Investigation into Natural Language toolkits

Taylor Brown (tab23@aber.ac.uk)

February 12, 2019

1 Introduction

This report looks at three different toolkits designed for Natural Language Processing (NLP) in Python. These toolkits are:

1. Natural Language ToolKit (NLTK)
2. SpaCy
3. Polyglot

These three were chosen for various reasons. NLTK is the toolkit that has been around the longest: it was first produced back in 2001, and is the toolkit used by academics for its extensive documentation and sheer power. SpaCy is much newer to the NLP scene, and is self confessed to be designed for developers over research[1]. SpaCy offers support in several languages, including German, Italian and Greek as well as English, and offers a very basic tutorial for those new to both spaCy and NLP. Finally, Polyglot is the newest toolkit of the three, and was designed for international use. It was built to support massive multilinugal applications, and boasts support for over a hundred languages in various areas of NLP.

This report will compare not only the ease of use and accuracy of the toolkits, but also takes into account the potential needs of the system in mind beyond the current project duration. These factors will be used to decide which toolkit should be used for information extraction in this system. The ease of use test for the toolkit is simple: implement named entity recognition. Since this task in a key feature of the information extraction section of the system, this was recognized to be the best way of determining which toolkit would be best suited to the system.

Each toolkit was tested with the same piece of text, 1, that was taken directly from an article in the archives that are used in the main system. This helps to determine how each implementation will respond to the actual data.

Figure 1: Testing text

A CURIOUS EXPERIMENT. A distinguished German biologist—Dr. Weisman —is making experiments in the way of trying to show that artificial modifications made in living animals may be reproduced in succeeding genera- tions. He has taken 900 white mice, and cut off their tails with a carving knife, or some other in- strument, and he hopes in time to produce from these mice that will be born tailless. This is not under. taken because a breed of tailless white mice is urgently needed, but to establish a great fact, if it be a fact, in evolution. Whatever success Dr. Weisman may attain, says a correspondent, his attempt iB much more on scientific lines than the theory recently set by an amateur naturalist, with much gravity and alleged circumstance, that the Manx or tailless cat is the product of a chance cross between the ordinary domestic tabby and the wild rabbit. As the Manx cat is a perfect cat in every- thing but its tail, showing nothing of the structure or habits of the rabbit, and as the pairing of a long tailed animal with a short tailed animal would not be likely to abolish the tail altogether: as the rabbit is entirely herbivorous and the cat almost entirely carnivorous, and as the cat would be muchm̂ore likely to eat the rabbit than to pair with it, the amateur naturalist can hardly be said to have brought to light a great scientific truth. What Dr. W eisman will do with his mice remains to be seen.

2 Natural Language ToolKit

This toolkit was complicated to get started with: because the toolkit is so big, many modules have to be imported for a series of different, but very related tasts. Though the toolkit seems complicated - largely attributed to its size - the actual implementation is fairly easy and intuitive. Break down sentences into words -¿ tag words with their associated part of speech -¿ chunk into entities.

This toolkit took 12.01 seconds to run, and accurately depicted all of the named entities in the text, though the results printed were hard to interpret compared to the other toolkits. It recognised the name of the scientist discussed, Dr Weisman, though only recorded it as a named entity once. It did not detect the Manx cat as any kind of named entity, and it recognised German as a geo-political entity, the label used for nationality amongst other things. The headline was incorrectly labeled as an organization, but this could be as due to the capitalization as to an issue with the chunker and, indeed, when the headline is changed to normal capitalization, the label vanishes.

3 SpaCy

The installation of spaCy was found to be much harder than NLTK, in that where spaCy is a module that requires Visual Studio where NLTK did not. It is very simple for beginners to use; an in depth tutorial is available in the same place as its installation guide. This also includes a detailed explaination of how techniques such as tokenization, part-of-speech tagging and named entity recognition can be implemented. Coincidentally, these are the aspects required for the comparision, and what would be used in the information extraction software, making this tutorial the most helpful of the three.

This toolkit took 30.36 seconds to run on the testing text, and did not miss a named entity in the above text. Several words were recognised as being a named entity incorrectly, such as the Manx species of cat, or the word 'much', which is just a mistyped word and is not tagged when spelt correctly. As the final software will spellcheck words, this should not be an issue. Nevertheless, the accuracy is less than desirable, and the code takes far longer to finish than would be prefereable.

4 Polyglot

An exception must be made to the testing criteria for polyglot: it was only considered against the other two toolkits because of the potential benefits of working in both Welsh and English for a collection that is designed to highlight the best and brightest of science in Wales. Unfortunately, though polyglot supports 16 different languages in its part-of-speech tagger, and 40 for its named entity chunker, it does not support Welsh in either of these. This was enough to prevent polyglot from being the toolkit of choice. For good measure, it was still tested in the same way as the other two, and found to be lacking in both power and accuracy, though it is fairly easy to use. It appears that the strength of this toolkit really does lie in its support for multiple languages perhaps even at once, as it appears to offer transliteration. Perhaps, if it expands services to include less common languages like Welsh, it would have a place in this project.

This toolkit was significantly faster than the other two, completing its run in just 4.49 seconds. It did, however, only recognise entities that are people, incorrectly identifying the '-' that occurs before the first 'Dr Weisman', and not recognising 'German' at all. This suggests that accuracy has been sacrificed for speed in this toolkit, and were it to be used for this project, a classifier would have to be built from the ground up.

5 Conclusion

The final choice of toolkit for the information extraction section of the system is NLTK. Though both polyglot and spaCy have their strengths, the former is simply too inaccurate for a system in which a certain level of imprecision must already be taken into account. SpaCy, meanwhile, could be considered an ideal if it did not take significantly longer to produce the same output as the NLTK. Anothe factor that led to the NLTK being the right choice for this system is its design. It would be fairly easy to build and train a classifier using the NLTK should the system later require one. The only exception would be if polyglot were to offer part of speech tagging and named entity recognition support in Welsh, or a good chunker were built by a native Welsh speaker, which would be worth the effort of improving the accuracy of the English classifier.

References

[1] https://spacy.io/usage/spacy-101 What spaCy isn't: 2019-02-09