Dominique Danco
LING 550
Final Project

PyLing

~A French-English Vocabulary Learning Application~

Motivation and existing products

The goal of this application is to create an easy-to-use vocabulary-learning application for English-speaking users who are learning French. Many products already exist for this or similar tasks. Related software includes:

Duolinguo:	programs and tools for learning different aspects of foreign languages
Quizlet:	online flashcard website
Rosetta Stone:	software for learning grammar and vocabulary
Memrise:	uses memorization tricks and mnemonics to better retain information
Knewton:	adaptive learning platform

While these applications have many useful features, I believe that there are some shortcomings in each of them. I would like to combine the features I have found to be the most useful, as well as add in some of my own.

Implementation

The application runs on PyQt5 using Python3. Upon starting, the user is asked what words they would like to learn.

Input options

Most vocab apps provide pre-made categories, and many allow users to create their own flashcards by putting in terms and definitions. Instead of requiring the users to provide the translations themselves, PyLing offers either pre-made categories or the option to translate the user's words for them. On one hand, this opens up the app to more chances for translation error, but it also means that the user can use the app as a way of learning the translations, and not be required to already have that knowledge.

Creation of the dictionary

For each word, first a dictionary made from a English-French dictionary text file is searched to find any translations. Then, the French lexical dictionary (Lexique) is searched to see if the word is an alternate form of any other word (for example, the conjugation of a verb), which will then also be checked in the English-French dictionary. The resulting product is a dictionary with all possible translations of the words that the user has chosen to be tested on.

The test

After choosing if they want to be prompted in English or French, the user is presented with the test words one at a time. They then are asked to translate that word (via a QLineEdit widget in PyQt). If they are correct, the test moves on to the next word in the dictionary, and if they are incorrect, they are shown the possible translations of the word. Here, the user has the option to reject the application's translation and remove the word from the dictionary if they believe the app's translation is incorrect.

Repetition until mastery

For each word, a record of tries and correct attempts is kept. For the word to be considered "mastered", the user must have gotten it correctly more than 50% of the time. Therefore, if the user gets it right the first try, it is immediately removed, but if they get it wrong twice, they'll then have to get it right three times before they are finished with that word.

Accent marks:

Different computers have different ways to insert accents, not all of which are obvious to the user. To avoid keyboard confusion, I chose to include buttons to add accent marks. Quizlet does something similar—however, Quizlet has chosen which accents are most likely, then placed buttons for those accents under the input bar. Instead of doing this, I made one "add accent" button, and then made a pop-up window with all possible French accent marks, to ensure that none are missed, and to avoid cluttering the main window.

Completing the test

Once all words have been removed from the testing dictionary, the user has completed the test. A performance report is then displayed, with the amount of times that each word was answered incorrectly. The user is also suggested another related list to learn—if any of the words that they asked to be tested on were in the pre-made lists, that list is deemed to be "similar". The user has the option to either study a similar list, restart with the same words, restart with new words, or exit the application.

Future directions

There are several additions that could be made to PyLing. First, the application could be adjusted to work with other languages than English and French. Another improvement that could be made is faster processing. Going through Lexique takes quite a long time, and this could perhaps be improved with faster methods of data structure construction. Another issue is that when the user selects to be quizzed in English, they are prompted with every possible English definition of the French word, which sometimes can be an overwhelming number of word options. It would be beneficial to find a way of classifying the possible translations so that only the most likely translations were presented. Lastly, I would like to make the application more adaptive—hopefully in future versions the app will be able to learn from the user and present new material in an intelligent and personalized way.

References and acknowledgements

-Lexique:
http://www.lexique.org/

-English-French dictionary:
http://www.slideshare.net/RalphPilkington/english-french-dictionary

-Thank you to Michael Mcauliffe for his lifesaving last-minute PyQt lessons, and much appreciated help.