Linguística Computacional I

Armenta Garcia Guadalupe Javier
Profesor: Dr. Ildar Batyrshin

Herramientas y Software en Natural language Processing

1. NLTK:

URL: https://www.aclweb.org/anthology/P04-3031.pdf

Natural Language Toolkit (AKA NLTK) is an open-source software powered with Python NLP. NLTK provides users with a basic set of tools for text-related operations.

NLTK features includes:

- Text classification
- Part-of-speech tagging
- Entity extraction
- Tokenization
- Parsing
- Stemming
- Semantic reasoning

2. Spacy

URL: https://spacy.io/

SpaCy is the next step of the NLTK evolution. NLTK is clumsy and slow when it comes to more complex business applications. In the same time, SpaCy provides users with smoother, faster, and efficient experience.

SpaCy is good at syntactic analysis, which is handy for aspect-based sentiment analysis and conversational user interface optimization. SpaCy is also an excellent choice for named entity recognition. SpaCy can be used for business insights and market research.

Another SpaCy advantage is word vectors usage. Unlike OpenNLP and CoreNLP, SpaCy works with word2vec and doc2vec.

Still, the main advantage of SpaCy over the other NLP tools is its API. Unlike Stanford CoreNLP and Apache OpenNLP, SpaCy got all functions combined at once, so it does not need to select modules on its own. However, Spacy does not support as many languages as NLTK. It does have a simple interface with a simplified interface with a

simplified set of choices and great documentation, as well as multiple neural models for various components of language processing and analysis.
SpaCy is also useful in deep text analytics and sentiment analysis.

3. GenSim

URL: https://www.machinelearningplus.com/nlp/gensim-tutorial/

Sometimes NLP researchers need to extract particular information to discover business insights. GenSim is the perfect tool for such things. It is an open-source NLP library designed for document exploration and topic modeling. It would help to navigate the various databases and documents.
The key GenSim feature is word vectors. It sees the content of the documents as sequences of vectors and clusters. And then, GenSim classifies them.
GenSim is also resource-saving when it comes to dealing with a large amount of data.
The main GenSim use cases are:

- Data analysis
- Semantic search applications
- Text generation applications (chatbot, service customization, text summarization, etc.)

4. TextBlob

URL: https://buildmedia.readthedocs.org/media/pdf/textblob/latest/textblob.pdf

TextBlob is the fastest natural language processing tool. TextBlob is an open-source NLP tool powered with NLTK. It could be enhanced with extra features for more in-depth text analysis.

You can use TextBlob sentiment analysis for customer engagement via conversational interfaces. Besides, it can build a model with the verbal skills of a broker from Wall Street.
Other TextBlob notable feature is a machine translation. Content localization has become trendy and useful. For that, it would be great to have your website/application localized in an automated manner. Using TextBlob, you can optimize the automatic translation using its language text corpora.

TextBlob also provides tools for sentiment analysis, event extraction, and intent analysis features. TextBlob has different flexible models for sentiment analysis. Thus, you can build entire timelines of sentiments and look at things in progress.
TextBlob provides a simple API for for the tasks related to NLP such as Parts of speech tagging, sentiment analysis, machine translation.

It features are:

- Noun phrase extraction
- Part-of-speech tagging
- Sentiment analysis
- Classification (Naive Bayes, Decision Tree)
- Tokenization (splitting text into words and sentences)
- Word and phrase frequencies

- Parsing
- n-grams
- Word inflection (pluralization and singularization) and lemmatization
- Spelling correction
- Add new models or languages through extensions
- WordNet integration

5. Intel NLP Architect:

URL: https://arxiv.org/pdf/1808.08953.pdf

Intel NLP Architect is the newer application in this list. Intel NLP Architect uses Python library for deep learning using recurrent neural networks. You can use it for:

- text generation and summarization
- aspect-based sentiment analysis
- and conversational interfaces such as chatbots

One of its most exciting features is Machine Reading Coprehension. NLP Architect applies a multi-layered approach by using many permutations and generated text transfigurations. In other words, it makes the output capable of adapting the style and presentation to the appropriate text state based on the input data.

The other great feature of Architect NLP is Term Set Expansion. This set of NLP tools fills in the gap of data based on its semantic features. An example can be described: When making research on virtual assistants, initially the input would be “Siri” or “Cortana.” Term Set Expansion (TSE) adds the other relevant options as “Amazon Echo.” In more complex cases, TSE is capable of scraping bits and pieces of information based on longer queries.

6. Textacy

URL: https://buildmedia.readthedocs.org/media/pdf/textacy/stable/textacy.pdf

Textacy uses Spacy for its core NLP functionality, but it handles a lot of the work before and after the processing. Textacy can easily bring many types of data without having to write extra helper code.

7. PyTorch NLP:

URL: https://arxiv.org/pdf/1812.08729.pdf}

PyTorch NLP is a great tool for rapid prototyping. It has pre-trained embeddings, samplers, dataset loaders, metrices, neural network modules, and text encoders.

8. Stanza

URL: https://arxiv.org/pdf/2003.07082.pdf

Stanza is a Python natural language analysis package. It contains tools, which can be used in a pipeline, to convert a string containing human language text into lists of sentences and words, to generate base forms of those words, their parts of speech and morphological features, to give a syntactic structure dependency parse, and to recognize named entities. The toolkit is designed to be parallel among more than 70 languages, using the Universal dependencies formalism.

Stanza is built with highly accurate neural network components that also enable efficient training and evaluation with your own annotated data. The modules are built on top of thePytorch library. You will get much faster performance if you run this system on a GPU-enabled machine.

In addition, Stanza includes a Python interface to the CoreNLP java package and inherits additonal functionality from there, such as constituency parsing, coreference resolution, and linguistic pattern matching.

To summarize, Stanza features:

- Native Python implementation requiring minimal efforts to set up
- Full neural network pipeline for robust text analytics, including tokenization, multi-word token (MWT) expansion, lemmatization, part-of-speech (POS) and morphological features tagging, dependency parsing, and named entity recognition
- Pretrained neural models supporting 66 human languages
- A stable, officially maintained Python interface to CoreNLP

Bibliography

Bird, S., Loper, E., (2004). NLTK: The Natural Language Toolkit. In *Proceedings of the ACL Interactive poster and Demonstration Sessions* (pp. 214-217)

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S., & McClosky, D. (2014, June). The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations* (pp. 55-60).

Rehurek, R., & Sojka, P. (2011). Gensim–python framework for vector space modelling. *NLP Centre, Faculty of Informatics, Masaryk University, Brno, Czech Republic*, *3*(2)..

DeWilde, B. (2017). textacy Documentation.

Aly, A., Lakhotia, K., Zhao, S., Mohit, M., Oguz, B., Arora, A., & Shah, R. (2018). Pytext: A seamless path from nlp research to production. *arXiv preprint arXiv:1812.08729*.

Khashabi, D., Sammons, M., Zhou, B., Redman, T., Christodoulopoulos, C., Srikumar, V., ... & Tsai, C. T. (2018, May). Cogcompnlp: Your swiss army knife for nlp. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Qi, P., Zhang, Y., Zhang, Y., Bolton, J., & Manning, C. D. (2020). Stanza: A Python Natural Language Processing Toolkit for Many Human Languages. *arXiv preprint arXiv:2003.07082*