File Handling in the NLP Suite

Table of contents

File handling in the Suite by file name and file content

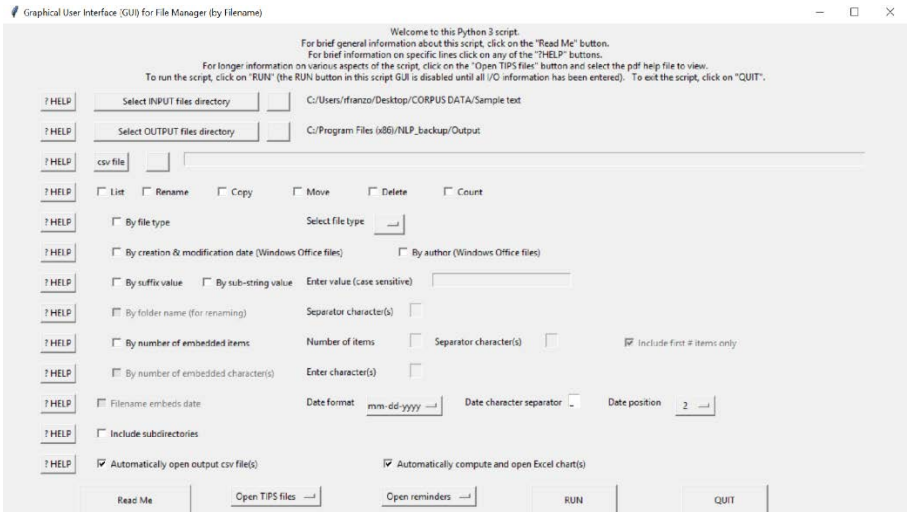
The NLP Suite provides several tools for handling files. These deal with both filenames and file content.

File name

The NLP Suite allows a user to handle files by their filenames and with specialized GUIs and TIPS files. Users can:

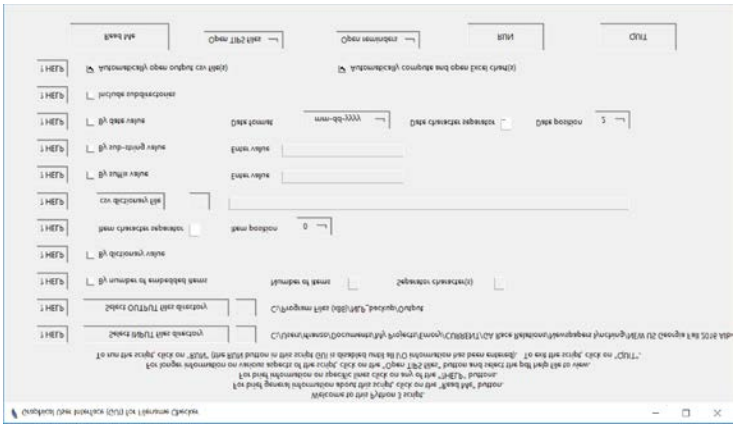
List, Rename, Copy, Move, Delete, Count files by filename properties. Use filename properties (e.g., extension type, embedded dates, number of embedded items) to filter files so as to **List, Rename, Copy, Move, Delete, Count files**.

TIPS_NLP_File manager.pdf



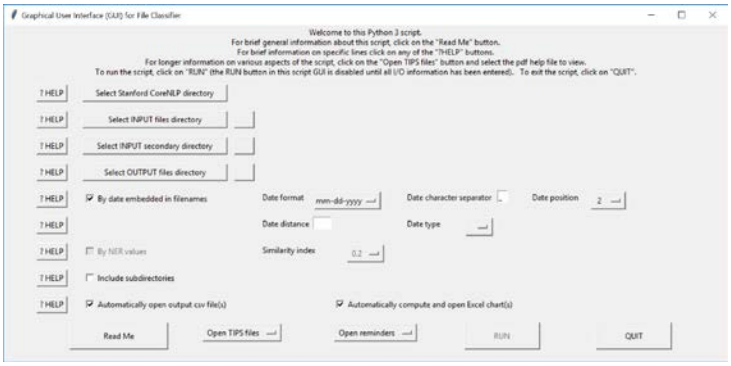
***Filename well-formedness*.** Check the **well-formedness of filenames** with embedded metadata

TIPS_NLP_Filename checker.pdf



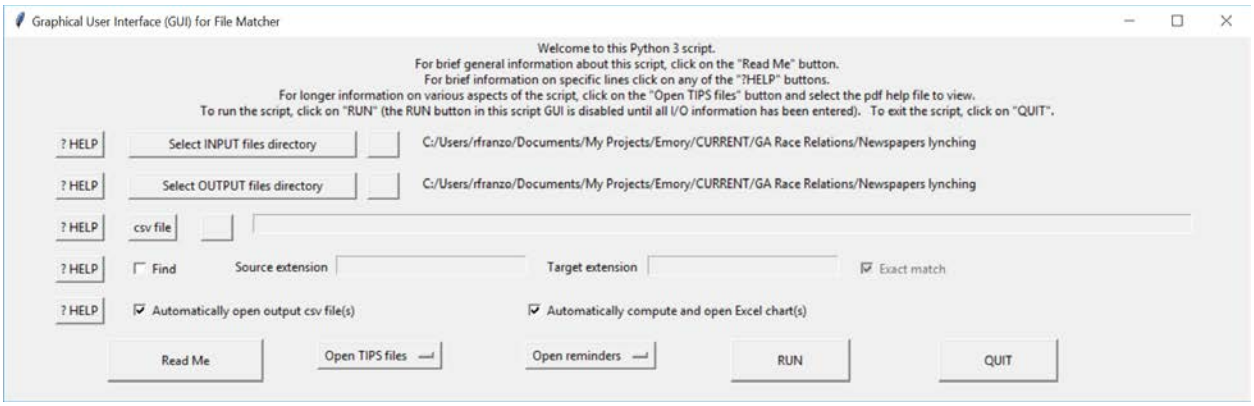
Dumb classifier by date embedded in filename. Use the date embedded in a filename to **classify documents by date range**

TIPS_NLP_File classifier (By date).pdf



Filename matcher. Search directories to **find files with same filename and different extensions**

TIPS_NLP_Filename matcher.pdf



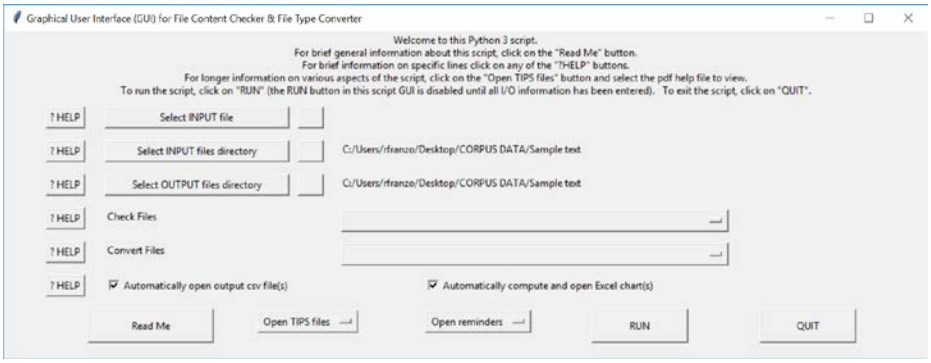
File content

A number of tools deal with file content rather than file name. Most tools come with specialized GUIs, so that they can be run independently. Others must be run from the main NLP GUI (via NLP_main.py).

These tools allow users to:

Convert & check file content

TIPS_NLP_File checker & converter.pdf

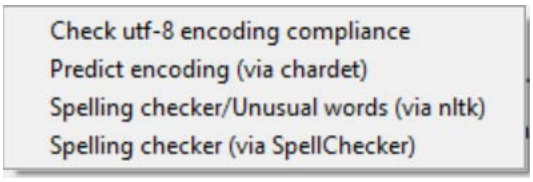


The **converter tools** convert the file content from pdf or docx to txt so that the files can be analysed via NLP tools. Txt type files are the only ones that these tools handle.

pdf to txt converter. The NLP Suite relies on the Python package pdfminer to convert pdf files into txt files (https://pypi.org/project/pdfminer/). Pdfminer can convert 2-column texts, such, for instance, as journal articles.

docx to txt converter. The NLP Suite relies on the Python package Python-docx (https://python-docx.readthedocs.io/en/latest/) to convert docx (not doc!) files into txt files (https://www.geeksforgeeks.org/python-working-with-docx-module/).

The **checker tools** check for **utf-8 encoding** of a txt file, for the **predicted encoding** of a txt file, for the use of **unusual words** in a text (including spelling), and **spelling checker** (via the package SpellChecker).



utf-8 encoding checker. All text files are encoded in specific ways that computers across different platforms and languages can understand. utf-8 is the most widely used type of encoding and compatible with most human languages. All NLP Suite scripts read, process, and save data in utf-8 encoding standard. In particular, the Stanford CoreNLP algorithms which form the basis of many of the **NLP Suite algorithms will break with non utf-8 text data**.

The tool checks a file for utf-8 compliance. It does not have a separate GUI and is run from the NLP_main.py file.

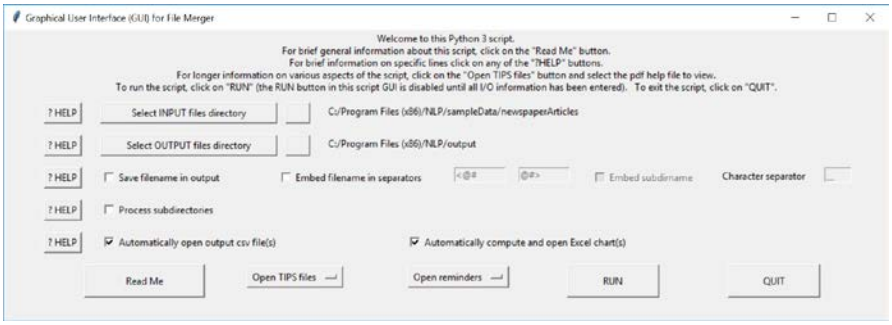
TIPS_NLP_Text encoding (utf-8).pdf

Spelling checker. The NLP Suite provides several tools for checking the spelling of your corpus. The tools do not have separate GUIs and are run from the NLP_main.py file.

TIPS_NLP_Spelling checker.pdf

File merger. The merge script allows users to merge several txt files into a single txt file and with the output option f including/excluding the input filename in the output

TIPS_NLP_File merger.pdf



File splitter. The file splitter functions provide several ways of splitting txt files into subfiles,

using a **previously merged file,** using a **Table of Contents (TOC)**, by **maximum number of words**, by **single words or collocations**, or by **special strings.**

TIPS_NLP_File splitter.pdf