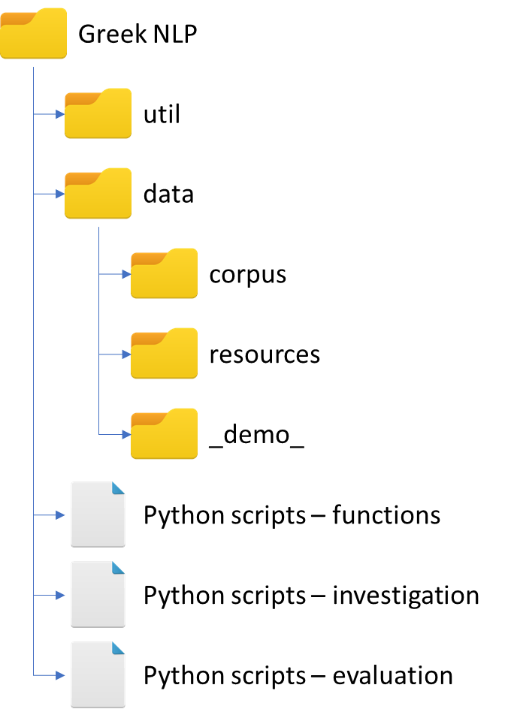
# Greek NLP – Read Me

The resources for Natural Language Processing of Modern Greek Corpora is provided in this *NLP Greek* repository and comprises of two sub-folders, *data* and *util*, along with a series of Python scripts.

Please note that for maintaining the small size of this repository it has been necessary to zip the corpora and remove the resources developed with this code. The resources can be easily replicated using the supplied corpora and code.



**util**

The *util* folder contains the scripts used to covert the semi-structured *INTERA* and *Greek Universal Dependencies* corpuses into the standardised NLTK tagged corpus format.

**data**

**corpus**

The results of the *util* corpus processes are stored in the *corpus* sub-folder under *data*. The text files making up the corpus can be found in their respective folders: *INTERA*, *UDGreek* and *tagged\_texts*, a manually created corpus.

**resources**

**The *resources* sub-folder contains both the initial linguistic resources which are used in the rules-based tagger (*Open\_Word\_Patterns.xlxs*) along with the resultant tagger resources: *Greek\_POS\_seq.pkl* (sequence tagger); *Greek\_POS\_clas.pkl* (classification tagger); and *Greek\_POS\_DL.h5* (deep learning tagger) and its related files. Note that the deep learning tagger is ver**y large and has been zipped, yet is still over 100MB in size. The taggers developed on the finer tagset are also provided in this directory.

**\_demo\_**

**The *\_demo\_* sub-directory contains sample data for the benefit of the inline examples provided in the *greek\_nlp\_pos* library.**

**Python scripts – functions**

**Two scripts are provided containing functions for NLP of Greek. The first, *greek\_nlp\_corpus*, is a specific set of functions that read in and initialise the various corpus while the second,** ***greek\_nlp\_pos*, provides the tools to develop, execute and evaluate the taggers and includes inline help and examples.**

**Additionally, a third script is provided (but not documented here), *tag\_util*, which contains a series of functions by Jacob Perkins and is provided with his *Python 3 Text Processing with NLTK 3 Cookbook*.**

**Python scripts – investigation**

**The investigation scripts are used to investigate and develop the three main taggers: *investigate\_tagger\_nltk\_sequential*; *investigate\_tagger\_nltk\_classifcaction*; and *investigate\_tagger\_nltk\_deeplearning*; as well as the taggers for the finer tagset: *investigate\_tagger\_finetags*.**

**Python scripts – evaluation**

**The one evaluation script, *evaluate\_taggers*, is used to evaluate each of the developed taggers against the test corpora.**