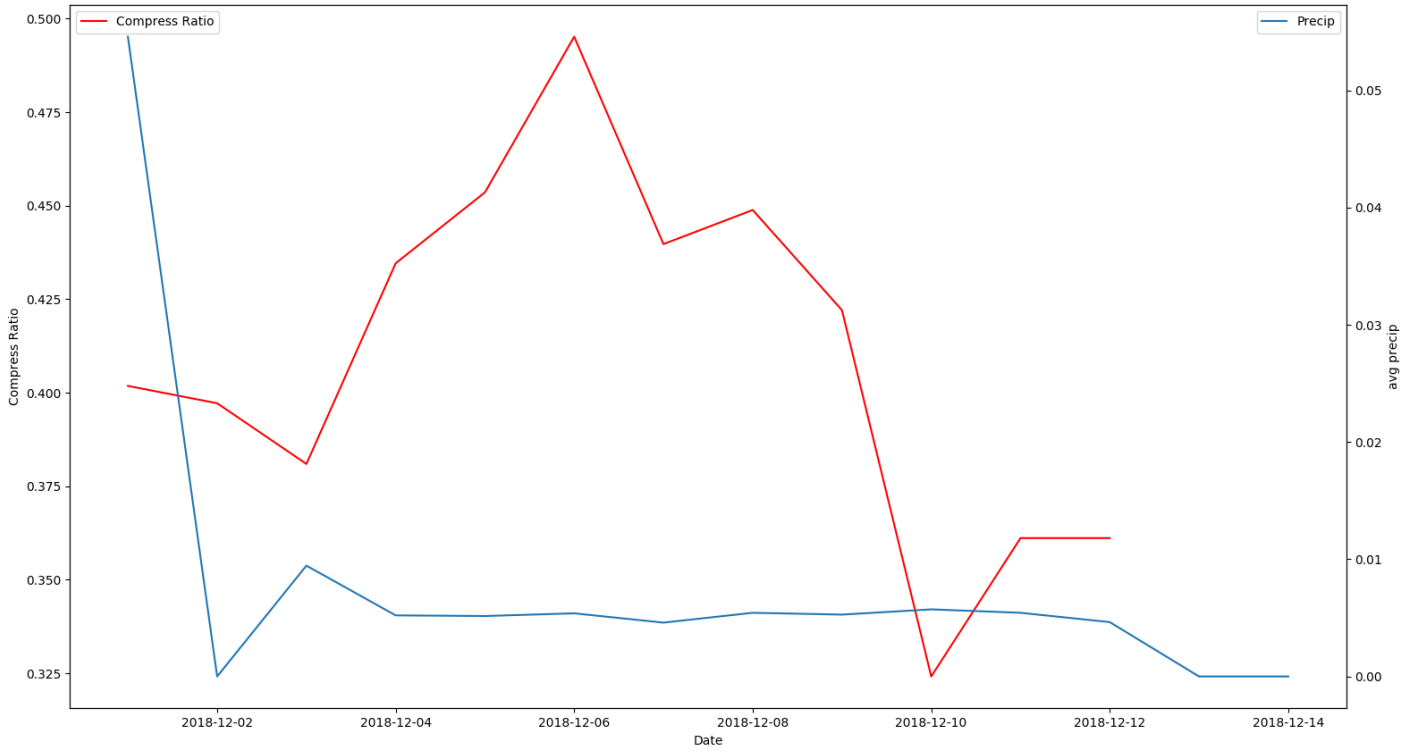
Original 3 units
Zip: 3 Units
Compress Ratio: 3/3



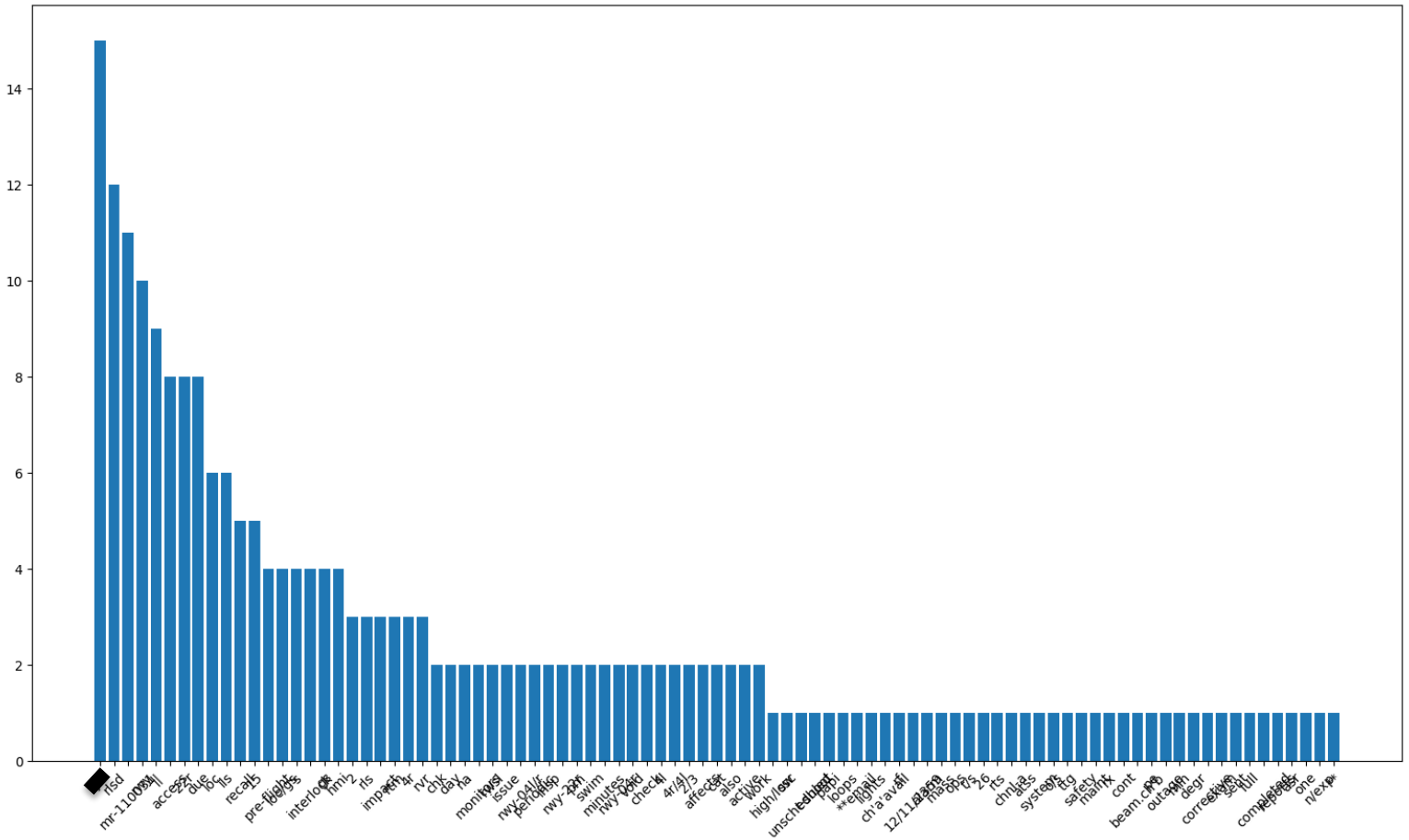
Time Series DCA



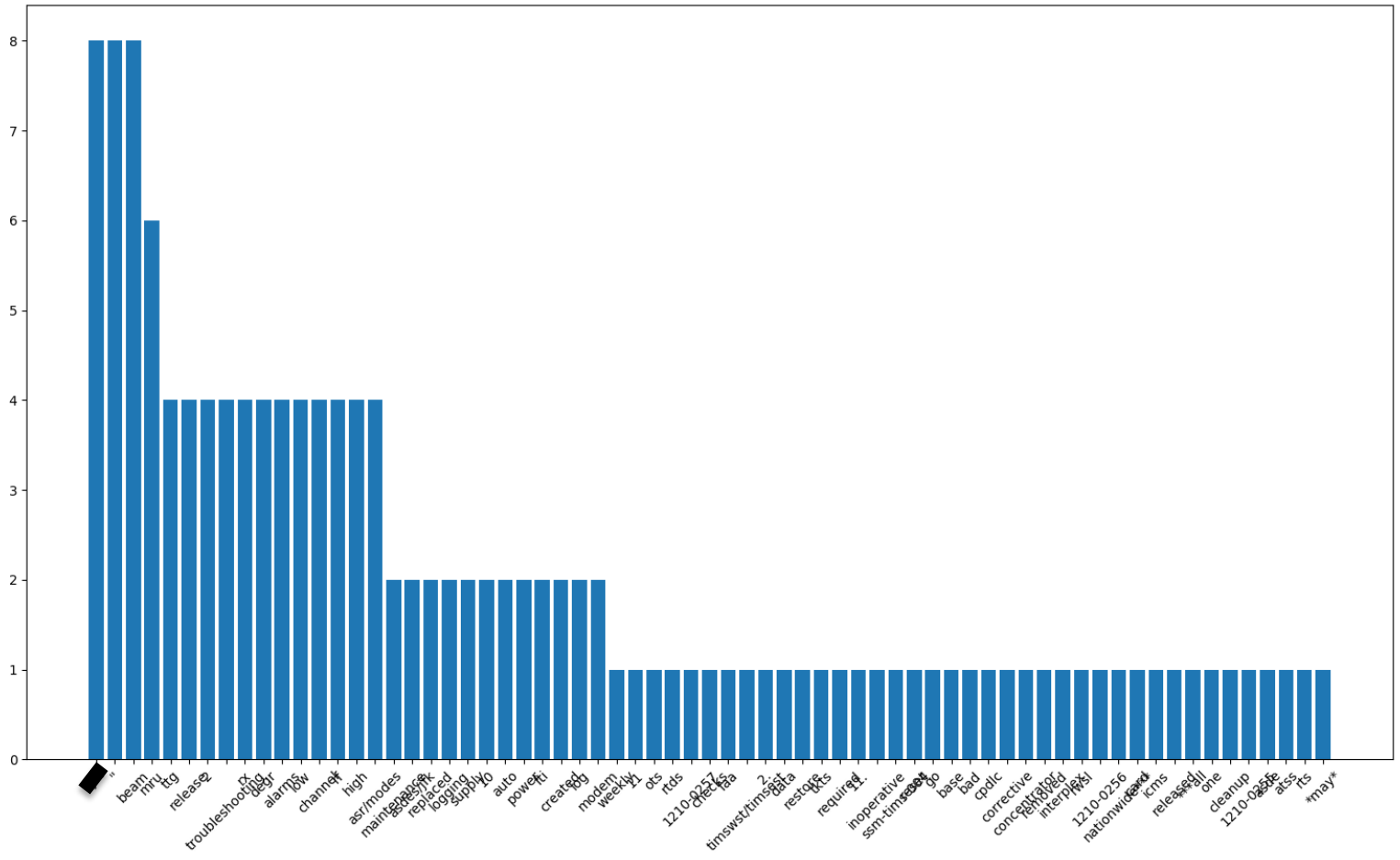
Time Series JFK



Frequency of Tokens - High



Frequency of Tokens - Low



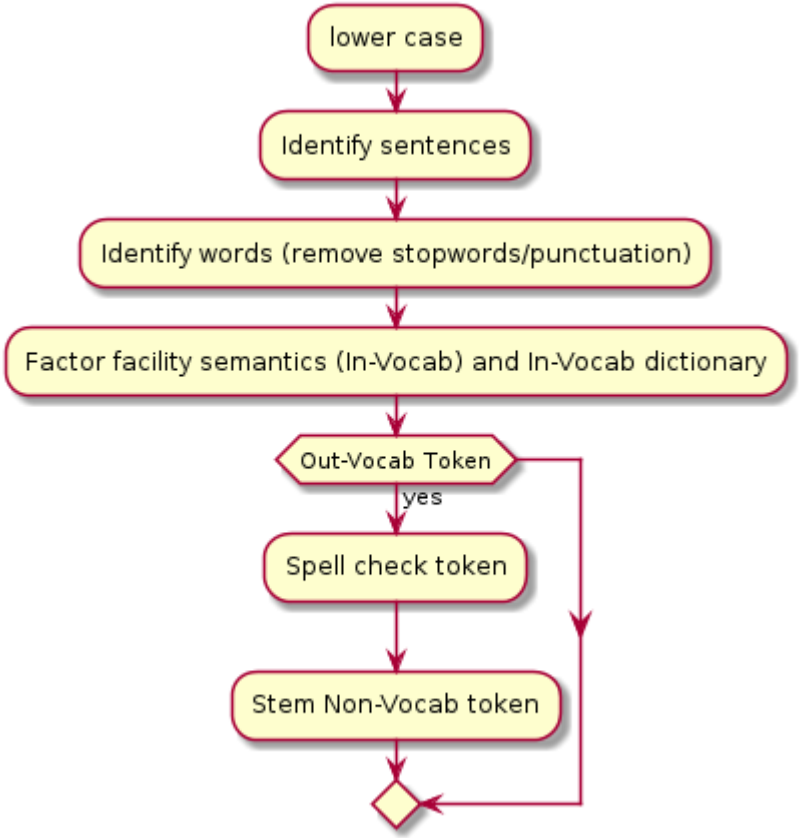
Topic Modeling - Overview

- Topic modeling is discovering abstract topics that occur in a collection of documents
- Latent Dirichlet Allocation (LDA) is a generative probabilistic model
  - Two step process to generate documents
  - Name is derived from discovering hidden variables with help from the Dirichlet distribution.
- LDA is a common technique in Topic Modeling

Topic Modeling – Text Normalization

- Inputs for LDA is Bag-of-Words (BOW) of each document
- BOW are count of tokens in a document
- Tokens are normalized words
- Normalization of words is the process to put unstructured text in a structured, clean, useable format for analysis

Text Modeling – Normalization Flow



Topic Modeling - Example

s1 I like ICEAA because it helps me cost estimte! The Minnesota Twins are the best team in baseball says COBEC/BA.

s2: ['i like iceaa because it helps me cost estimte!', 'the minnesota twins are the best team in baseball says cobec/ba.']

s3: ["(False, 'like')", "(False, 'iceaa')", "(False, 'helps')", "(False, 'cost')", "(False, 'estimte')", "(False, 'minnesota')", "(False, 'twins')", "(False, 'best')", "(False, 'team')", "(False, 'baseball')", "(False, 'says')", "(False, 'cobec/ba')"]

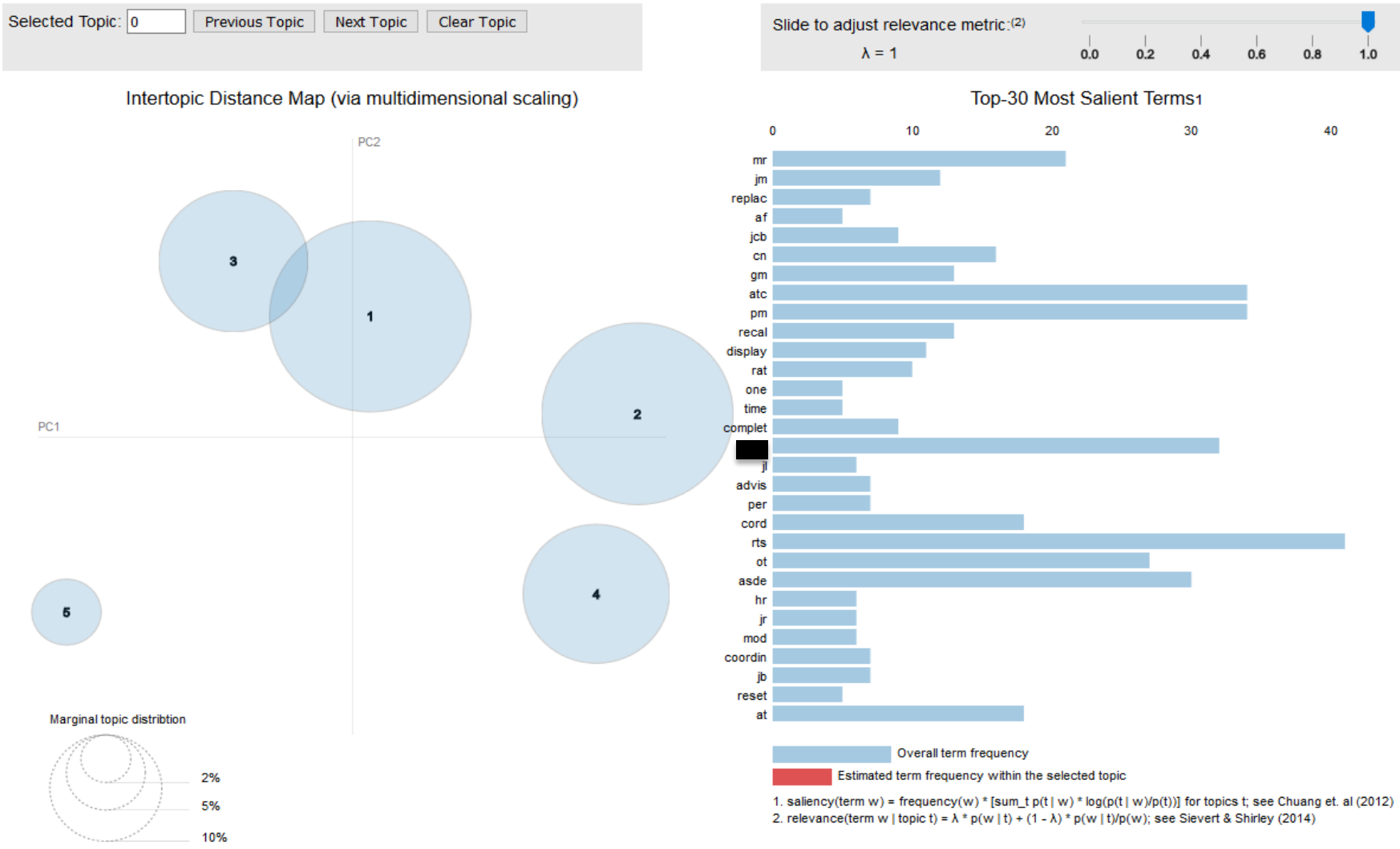
s4: ["(False, 'like')", "(False, 'iceaa')", "(False, 'helps')", "(False, 'cost')", "(False, 'estimte')", "(False, 'minnesota')", "(False, 'twins')", "(False, 'best')", "(False, 'team')", "(False, 'baseball')", "(False, 'says')", "(True, 'cobec')", "(True, 'ba')"]

s5: ["(False, 'like')", "(False, 'iceaa')", "(False, 'helps')", "(False, 'cost')", "(False, 'estimte')", "(False, 'minnesota')", "(False, 'twins')", "(False, 'best')", "(False, 'team')", "(False, 'baseball')", "(False, 'says')", "(True, 'cobec')", "(True, 'ba')"]

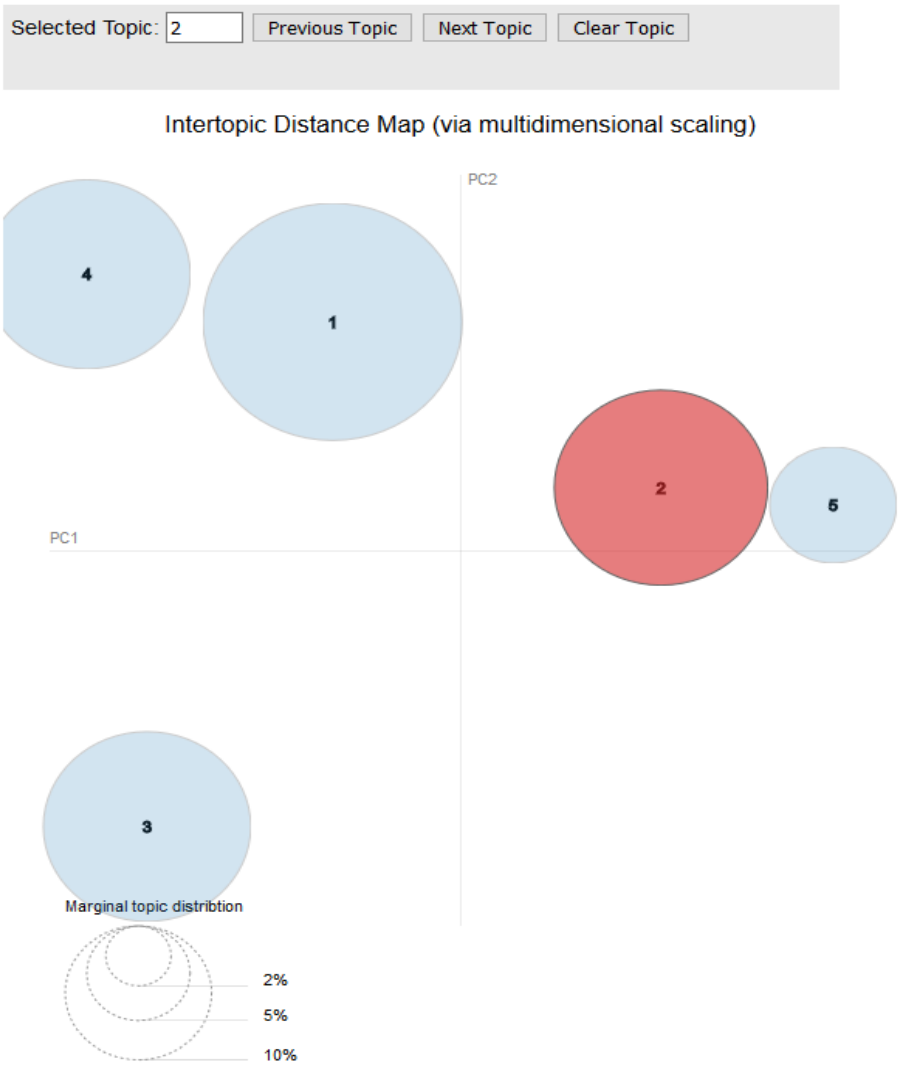
s6: ["(False, 'like')", "(False, 'idea')", "(False, 'helps')", "(False, 'cost')", "(False, 'estimate')", "(False, 'minnesota')", "(False, 'twins')", "(False, 'best')", "(False, 'team')", "(False, 'baseball')", "(False, 'says')", "(True, 'cobec')", "(True, 'ba')"]

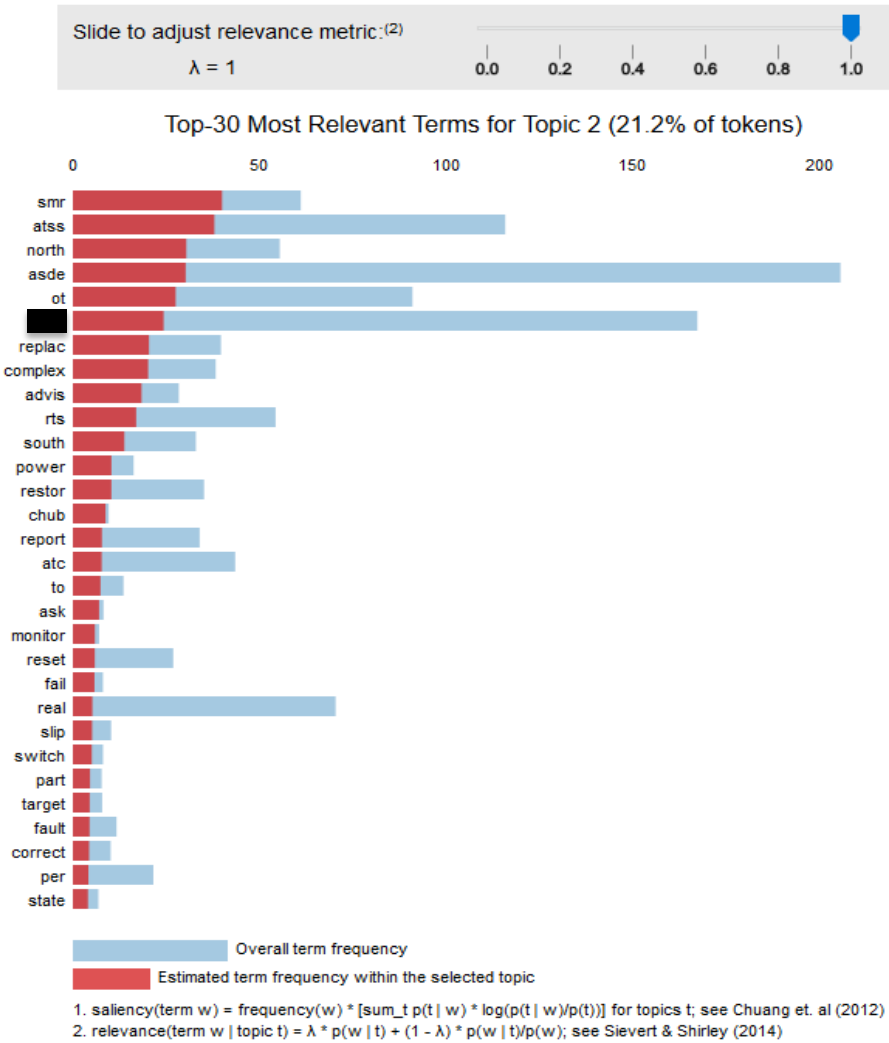
s7: ["(False, 'like')", "(False, 'idea')", "(False, 'help')", "(False, 'cost')", "(False, 'estim')", "(False, 'minnesota')", "(False, 'twin')", "(False, 'best')", "(False, 'team')", "(False, 'basebal')", "(False, 'say')", "(True, 'cobec')", "(True, 'ba')"]

Airport 1 - 5 Topics

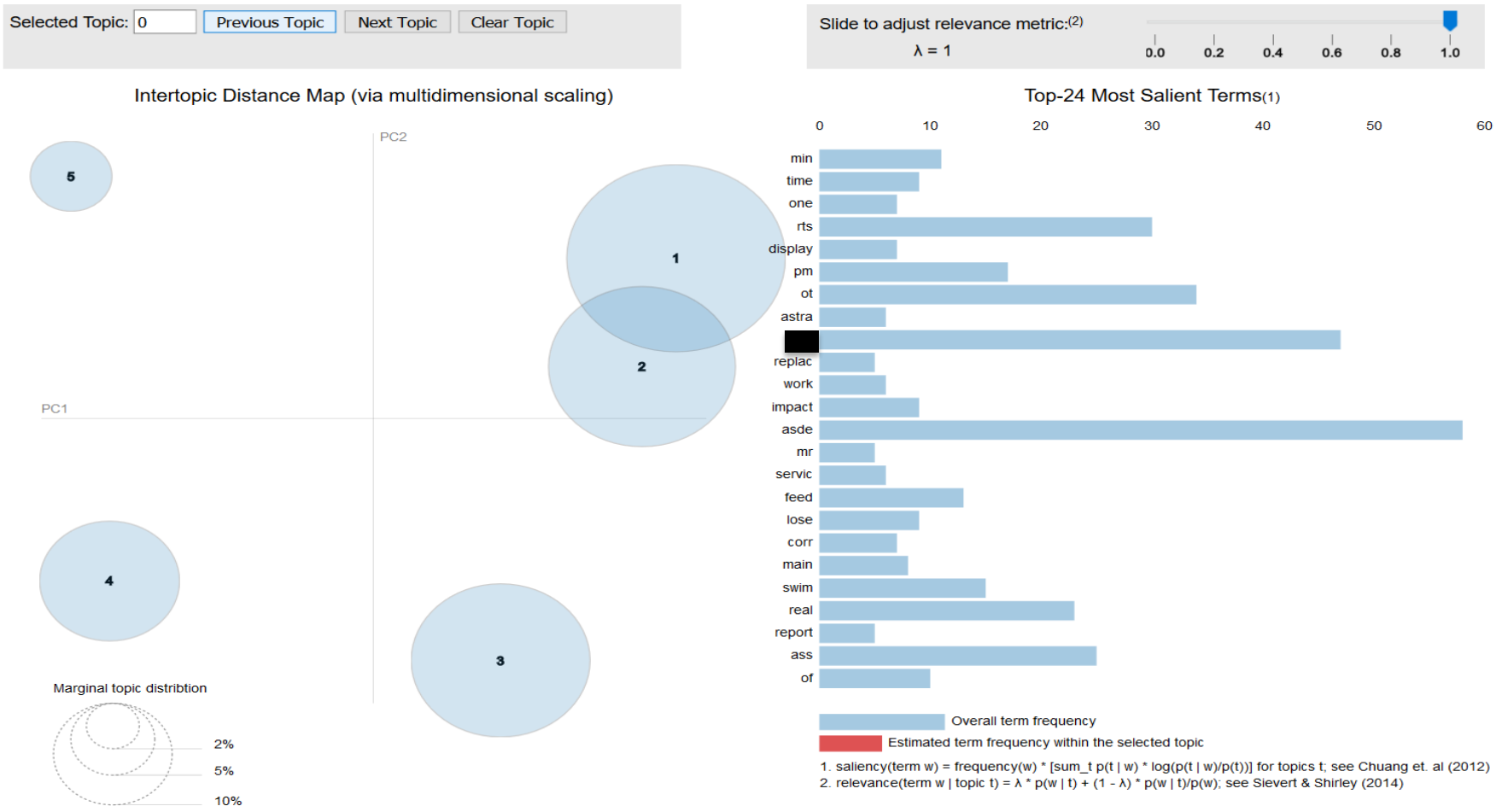


Airport 2 – 5 Topics





Airport 3 - 5 Topics



Project Management

- Expectations are broader than just analyzing data; other required roles:
  - Subject Matter Experts
  - Data Engineers
  - System Architects
  - Visualization Developers/Story Tellers
- Each of these roles have specialized tools and processes to support the role's function
- Determining how well we define these roles and which tool/processes to use depends on the project
- Two types of projects:
  - Long-Term
  - Short-Term

Long Term Management

- Characteristics of Long Term projects:
  - Work exceeds 6 months
  - Multiple people working on the project
- More effort may be needed in project management
  - Robust Issue Management System
  - Configuration Management
    - Team members need to have standard policies:
      - Naming/file conventions
      - Use GIT (or SVN)
      - Code line policies
  - Folder structure by project components
  - Optimization of code is just as important as getting results

Examples of Code Line Policies

- Code line policies help keep files organized and projects from becoming dirty
- Example policy - Before every commit:
  - Code passes manual and automated tests
  - Pull from target branch before committing
  - Reference issue ID what caused changed in comment of commit

Example of Large Folder Structure

- /root
  - /core
    - /database
    - /logger
  - /parser
  - /retriever
  - /database
    - /schema1
  - /documents
  - /reference
  - Readme.md

Short Term Management

- Short term projects are less than 6 months
  - Quick analysis, usually < 2-4 weeks
- Issue management can be less structured
  - Team members must be communicating continuously during the project
  - Tools: Microsoft Teams, Slack, or Trello
- Configuration Management:
  - Tools like GIT can be too cumbersome, using services like OneDrive, Dropbox, and Google Drive are sufficient
  - Best Practices: Standardized folder structures when working multiple analysis projects
  - For exploratory analysis, keep number of files small and write code snippets for running by hand in console
  - Focus on results; less importance on optimization

Example of Small Project Folder Structure

- /src
  - /python
  - /sql
  - /tableau
- /data
  - /outside
  - /derived
- /document
  - /figures
- /reference
- Readme.md

This Project

- This project was a small project built on the shoulders of two large projects
- Tools Used:
  - OneDrive
  - Python, SQL (Postgres/Oracle), Emacs
    - 3rd party libraries: nltk, pandas, postgis, genism, pyLDAvis
- Folder Structure:
  - /data
    - /outside
      - /rmls
    - Derived
      - /lda
      - Vocab_list.txt
  - /documents
    - /figs
  - /references
  - /source
    - /sql
      - Or_explore.sql
      - Pg_explore.sql
      - Pg_artItws_function_calcPrecip5nm_avg.sql
    - /python
      - Db_mgr.py
      - Explore.py
  - Diary.org

Conclusion

- Time Series analysis of RMLS and ITWS
- Frequency diagrams of words for high and low precipitation times
- Topic modeling of log messages
- Project management of data science projects require several roles
- Tools and processes depends on the size and complexity of the project

Questions