# Text Analytics Project Manifest

Ben Ballard (bkq5nt)

Exploratory Text Analytics - Final Project

## Project Overview

This project analyzes public safety communications through text analytics, focusing on daily police reports from the Annapolis Police Department. The analysis spans approximately seven years of reports, examining patterns in public reporting.

## Data Collection and Sources

**Word Docs included:**

* **Final Report:** WordDocument

**Core Files:**

* **Final Project Ballard:** Jupyter Notebook
* **Data Scrapper:** Jupyter Notebook

**Data Files:**

* **data/Annapolis\_PD\_Raw:** raw csv file in data sub folder. (Output from Data\_Srapper)

### Primary Source Data (F0)

* **Provenance**: Web scraping of Annapolis Police Department's daily public communications
* **Source URL**: [https://www.annapolis.gov/list.aspx?PRVMSG=253](https://www.annapolis.gov/list.aspx?PRVMSG=253%20%20)
* **Coverage**: Approximately 2,000 reports spanning 7 years
* **Notebook to scrape data:** Data\_Scrapper.ipynb

**F0 Data Structure**

* **document\_id**: Unique identifier for each document
* **title**: Title of the daily report
* **link**: Source URL for original report
* **date\_sent**: Publication timestamp
* **content**: Complete report text

### F1: Machine Learning Corpus Format

The first transformation separates individual incidents from daily reports, creating minimal discursive units. Use regular expressions to identify and extract incident reports, preserving context and metadata.

**Structure:**

* **document\_id:** Links to original F0 document
* **incident\_id:** Specific incident identifier when available
* **report\_date:** Filing timestamp
* **incident\_date:** Occurrence timestamp
* **text:** Incident description/report content
* **link:** Reference to source document
* **word\_count:** Token count
* **doc\_length:** Character count

### F2: Standard Text Analytic Data Model (STADM)

The STADM implementation divides data into three normalized tables that maintain referential integrity through consistent indexing.

**LIBRARY Table (F2\_Doc\_Table):**

* Primary document metadata
* Maintains incident and report date relationships

A screenshot of a computer

Description automatically generated

**TOKEN Table (F2\_Token\_Table):**

* Records individual word occurrences
* Preserves document position and sentence structure
* Maintains links to source documents

A screenshot of a computer

Description automatically generated

**VOCAB Table (F2\_Term\_Table):**

* Unique term dictionary
* Document and total frequency counts
* Term identification scheme

A screenshot of a computer

Description automatically generated

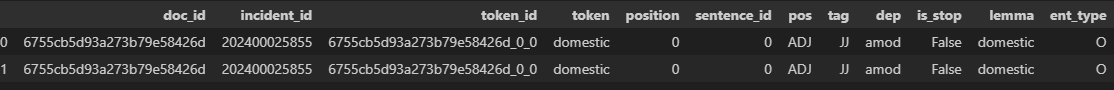
### F3: NLP Annotated Format

Enrich token and vocabulary data with linguistic annotations using NLTK and spaCy libraries. These annotations add layers of linguistic understanding to the text.

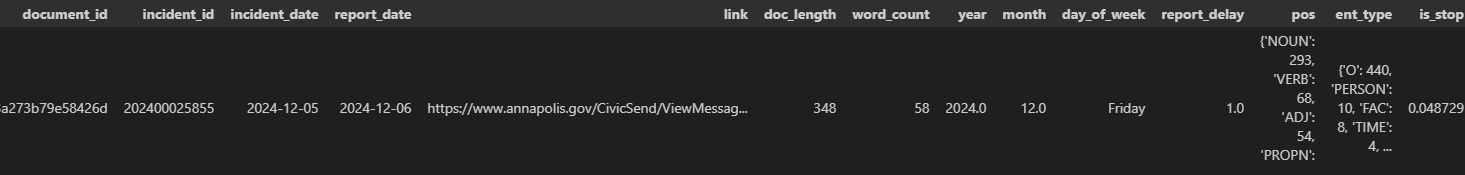
**Annotations Include:**

* Parts of speech tagging
* Named entity recognition
* Entity type classification
* Dependency parsing
* Sentiment scoring
* Lemmatization

**Token F3**



**Library F3**



### F4: Vector Space Representations

The vector space model transforms text into numerical representations suitable for mathematical analysis. Implement TF-IDF vectorization to capture term importance across documents.

**Implementation**:

* TF-IDF vectorization
* Document-term matrix creation
* Term frequency normalization
* Dimensionality: (2885, 10219)
* Storage: Sparse matrix format

### F5: Analytical Models

Final stage applies several analytical techniques to uncover patterns:

**Generated Analyses:**

* Topic modeling (LDA w/ 5 topics identified)
* Principal Component Analysis: identifying variation in reports
* Word embeddings: semantic relationship between terms
* Sentiment analysis results: document and term level scores

### Processing Notes

**Data Quality Considerations**

* Incident reports extracted using regex pattern matching
* Text cleaning removes HTML artifacts and formatting
* Some reports mix incident descriptions with announcements
* Temporal gaps between incident and report dates vary
* Web scraping artifacts required additional cleaning

**Usage Guidelines**

* Dataset represents public communications rather than complete incident records
* Temporal analysis should consider reporting delays
* Community announcements are intermingled with incident reports
* Location information preserved at block level for privacy

**Technical Implementation**

* Regular expressions for incident extraction
* NLTK and spaCy for linguistic annotation
* scikit-learn for vectorization and modeling

**File Organization**

All data files maintain consistent indexing schemes allowing reconstruction of relationships between different analytical stages. The OCHO (One Column Header Only) principle is maintained throughout the pipeline to ensure data integrity.