**HHS: Working Smarter, Not Harder with AI and NLP**

Civic Digital Fellow Presentation to HHS CTO – July 2019

Alex Curtis - M.S. in Transportation Planning and Engineering at New York University

Email: [imcurtis@gmail.com](mailto:imcurtis@gmail.com) or [Alexander.Curtis@hhs.gov](mailto:Alexander.Curtis@hhs.gov)

**Problem**: HHS receives thousands of public comments and unstructured text from rule making and its hundreds of FACA working groups, like the Tick-Borne Disease Working Group (TBDWG). This results in 1000s of submissions that HHS manually reviews, which cost time, resources, and taxpayer dollars.

**Proposed Solution**: Emerging technology solutions exist. Artificial intelligence (AI) and natural language processing (NLP), for example, can mine millions of public comments and efficiently distill the textual data to identify key themes/topics as outputs.

**Background**: AI and NLP – Artificial intelligence, or AI, is intelligence demonstrated by **machines** as opposed to the **natural intelligence** of humans. Another way of defining the term “AI” could be that it is used to describe machines that mimic certain cognitive functions of the human mind, like problem solving.

NLP is a component of AI that is concerned with the ability of a computer program to understand human language as it is spoken. NLP is split into two general techniques: semantic and syntax analysis. Semantic techniques apply algorithms to understand meaning and structure of sentences, while syntax techniques are used to assess meaning from a language based on grammatical rules.

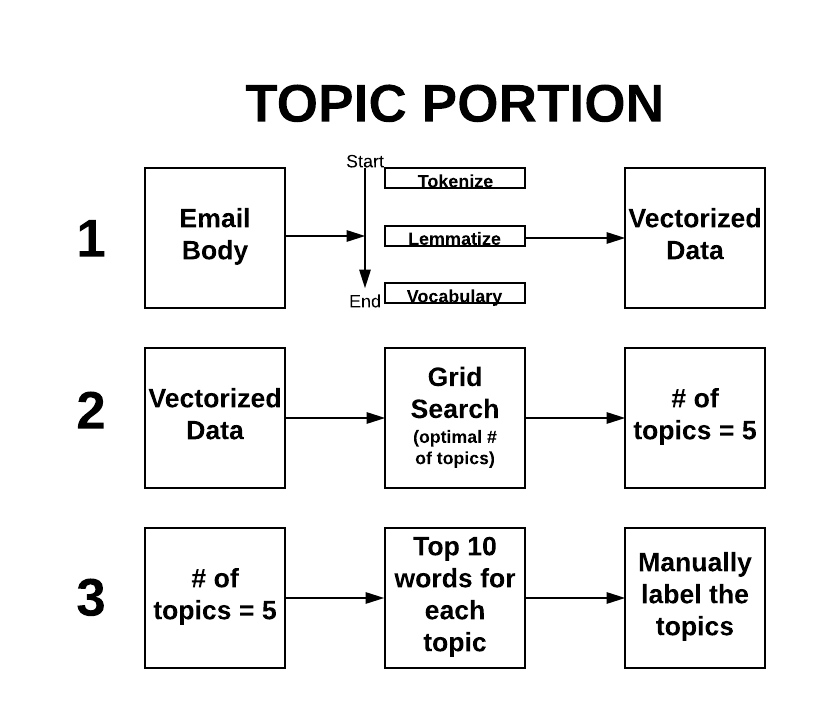
**Pilot Summary Overview**: Over 10 weeks in summer 2019, Civic Digital Fellows in the HHS Office of the CTO explored how HHS can efficiently advance its mission by harnessing existing technologies, specifically NLP, to digest and synthesize large quantities of unstructured text/data. NLP has the flexibility to be applied to any large text repositories, yet this proof-of-concept pilot used the written public comments submitted to the TBDWG as input data.

Figure 1. Topic Portion. Diagram that explains the flow of information from text data to resulting topic

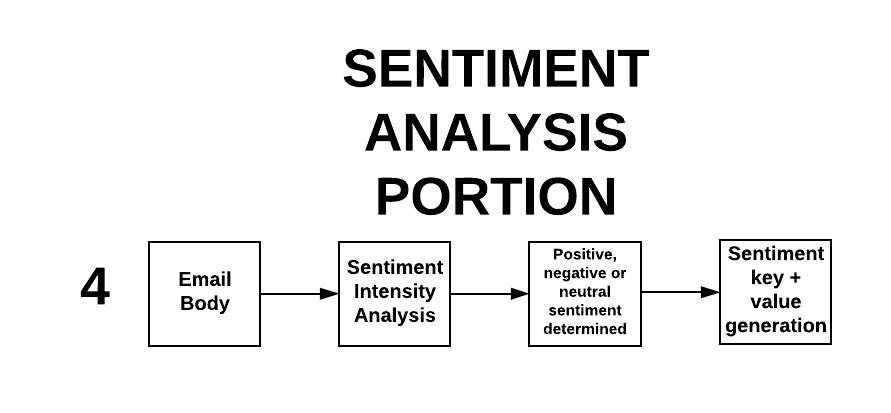


Figure . Sentiment Analysis Portion. Diagram that explains flow of information, from email body to sentiment (positive, negative or neutral) generation

For proof-of-concept, I prototyped an automation process for analyzing public written comments (refer to figures 1 and 2 above). First, I ingested raw and unstructured text into an NLP algorithm that separated the text data into different topics. Input data included over 900 comments across almost 700 pages of text. Requested outputs included key topics/themes that surface from the text, and the distribution of sentiments across documents. What resulted the analysis were four distinct topics, as well as sentiment scores for each document. To determine topic names, I ran a script that produced the top 10 words the occurred in each topic cluster. Then, using contextual knowledge, each topic was named.

To explore how public comments and sentiment changed over time, I divided comments by two different time windows—before and after November 15, 2018, which was the publication date of the TBDWG 2018 Report to Congress. This temporal division and analysis helped to determine any significant change in sentiment distribution based on that milestone.

**Outputs with Recommendations**:

This pilot digested 946 documents, at over 1.6 million characters. The top four topics that emerged from the written public comments, with their respective distribution of positive, negative and neutral sentiments are shown in the table below. How the sentiment of each document was determined is explained in Appendix A.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Table. 1: Sentiment Analysis Results** | | | | | | |
| **Theme** | **Positive Before 11/15/18** | **Neutral Before 11/15/18** | **Negative Before**  **11/15/18** | **Positive After 11/15/18** | **Neutral After 11/15/18** | **Negative After**  **11/15/18** |
| Patient experience | 32% | 21% | 47% | 13% | 17% | 70% |
| Public engagement | 56% | 26% | 18% | 56% | 24% | 20% |
| Physical manifestation of disease | 52% | 36% | 12% | 0% | 100% | 0% |
| Doctor-patient relationship | 49% | 21% | 30% | 33.3% | 33.3% | 33.3% |

Recommendation 1 – Following the completion of this prototype, Civic Digital Fellows will submit a report to HHS CTO with recommendations on the long-term viability of this type of NLP project, including: how to structure the input data and produce output data through automation, as well as finding and applying different methods of sentiment analysis.

Recommendation 2 – Conduct a rigorous assessment on how this NLP technology may be applied to create HHS efficiencies and cost savings. Assess interagency need and technology potential within federal government, as well as the current use and best practices from industry.

Recommendation 3 – Submit 10x proposal (below) for GSA support to further scope and potentially scale NLP for public comment submissions, which will save HHS time, resources, and taxpayer dollars.

**Proposal for10x Funding**: Currently, HHS receives and manually reviews thousands of public comments and unstructured text from rule making and its hundreds of FACA working groups, which cost time, resources, and taxpayer dollars. A more time and cost effective approach would be to automate this review process, by which major themes/topics are distilled and respondent sentiment determined, using AI and NLP techniques. Were a program like this adopted within an HHS pilot program, it could have the potential to scale to other parts of government which face the same resource and time limitations when it comes to reviewing public comments.

**Conclusion**: This HHS CTO pilot is viable, and yielded efficiencies that we recommend be further developed and scoped out for all written public comments to HHS, including FACA working groups and rulemaking. Noted is the increase in negative sentiment after the publication of the TBDWG report to Congress (see Appendix A). Scaling this pilot will require resources, and one potential source is GSA’s 10x funding opportunity.

**Appendix A — Technical Details**:

Input Information

Inputs included:

The initial set of files is split into four types:

1. 2017 to present -- BOTH: All written comments from both tickbornedisease@hhs.gov emails + public meeting written comments (together)
2. 2017 to present – EMAIL ONLY: All written comments from only tickbornedisease@hhs.gov emails (sans public meeting written comments)
3. June 4, 2018, to present -- BOTH: All written comments from both tickbornedisease@hhs.gov + public meeting written comments from June 4 (together – with only our last meeting)
4. June 4, 2018, to present – EMAIL ONLY: All written comments from only tickbornedisease@hhs.gov (sans public meeting written comments)

The Technology

Since the public comments were originally presented in a Word document, the text had to be transformed into a format that could be readable by a Python computer script. The document that was analyzed first was the 2017-to-present email only comments. Then, the first step in this process of topic modelling was to parse the data given the format of every comment within each of the four documents. Each comment functions as a document, in terms of the resulting topic model.

The basic structure of each comment is as follows:

* Date Received
* Response Date
* Subject
* Body

[parser, preprocessing]

For each of these four sections, the parser created a new field for every instance of a comment. The csv file that resulted from this is then updated until all comments are read through. These four sections correspond to four new columns.

The Body section of this table was then isolated, and tokenized for preprocessing. Tokenization in this case consists of breaking down the sentences in the body into individual words. Further processing, or lemmatization, was done to keep only noun, adjective, verb and adverb words. Lemmatization means reducing a group of words to their root, e.g. **better**, **best** 🡪 **good**, after the procedure.

Next, a vocabulary was created, first by removing a set of stopwords, or common words or phrases like “the”, “and” or “www” in the case of comments that included URLs. The data was then vectorized, or turned into numerical format, so that it could be analyzed by the computer code.

[**topic model**, grid search, LDA, top 10 words]

To create the topic model, first a process called grid search was performed on a fitted model of the vectorized data to determine the optimal number of topics that should be generated. The answer that resulted was **four topics**. The model used was latent Dirichlet Allocation, which is a statistical model that allows sets of observations to be explained by unobserved groups which can help explain why some parts of the data are similar. In the case of words being collected in documents, the LDA suggests that each document is a mixture of a small number of topics and that each word’s presence in that document is related to one of the existing topics in the document.

After the four topics were generated, the top 10 words for each of them were produced. This was part of an attempt to then determine what each topic represents. In an otherwise automated process, at this stage, topics were determined in a somewhat informal, holistic fashion.

The four topics are now understood as the following:

1. Patient experience
2. Public engagement
3. Physical manifestation of the disease
4. Doctor-patient relationship

[sentiment analysis]

With topics generated, the next stage was to produce a sentiment analysis of each document, as part of a larger effort to see which words drive the sentiments towards positive, negative or neutral in both the documents themselves and across the four topics. A package specific to measuring sentiment was imported into the Python script, including an analyzer that contains a library of words with different weights for sentiments. An entire sentence or document can be analyzed and the sentiment is determined by adding up the different components. Some words like “bad” or “sad” would have a certain negative weight, whereas “funny” and “bright” will have a certain positive weight, for example.

Two methods were used to determined sentiment. In the first method, a column was created that took the highest score, known as a polarity score, from each document and input it into their respective rows. However, the highest score was almost always neutral—of the four total categories, which were: positive, negative, neutral and compound. Since almost all the cells originally resulted in neutral, those data would not be particularly informative of sentiments and which words drive them. So, a slight adjustment was made to the script to select only the highest values from either the positive and negative sections of the polarity scores. The final output from this script was either a positive or negative sentiment for each row/document, and the corresponding sentiment value.

In the second method, sentiment was determined by isolating the “compound” value, which is the single number that scores the sentiment for a document. The value ranges from -1 (extremely negative) to 1 (extremely positive). To distinguish positive, negative and neutral scores from each other, a threshold with an absolute value of 0.5 was established. Any document with a compound score greater than 0.5 is considered positive, while any document with a compound score less than -0.5 is considered negative and any document with a compound score in between those two values is considered neutral. The dataset was also subset into comments that came in before *and* after the publication of the Tick-Borne Disease Working Group 2018 Report to Congress on November 15. The distribution of positive, negative and neutral sentiments for the entire dataset, and the two subsets, can be seen below:

|  |  |
| --- | --- |
| **Entire Dataset (11/6/17 to 4/24/2019)** | |
| **Sentiment** | **Distribution** |
| Positive | 48% |
| Negative | 29% |
| Neutral | 23% |

|  |  |
| --- | --- |
| **Subset 1 (11/6/17 to 11/15/2018)** | |
| **Sentiment** | **Distribution** |
| Positive | 50% |
| Negative | 27% |
| Neutral | 23% |

|  |  |
| --- | --- |
| **Subset 2 (11/16/18 to 4/24/2019)** | |
| **Sentiment** | **Distribution** |
| Positive | 35% |
| Negative | 42% |
| Neutral | 23% |

Note: The script which assigns the sentiment key based on compound score can be adjusted to a different threshold, e.g. an absolute value of 0.2 or 0.8 instead of 0.5, which has been presented here.