DATA SCIENCE @NLM

OPEN HOUSE

Chart and respond to trends in information seeking by classifying web visitor queries

Dan Wendling (LO-PSD) // *Hackathon team–September 2018*

Semantic Search Log Analysis //

Activity logs for internal search of large web sites are often too verbose and inharmonious to analyze. The site www.nlm.nih.gov has around 100,000 visitor queries per month, with many variations on the same conceptual ideas. Combining log entries such as “ACA” and “Affordable Care Act” and “ObamaCare” across tens of thousands of rows, for example, is far too difficult for a human to do. An informal poll at the 2018 HHS Digital Community Day found that only two HHS web managers out of dozens, were looking for meaning in their search logs.

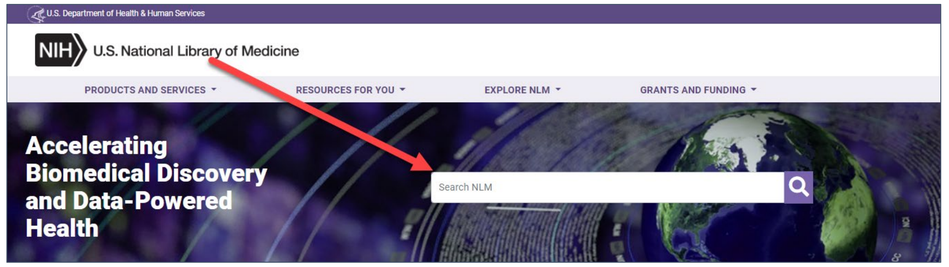
Site search represents the **direct expression of our visitors’ intent.** We could use this data to improve our staff’s awareness of what customers want from us. For example, we could:

1. **Cluster and analyze trends we know about.** For multi-faceted topics that directly relate to our mission, we could create customized analyses using Python to collect the disparate keywords people might search for into a single “bucket.” Where can we create a better match between user interest and our content? Where might we improve our site structure and navigation?
2. **Focus staff work onto new trends, as the trends emerge.** When something new starts to happen that can be matched to our mission statement, we can start new content projects to address the emerging need in HTML pages, social media posts, etc.

Method – UMLS “Bag of Phrases” //

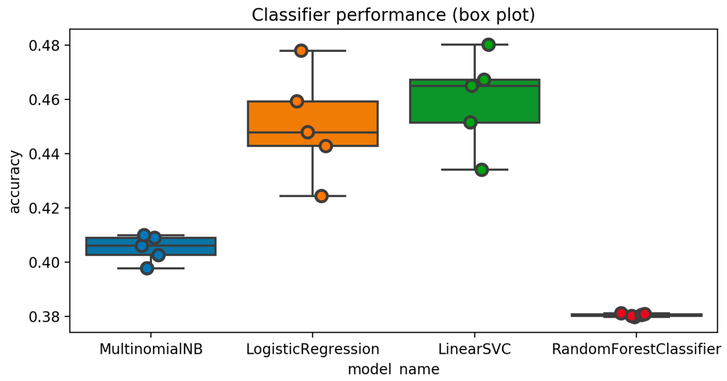
The current study used Python to analyze 6 months of data (12/1/2018 to 5/31/2019), 616,000 rows of data, to improve the training set for supervised machine learning in scikit-learn. Rather than using the “bag of words” method common to many machine learning explanations, this project uses NLM’s **Unified Medical Language System** as a **“bag of phrases”** that allows us to assign semantic meaning, so the search “spinal cord injuries pressure ulcers” will be treated as a three-word type of injury and a two-word medical finding. The partly automated procedure is as follows:

1. Tag all entries longer than 30 characters as Bibliographic entities (usually titles of documents; can be re-analyzed later)
2. Tag foreign-character terms and exact-match them to UMLS foreign terms on local machine
3. Exact match to UMLS-English terms, on local machine
4. Exact match to “gold standard” file of previously matched words, including UMLS and those related to the NLM organization, products and services; to journal names and abbreviations; and other sources
5. Exact match to “quirky matches” such as medical misspellings or other variants; if someone typed “Virtual Human” we assume they meant the NLM product “Visible Human”
6. Fuzzy match (Levenshtein Distance) to the gold standard file, allowing for approximate spellings – “high-confidence guesses”
7. Normalized-string match using the UMLS API, which includes various lexical tools that don’t need to be recreated locally
8. Manual match, through a Python-Django browser interface, of frequent queries not already tagged
9. UMLS term extraction against multi-concept unmatchable queries.



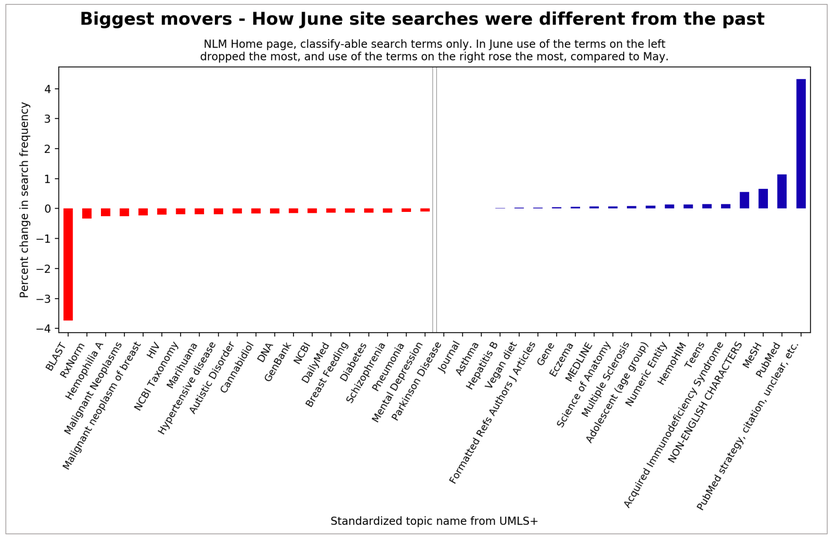
Results //

83% of log rows were tagged, most correctly, but with many errors as well. Running Susan Li’s 4-classifier (untuned) benchmark yielded the results below. In this experiment, term prediction capability in the training set was found to be low, with LinearSVC showing the best result:



Terms are tagged at 3 levels: the UMLS Semantic Network Ontology with around 15 categories, the UMLS Semantic Types with around 130 categories, and many terms have a “preferred” or standardized version.

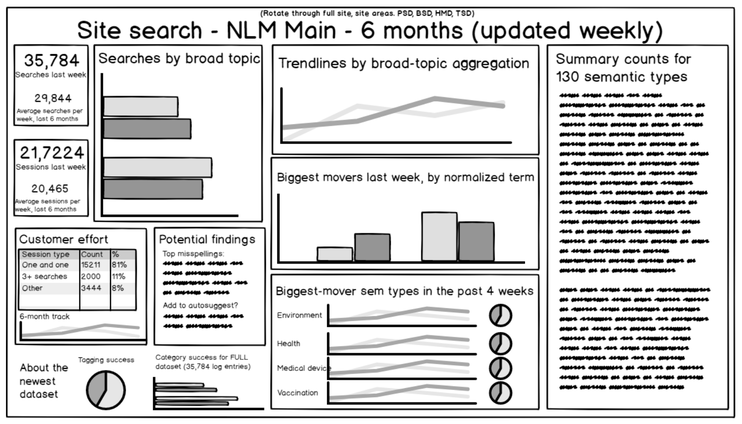
The before-and-after report below from 2018 helped us understand how searching from the home page changed after a home-page redesign, using preferred terms:



Next steps //

While the classifiers could be tuned to improve results, the training set performs poorly in topics with no representation in the UMLS, such as those related to the NLM organization, NLM products and services, staff names, author names, journal names and abbreviations, drug names, etc. Many proper nouns in the search logs are not currently matchable. For successful classification of this NLM web site, better lists of these are required. Existing lists should be obtained, and a tagging interface was created to capture unmatched but frequently searched (i.e., new and trending) queries. We should also test matching to MeSH, a smaller vocabulary than the UMLS. With a better foundation in place, we should explore processing with a deep learning tool such as TensorFlow, Keras, or PyTorch. The project will be submitted for consideration in future hackathon events. The eventual goal is to post a Python code package that HHS web managers can use to classify their own search logs.

A 16:9 (television-sized) visualization envisioned to summarize results:



Influences & Thanks //

- McCray AT, Burgun A, Bodenreider O. (2001). Aggregating UMLS semantic types for reducing conceptual complexity. *Stud Health Technol Inform*. 84(Pt 1):216-20. PMID: 11604736. See also https://semanticnetwork.nlm.nih.gov/
- Lai KH, Topaz M, Goss FR, Zhou L. (2015). Automated misspelling detection and correction in clinical free-text records. *J Biomed Inform*. Jun;55:188-95. doi: 10.1016/j.jbi.2015.04.008. Epub 2015 Apr 24. PMID: 25917057.

- PSD/RWS Management; 2017 HHS Data Science CoLab Bootcamp (HHS-CTO and participants); UMLS staff; OCCS/AB Research & Development; OCCS Desktop Support; Data Society staff; NCBI Hackathon staff and participants; Jessica Chaiken; reviewers; Susan Li’s blog.