

---

# Tracking FEMA

---

Kevin Kays, Emily Maier, Tyler Leskanic, Seth Cannon  
Spring 2015 CS4624 Professor Edward Fox  
Department of Computer Science at Virginia Tech

---

# Background

---

- Client: Seungwon Yang, postdoctoral researcher at GMU.
  - Goal: Using articles from CTRnet/IDEAL collections, graph trends for emergency management agencies.
-

# Deliverables

---

- A website showing interactive visualizations
  - All the scripts used for processing and visualizing data (for future work)
  - Git: <https://git.cs.vt.edu/kkays/trackingfema>
-

# Implementation

---

Divided Implementation into four parts:

- Parsing (Emily)
  - Processing (Kevin)
  - Visualizations (Seth)
  - Content Management System (Tyler)
-

# Parsing

---

- Python scripts to download all of the articles we're using
  - Downloaded article base used iteratively in order to create output files
  - Output title, subtitle, author, date, body text, and anchor text
-

# Parsing

---

- Python using the BeautifulSoup library
  - It can pull HTML elements out of an article using almost any imaginable criteria
  - Each metadata type has set of functions that examine the document
  - Functions run successively until match found
-

# Processing

---

- Python and NLTK ([www.nltk.org](http://www.nltk.org))
  - Run as a timed job in the background
  - Two major components:
    - Raw article processor creates “Mention Files”
    - Summary file creator organizes information
-

# Processing

---

- **Raw Article Processor**
    - Finds entity names in articles
    - Records count of entity names, organized by entity type (location, person, agency)
  - **Summary File Creator**
    - Creates single “Summary Files” for visualizing
    - “Locations by Disaster”, “Related people and agencies”, etc.
-



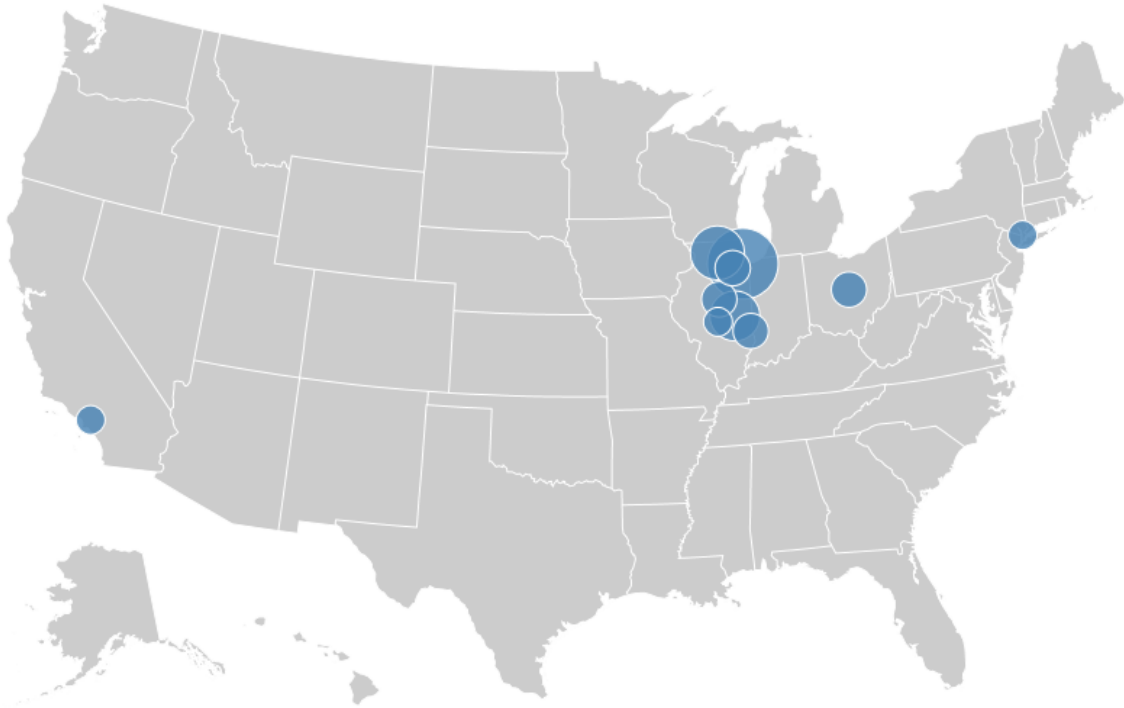
# Visualization

---

- Graphs using D3
    - JavaScript framework
  - Data Representation
    - Location by disaster
    - People involved by disaster
    - Organizations involved by disaster
-

# Northern Illinois University Shooting

---

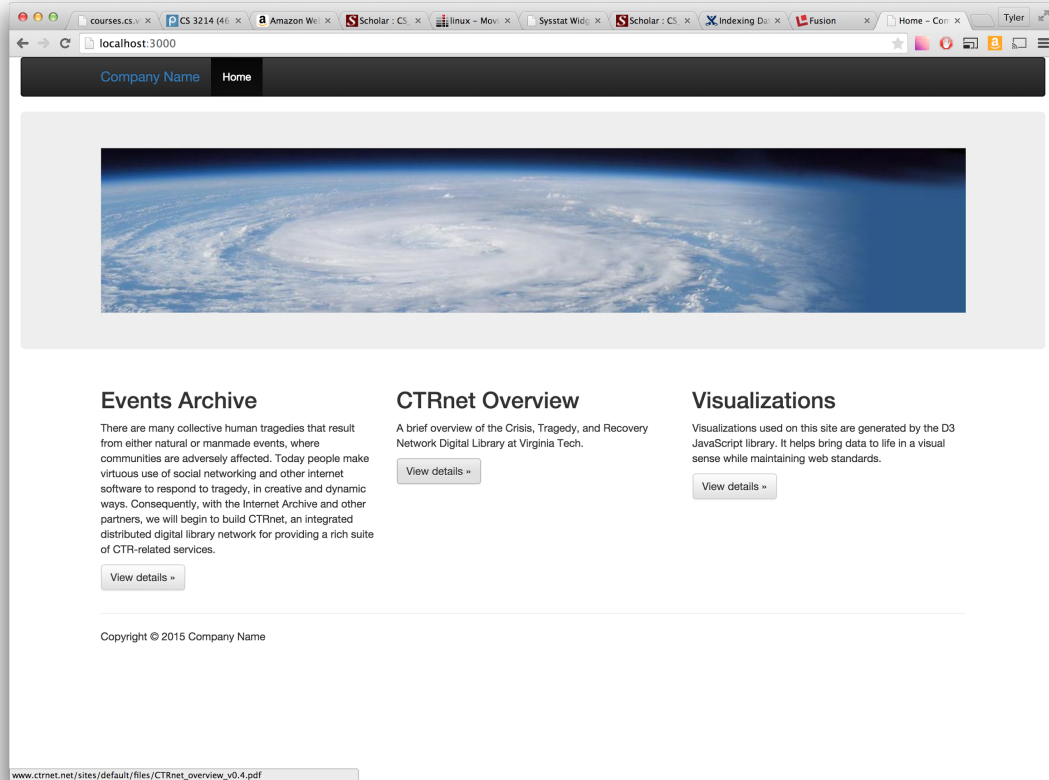


# CMS

---

- RefineryCMS 2.1.5
    - Provide user friendly way to edit content of the site.
    - Built in webserver since it's built on Rails.
    - Future possibility: execute parsing/data processing scripts through admin tool.
  - Bootstrap Front End Framework
    - Provides out of the box styling, that can be customized.
    - Enabled for fast responsive design so site can be used from a mobile phone
-

# Homepage Layout



# Suggested Improvements

---

- Automate loading data into the visualizations
  - Process more types of information (dollar amounts, etc)
  - Add more metadata types to the parser
  - Use a different parser more like how humans read
-

# What we learned

---

- NLTK is complicated
  - Have more than one person per role
-

# Acknowledgements

---

- Seungwon Yang, our client, for the great opportunity, project idea, and suggestions.
  - Dr. Fox and Liangzhe Chen for the quick feedback and guidance with our report
-

---

# Thank You

---

Questions?

---