Natural Language Processing (NLP): Software Options

SOFTWARE OPTIONS 1
Topic modeling 1
Lemmatizers 1
NER (Name Entity Recognition) 1
POS (Part of Speech tagger), Tokenizer, Sentence Splitter 1
GENERAL NLP TOOLS 2
Apache OpenNLP 2
ClearNLP 2
GATE 2
NLTK 2
Stanford CoreNLP 2
CoreNLP and its multiple NLP tools 3
CoreNLP and the languages it can deal with 3
FURTHER TEXTUAL ANALYSIS TOOLS 3
AntCONC 3
Automap 3
KNIME 4
Linguistic Inquiry and Word Count (LIWC) 4
SEANCE (SENTIMENT ANALYSIS AND COGNITION ENGINE) 7
TACIT (Text Analysis, Collection and Interpretation Tool) 7
Voyant 8
LEXICAL RESOURCES 8
FrameNet 8
WordNet 8

SOFTWARE OPTIONS

Topic modeling

A good **freeware topic modeling** program for the English language is Mallet (http://mallet.cs.umass.edu/), although it is no longer supported. Gensim is a good Python implementation of topic modeling (https://www.machinelearningplus.com/nlp/topic-modeling-gensim-python/).

Lemmatizers

Good **freeware lemmatizers** for the English language (but some can process other languages as well) are the Stanford lemmatizer, TreeTagger, AntCon, LemmaGen.

NER (Name Entity Recognition)

Good **freeware NER** programs are the Stanford NER, Open NLP Apache, GATE, NLTK.

POS (Part of Speech tagger), Tokenizer, Sentence Splitter

Good **freeware Part of Speech tagger, tokenizer, and sentence splitter** are GATE, Stanford

parser.

GENERAL NLP TOOLS

Apache OpenNLP

You can read about Apache OpenNLP at https://opennlp.apache.org/). From the Apache OpenNLP website: “The Apache OpenNLP library is a machine learning based toolkit for the processing of natural language text. It supports the most common NLP tasks, such as tokenization, sentence segmentation, part-of-speech tagging, named entity extraction, chunking, parsing, and coreference resolution. These tasks are usually required to build more advanced text processing services. Apache OpenNLP also includes maximum entropy and perceptron based machine learning.”

ClearNLP

ClearNLP is not directly supported in PC-ACE. You can read about it at https://code.google.com/archive/p/clearnlp/ and download it at https://github.com/clir/clearnlp.

From the ClearNLP website: "This project provides fast and robust NLP components implemented in Java."

It appears that the last updates date back to 2013.

GATE

GATE, an NLP tool developed at the University of Sheffield, UK, can be downloaded at https://gate.ac.uk/ From the GATE website: “GATE is an open source software toolkit capable of solving almost any text processing problem. It has a mature and extensive community of developers, users, educators, students and scientists. It is used by corporations, SMEs, research labs and Universities worldwide. It has a world-class team of language processing developers.

NLTK

NLTK (Natural Language Toolkit) can download it at http://www.nltk.org/install.html. From the NLTK website: “NLTK is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries.

NLTK IS NO LONGER SUPPORTED AND UPDATED.

Stanford CoreNLP

Several different (mostly freeware) tools are available in computational linguistics. But, certainly, the freeware, open source, Stanford CoreNLP, provides the gold standard in NLP.

We read on the Stanford CoreNLP website (http://nlp.stanford.edu/software/corenlp.shtml):

> Stanford CoreNLP provides a set of natural language analysis tools which can take raw text input and give the base forms of words, their parts of speech, whether they are names of companies, people, etc., normalize dates, times, and numeric quantities, and mark up the structure of sentences in terms of phrases and word dependencies, indicate which noun phrases refer to the same entities,

indicate sentiment, etc. Stanford CoreNLP is an integrated framework. Its goal is to make it very easy to apply a bunch of linguistic analysis tools to a piece of text. Starting from plain text, you can run all the tools on it with just two lines of code. It is designed to be highly flexible and extensible. With a single option you can change which tools should be enabled and which should be disabled. Its analyses provide the foundational building blocks for higher-level and domain-specific text understanding applications.

Stanford CoreNLP is written in Java and licensed under the GNU General Public License (v3 or later; in general Stanford NLP code is GPL v2+, but CoreNLP uses several Apache-licensed libraries, and so the composite is v3+). Source is included. Note that this is the full GPL, which allows many free uses, but not its use in proprietary software which is distributed to others. The download is 260 MB and requires Java 1.8+.

The Stanford CoreNLP provides the gold standard among NLP parsers. The Stanford CoreNLP parser uses a state-of-the-art Probabilistic Context free Grammar (PCFG) (Klein and Manning 2003).

Klein, Dan and Christopher D. Manning. 2003. "Accurate Unlexicalized Parsing." In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, pp. 423–430. Association for Computational Linguistics.

To run the Stanford CoreNLP you need to download and install the freeware Java and Stanford CoreNLP and other CoreNLP components. See relevant TIPS file.

```
Invalid maximum heap size: -Xmx4g
The specified size exceeds the maximum representable size.
Error: Could not create the Java Virtual Machine.
Error: A fatal exception has occurred. Program will exit.
```

CoreNLP and its multiple NLP tools

Stanford CoreNLP integrates many of our NLP tools, including the part-of-speech (POS) tagger, the named entity recognizer (NER), the parser, the coreference resolution system, the sentiment analysis, and the bootstrapped pattern learning tools.

CoreNLP and the languages it can deal with

The basic distribution provides model files for the analysis of English, but the engine is compatible with models for other languages. Below you can find packaged models for Chinese and Spanish, and Stanford NLP models for German and Arabic are usable inside CoreNLP.

FURTHER TEXTUAL ANALYSIS TOOLS

AntCONC

AntCONC is a freeware textual analysis software developed by Lawrence Anthony. From the AntCONC website (http://www.laurenceanthony.net/software.html) we read: "A freeware corpus analysis toolkit for concordancing and text analysis."

Automap

From the Automap website (http://www.casos.cs.cmu.edu/projects/automap/) one reads: Extract, Analyze and Represent Relational Data from Texts

AutoMap is a text mining tool developed by CASOS at Carnegie Mellon. Input: one or more unstructured texts.
Output: DyNetML files and CS files.
AutoMap is designed to work seamlessly with ORA.

AutoMap enables the extraction of information from texts using Network Text Analysis methods. AutoMap supports the extraction of several types of data from unstructured documents. The type of information that can be extracted includes: content analytic data (words and frequencies), semantic network data (the network of concepts), meta-network data (the cross classification of concepts into their ontological category such as people, places and things and the connections among these classified concepts), and sentiment data (attitudes, beliefs).
Extraction of each type of data assumes the previously listed type of data has been extracted.

AutoMap exists as part of a text mining suite that includes a series of pre-processors for cleaning the raw texts so that they can be processed and a set of post-processor that employ semantic inferencing to improve the coding and deduce missing information. These pre-processors include such sub-tools as a pdf to txt converter, non-printing character removal, and limited types of deduplication. Text pre-processing condenses data into concepts, which capture the features of the texts relevant to the user. Statement formation rules determine how to link extracted concepts into networks. The postprocessors include such procedures that link to gazetteers and augment the coding with latitude and longitude, belief inference procedures, and secondary data cleaning tools. In addition there are a series of support tools for creating, maintaining, and editing delete lists, generalization thesauri, and meta-network thesauri.

AutoMap uses parts of speech tagging and proximity analysis to do computer-assisted Network Text Analysis (NTA). NTA encodes the links among words in a text and constructs a network of the linked words.

AutoMap subsumes classical Content Analysis by analyzing the existence, frequencies, and covariance of terms and themes.

KNIME

KNIME Analytics Platform can download at https://www.knime.org/downloads/overview. From the KNIME website: “KNIME® Analytics Platform is the leading open solution for data-driven innovation, helping you discover the potential hidden in your data, mine for fresh insights, or predict new futures. Our enterprise-grade, open source platform is fast to deploy, easy to scale and intuitive to learn. With more than 1000 modules, hundreds of ready-to-run examples, a comprehensive range of integrated tools, and the widest choice of advanced algorithms available, KNIME Analytics Platform is the perfect toolbox for any data scientist. Our steady course on unrestricted open source is your passport to a global community of data scientists, their expertise, and their active contributions.”

Linguistic Inquiry and Word Count (LIWC)

Contrary to the other NLP programs described here, LIWC is not free, although a license for 2 computers costs about $100. If you don’t want to pay that price, you can try out LIWC online for free (http://liwc.net/liwcresearch07.php) but, be aware that LIWC will keep a copy of your data.

From the LIWC website (http://liwc.wpengine.com/) one reads:

The way that the Linguistic Inquiry and Word Count (LIWC) program works is fairly simple. Basically, it reads a given text and counts the percentage of words that reflect different emotions, thinking styles, social

concerns, and even parts of speech. Because LIWC was developed by researchers with interests in social, clinical, health, and cognitive psychology, the language categories were created to capture people's social and psychological states.

HOW DOES LIWC ANALYZE LANGUAGE?

The LIWC program includes the main text analysis module along with a group of built-in dictionaries. The text analysis module was created in the Java programming language and runs identically on PC and Mac computers. LIWC reads written or transcribed verbal texts which have been stored in a digital, computer-readable form (such as text files). The text analysis module then then compares each word in the text against a user-defined dictionary. As described below, the dictionary identifies which words are associated with which psychologically-relevant categories.

After the processing module has read and accounted for all words in a given text, it calculates the percentage of total words that match each of the dictionary categories. For example, if LIWC analyzed a single speech that was 2,000 words and compared them to the built-in LIWC2015 dictionary, it might find that there were 150 pronouns and 84 positive emotion words used. It would convert these numbers to percentages, 7.5% pronouns and 4.2% positive emotion words.

Whereas the text analysis module identifies and categorizes words, the heart of the program is a group of dictionaries that tell the text analysis module which words to identify and classify.
LIWC2015 comes with three internal dictionary systems: the LIWC2015 dictionary and the previous dictionaries, LIWC2007 and LIWC2001. The LIWC2015 master dictionary is composed of almost 6,400 words, word stems, and selected emoticons. For each dictionary word, there is a corresponding dictionary entry that defines one or more word categories. For example, the word cried is part of five word categories: Sadness, Negative Emotion, Overall Affect, Verb, and Past Focus. Hence, if the word cried was found in the target text, each of these five subdictionary scale scores would be incremented. As in this example, many of the LIWC2015 categories are arranged hierarchically. All sadness words, by definition, will be categorized as negative emotion and overall affect words.

HOW DOES LIWC KNOW WHICH WORDS TO LOOK FOR IN MEASURING A PSYCHOLOGICAL DIMENSION?

For every LIWC dimension, we have built a separate list or dictionary made up of relevant words. LIWC2015, for example, measures the degree to which texts reveal interests in power, status, and dominance using its Power dictionary. By definition, someone who is concerned with power is more likely to be sizing other people up in terms of their relative status. Such a person will be more likely to use words such as boss, underling, president, Dr., strong, and poor when compared with someone who simply doesn't care about power and status.

The hard part of building a power-related dictionary is in determining what words should be in the dictionary and which ones to omit. When first coming up with a Power dictionary , we relied exclusively on human judges. We started with well-known standard dictionaries and thesauruses and, later, asked our research team to generate every word they could think of related to power, dominance, and status. Once a master list of power-related words were collected, a new group of judges evaluated each word and decided if it truly reflected the overarching concept. Each word would be retained in the original dictionary only if the judges all agreed that it was appropriate.

As with past versions of the dictionary, LIWC2015 took advantage of the judges' ratings in determining the core of each word category. Unlike previous version of LIWC, however, we added several steps to refine the dictionary creation process. Using very large data sets, we tested to see if each of the words of a dictionary were in fact related to each other in a statistically valid way. This allowed us to evaluate if each word in the Power category (and all other categories) was truly related to the other words in a way that we would expect. Finally, we could then use our Power dictionary to identify other common words that we may have missed, or exhibited a more specific relationship with Power as a concept.

HOW DOES LIWC KNOW WHAT CATEGORIES ARE PSYCHOLOGICALLY IMPORTANT?

When first built in the 1990s, we drew from the dominant theories in psychology, business, and medicine as well as common sense. Over the years, the theories changed and we learned that some psychological states were more closely related to language than others. We also listened to LIWC users to get a sense of what behaviors, needs, thinking styles, or other psychological states they felt that language might reflect.

HOW DO WE KNOW IF WORD USE VALIDLY REFLECTS PEOPLE'S PSYCHOLOGICAL STATES?

Let's rephrase that: If a person is using a high rate of anger words, are they really angry? This is a tough question to answer directly. It also points to the importance of hundreds of scientific studies that have been conducted since the early 1990s.

There have indeed been several studies that find that when people report themselves as being angry they use more anger-related words. Analyses of speeches, writings, and conversations show that people rate texts that are high in anger words as expressing higher rates of hostility. But is the speaker really angry? Is it possible that she or he is just pretending to be angry? This is a judgment call, and context matters. For example, if you're analyzing the words of a Wikipedia page on "anger management", the results likely have little to do with how angry the author was at the time of writing. If these questions are important to you, check out the reference section below.

DOES LIWC MAKE MISTAKES IN CATEGORIZING PERSONALITY AND LANGUAGE? JUST HOW PRECISE IS IT?

Don't be fooled. LIWC, like all text analysis tools, is a relatively crude instrument. It makes many errors in identifying and counting individual words, especially words in isolation. Consider the word mad – a word that is counted in the Anger, Negative Emotion, and Overall Affect dictionaries. Usually, mad does reflect anger. Sometimes it expresses joy (he's mad for her) and mental instability (mad as a hatter). Fortunately, this is seldom a problem because LIWC takes advantage of probabilistic models of language use. Yes, in a given sentence, the word mad might be used to express positive emotion. However, if the author is expressing a positive state of affairs, they will generally tend to use relatively high rates of other positive emotion words and few anger words. Small classification errors like this rarely impact the conclusions that can be drawn from the results because they are offset by the way that words are most commonly used by people.

Just as individual words may be misclassified, LIWC also does not understand irony, sarcasm, or metaphor. Again, it is all probabilistic. If someone is being mean spirited in their use of sarcasm, there is a good chance that LIWC will capture hostility in other word choices.

Please be careful in interpreting your LIWC output. The more words that you analyze, the more trustworthy are the results. A text of 10,000 words yields far more reliable results than one of 100 words. Any text with fewer than 50 words should be looked at with a certain degree of skepticism.

HELPFUL REFERENCES

Gottschalk, L.A. (1997). The unobtrusive measurement of psychological states and traits. In Text Analysis for the Social Sciences, (Carl W. Roberts, editor). 117-129.

Hirsh, J.B., & Peterson, J.B. (2009). Personality and language use in self-narratives. Journal of Personality in Research, 43, 524-527.

Krippendorff, K., & Bock, M. A. (2009). The Content Analysis Reader. Sage.

Mehl, M. R. (2006). Quantitative text analysis. Handbook of Multimethod Measurement in Psychology, 141-156.

Pennebaker, J.W. (2011). The Secret Life of Pronouns: What Our Words Say About Us. New York: Bloomsbury (www.secretlifeofpronouns.com)

Tausczik, Y.R., & Pennebaker, J.W. (2010). The psychological meaning of words: LIWC and computerized text analysis methods. Journal of Language and Social Psychology, 29, 24-54.

SEANCE (SENTIMENT ANALYSIS AND COGNITION ENGINE)

SEANCE - SENTIMENT ANALYSIS AND COGNITION ENGINE - is not directly supported in PC-ACE. You can download it at http://www.kristopherkyle.com/seance.html.

From the SEANCE website:

SEANCE is an easy to use tool that includes 254 core indices and 20 component indices based on recent advances in sentiment analysis. In addition to the core indices, SEANCE allows for a number of customized indices including filtering for particular parts of speech and controlling for instances of negation. SEANCE takes plain text files as input (it will process all plain text files in a particular folder) and produces a comma separated values (.csv) spreadsheet that is easily read by any spreadsheet software.

TACIT (Text Analysis, Collection and Interpretation Tool)

From the TACIT website (http://tacit.usc.edu) one reads: "Text analysis has become increasingly important for psychological research, and measuring psychological and demographic properties using computational text analysis is slowly becoming a field norm.

Though several limited-method tools for text analysis are already available (e.g. LIWC), and some have become part of standard statistical packages (e.g., SPSS Text Analytics), a unified, open-source architecture for gathering, managing and analyzing text does not exist.

The Computational Social Science Lab (CSSL) at the University of Southern California introduces TACIT: An Open-Source Text Analysis, Crawling and Interpretation Tool.

TACIT's plugin architecture has three main components:

- Crawling plugins, for automated text collection from online sources (e.g., US Senate and Supreme Court speech transcriptions, Twitter, Reddit)
- Analysis plugins, including LIWC-type word count, topic modeling, sentiment analysis, clustering and classification.
- Corpus management, for applying standard text preprocessing to prepare and store corpora.

TACIT's open-source plugin platform allows the architecture to easily adapt with the rapid developments text analysis.

Citation - Dehghani, M., Johnson, K., M., Garten, J., Balasubramanian, V., Singh, A., Shankar, Y., Rajkumar, A., Parmar, N. J., Hoover, J., Pulickal, L. and Boghrati, R.,
Tacit: An Open-Source Text Analysis, Crawling and Interpretation Tool. Available at SSRN: http://ssrn.com/abstract=2660651.

Voyant

Voyant is available for textual analysis online at: https://voyant-tools.org/

From the Voyant website (https://voyant-tools.org/docs/#!/guide/about) we read: "

> Voyant Tools is a web-based text reading and analysis environment. It is a scholarly project that is designed to facilitate reading and interpretive practices for digital humanities students and scholars as well as for the general public.
>
> **What you can do with Voyant**:
>
> - Use it to learn how computers-assisted analysis works. Check out our examples that show you how to do real academic tasks with Voyant.
> - Use it to study texts that you find on the web or texts that you have carefully edited and have on your computer.
> - Use it to add functionality to your online collections, journals, blogs or web sites so others can see through your texts with analytical tools.
> - Use it to add interactive evidence to your essays that you publish online. Add interactive panels right into your research essays (if they can be published online) so your readers can recapitulate your results.
> - Use it to develop your own tools using our functionality and code.

LEXICAL RESOURCES

Lexical resources are data structures/dictionaries that contain various information about words. They are very well-developed for the English language but less so for other languages. They all rely on manual annotations.

FrameNet

In computational linguistics, FrameNet is a project housed at the International Computer Science Institute in Berkeley, California which produces an electronic resource based on a theory of meaning called frame semantics. FrameNet reveals for example that the sentence "John sold a car to Mary" essentially describes the same basic situation (semantic frame) as "Mary bought a car from John", just from a different perspective. A semantic frame can be thought of as a conceptual structure describing an event, relation, or object and the participants in it. The FrameNet lexical database contains around 1,200 semantic frames, 13,000 lexical units (a pairing of a word with a meaning; polysemous words are represented by several lexical units) and over 190,000 example sentences. FrameNet is largely the creation of Charles J. Fillmore, who developed the theory of frame semantics that the project is based on, and was initially the project leader when the project began in 1997.[1] Collin Baker became the project manager in 2000.[2] The FrameNet project has been influential in both linguistics and natural language processing, where it led to the task of automatic **Semantic Role Labeling**.

To disambiguate, e.g., kill a person (violence) or an argument (communication)

WordNet

WordNet is a lexical database for the English language.[1] It groups English words into sets of synonyms called synsets, provides short definitions and usage examples, and records a number of relations among these synonym sets or their members. WordNet can thus be seen as a combination of dictionary and thesaurus. While it is accessible to human users via a web browser, its primary use is in automatic text

analysis and artificial intelligence applications. The database and software tools have been released under a BSD style license and are freely available for download Lexical the WordNet website. Both the lexicographic data (lexicographer files) and the compiler (called grind) for producing the distributed database are available.