NILE Tutorial

I. About NILE

Narrative Information Linear Extraction (NILE) is a software package that can be used to identify terms and named entities from texts such as clinical notes, articles, or other medical texts. This tutorial covers the setup and use of the Windows executable version of NILE, distributed by CELEHS at Harvard Medical School here.

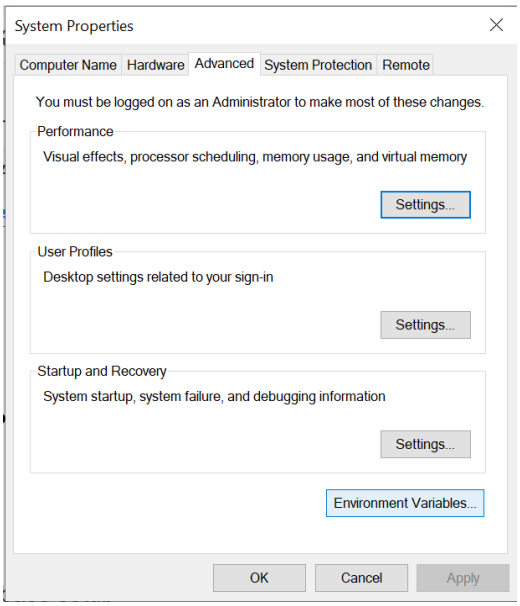
Prerequisites

The desktop version of NILE requires 8GB or more of memory and benefits from a multiple-core CPU if a large dictionary or large text corpus is to be processed. It also requires Java, which can be downloaded online here.

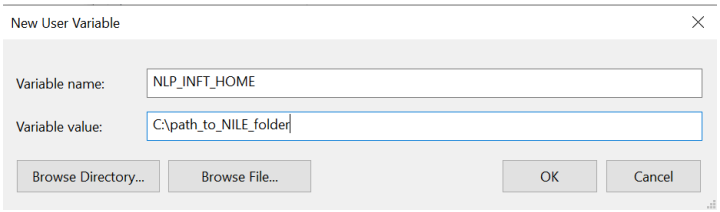
II. Setup and Installation

NILE setup

NILE is distributed in a zipped folder; to begin using the software, extract the contents of the file and set the environment variable.
Use the Start menu to navigate to **System Properties** or **System Settings**. You may also be able to search for “Environment variables” directly using the Start menu.



Navigate to **Environment Variables**, select **New**, and enter the value **NLP_INFT_HOME** as the Variable Name and the local path to the main NILE folder (named **NLP_Interface**) as the Variable Value.



Database setup

NILE can support text data stored in MSSQL or MySQL databases. Some formatting is required; it may be easiest to create a separate view or table with all the required parameters. To see an example table, you can download the sample SQL file from the NILE download page.

The view or table should contain the following columns:

Patient Identifiers: these should be unique numeric identifiers for each patient. At processing time, NILE will proceed sequentially through the values in this column, so it can simplify results processing to create a new variable starting at 1 rather than using an existing patient identifying variable

Document Identifiers: these should be unique identifiers for each document. They need not be sequential or numeric

Dates: date and/or time records. In the results, time components will be truncated

Document Types: this column must exist but does not need to be used. It can indicate the originating department, the type of report, or any other useful variables

Text: the text to be processed, in UTF-8 encoding. Text need not (and possibly should not) be preprocessed to remove punctuation, capitalization, or other features - this can impact the ability of the software to detect negation or stopwords.

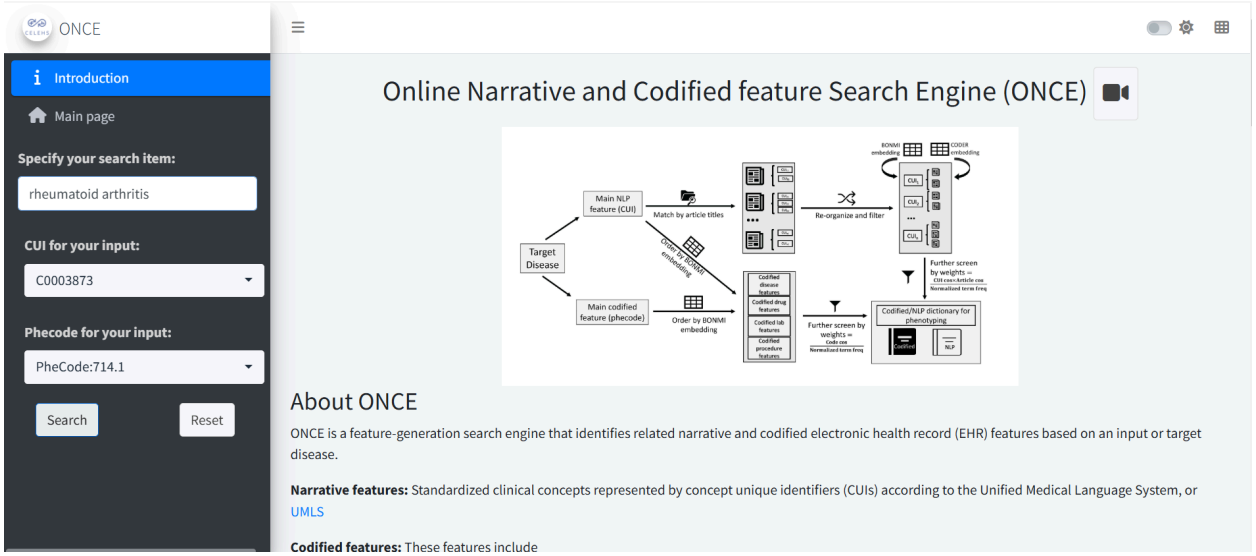
Additional columns are allowed but will not be preserved in the results.

Dictionary creation and curation

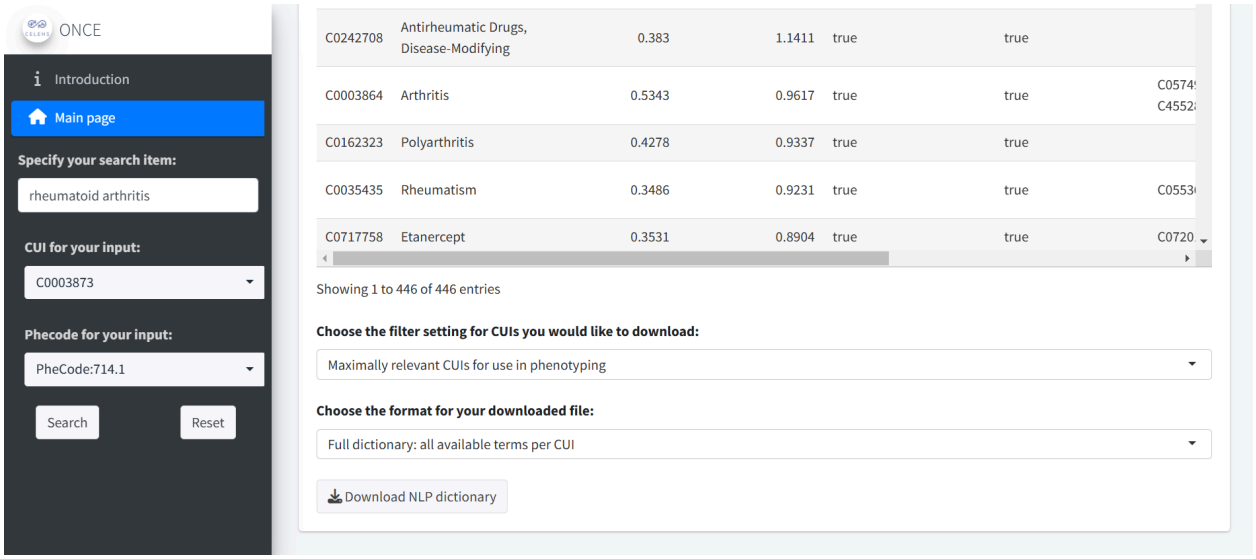
An important component of the NILE project setup is the dictionary: the set of terms the software will search for in the text data, paired with a set of codes used to identify them. The Unified Medical Language System has many resources for finding medical concepts (including many ways they may be spelled, phrased, or abbreviated) and may be useful in creating a dictionary. Also provided by CELEHS is the ONCE app, which generates a list of terms (NLP features) that are related to a searched condition.

ONCE demo

These steps outline the process for downloading a NILE-ready dictionary for the condition rheumatoid arthritis.



At the search bar on the top left of the ONCE home page, enter **rheumatoid arthritis** and press **Search**. The results may take several seconds to populate.



Scroll to the bottom of the results table and select **Full dictionary: all available terms per CUI** from the dropdown menu and select **Download NLP dictionary**.
For more information about ONCE, please see the video tutorial and description on the Information page.

> Regardless of the source of the dictionary, a specific format is required: the dictionary should be a two-column file with columns separated by the pipe (|) character. The first column should contain the string term to be identified in the text, and the second column should contain the corresponding code. The dictionary should not contain a header or column names, and the file should be named **<Your_project_name>_dict.txt**. For

example, a dictionary for a project called “RA_test” would be named **RA_test_dict.txt**. The dictionary should be placed in the **input** folder of the project, in the same directory as the **NLPproperties.txt** file.

Project configuration

To start a new NILE project, create a new folder under **NLP_Interface** > **project** by copying the existing folder and providing the desired project name.
In the file **location.txt** in the **project** folder, enter the name of your new project folder exactly as it appears, with no extraneous white space or punctuation. At processing time, this serves as a parameter that instructs NILE to process the instructions in that project only.
Inside the newly created project folder, open the **input** directory and the text file **NLPproperties.txt** in the text editor of your choice.

III. Properties setup

The **NLPproperties.txt** file contains many important parameters for your NILE project. The file contains some comments about the parameters, and additional notes can be found here:

`server` is the name of the MSSQL server engine
`dbname` is the database name

`user` and `password` may be required depending on authentication requirements for the database server

`tablename` is the table containing the text to be processed

The next 5 arguments correspond to the required columns described in Database setup
`patientIDcol` column containing patient identifiers
`docIDcol` column containing document identifiers
`datecolumn` column containing dates
`notecolumn` column containing the text/note
`notetype` column containing document type

`patstart`, `patincrement`, and `patend` indicate: the patient ID of the first patient to be processed, the number of patients to be included in each batched output file, and the patient ID of the last patient to be processed

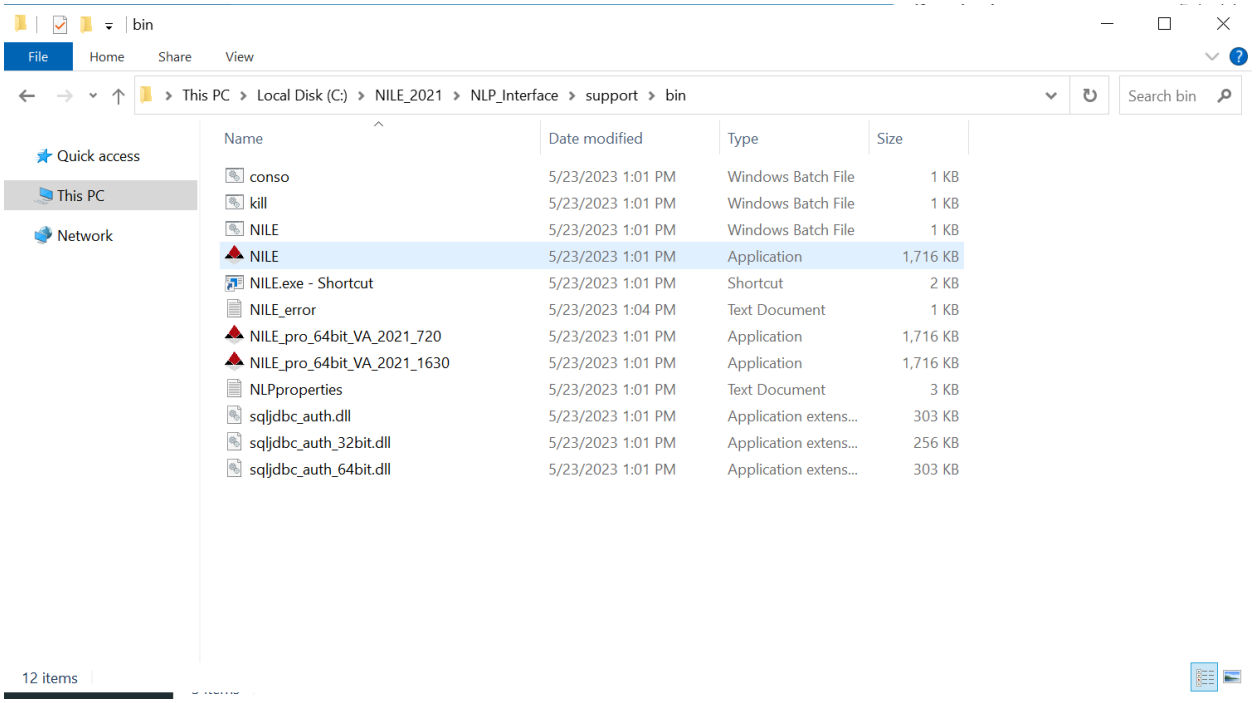
`notelengthlimit` indicates that texts with fewer than this number of characters **will not be processed**. In testing, 500 is a good heuristic for clinical notes, but another threshold may be better for other texts.

`outputfolder` this argument should contain the **fully specified path** (i.e. starting with C:\ or the drive name) to the **output** folder under your NILE project.

After completing all necessary fields, save the file.

IV. Running NILE

To run NILE, navigate to **NLP_Inferface** > **support** > **bin** and run the executable file **NILE.exe** by double clicking.



You may wish to confirm NILE has started by opening the Task Manager. Note that if the patient sample or dictionary is small, NILE may not run for long enough to appear in the Task Manager. If so, you can confirm NILE has run in the **log** or **output** files.

V. Reading logs

Access the main log for a NILE job at **project** > **<your project name>** > **log**. For a detailed log with per-note information and statistics, access **project** > **<your project name** > **output** and select the file ending in "*_log*" for your desired patient list.

VI. Output and postprocessing

NILE output contains no column headers, but the columns (as defined in the NLPproperties file) are:

patientIDcol, docIDcol, datecolumn, CUIs
Where CUIs is a comma-separated list of CUIs each appended with a certainty tag of Y, N, or U. The output may be processed using R, Python, or a similar software to reformat for use in phenotyping or other downstream tasks.