Corpus Datacraft and Statistical Manipulation

Omar Omokhodion

**introduction**

Humanity is in dire need of polyglots. The capability to transcend rigid borders of the mind presents our species with mutual opportunity and incites collaborative efforts in the arts and sciences. The primary repository of knowledge resides in script and the system of language built along side a written form. With the matter at heart, analytic techniques that provide modernized solutions to the problem of illiteracy should be made more prolific. The topic of this paper details a bundle of useful procedures that can be deployed together as a tool in linguistic analysis towards the application of learning or memorization with discrete information.

**description**

The purpose of this tool is to serve as a semi-autonomous method of analysis for written material. There are two functions, the first being a process that outputs a series of the most relevant and important bigram concepts. The second is flashcard style language task to help memorize the constituents. The current version of the project is available on github and is anticipated to receive future updates, yet all procedures are given in plain text without current programmatic interface. The program follows a pipeline of processes that the user will coordinate. This can be initiated under the litter.exe application.



**Figure 1.1** Regular Expressions of National Scripts in the Occident



**Figure 1.2** Regular Expressions of National Scripts in the Orient

A preliminary schematic of common writing styles and their frequent characters is drawn out in Figure 1.1. Each node is a modern national script whose linguistic influence is shown and grouped by arrows and borders. One can see that Greek leads to Latin and in turn ends up in English through French as a very rudimentary observation in language relation. Each of the systems also has a set of commonly recognized lettering or radicals. The Figure 1.2 shows ideographic oriental scripts and the vast array of relevant terminal symbols that may be found when analyzing Chinese, Korean, or Japanese corpus sources. The green group of Figure 1.2 has the right-left scripts of Arabic and Hebrew. With the last group, the phonetically alphabetic scripts which frequently feature spaced segmentation are given and offer the greatest amount of graphic morphology. Under these umbrellas, estimations can be made on how best to present the information within a corpus. The major distinctions are space segmentation and direction. However, continental European graphs, though commonly utilizing the standard a-z range of the American Standard Code for Information Interchange, make frequent use of diacritic vowels that occupy higher encodings of the eight bit listings. Unicode does use special backwards looking characters that inflect and join with others to bring about this type of spelling. These forms can be verified by knowing the regular digitized script and orthography of a language. With the assumption that the band of encoding characters represents the desired script, the program will externalize any word that contains characters using the exclusion group, and replace their count as '\_EXTERNAL\_GRAPH\_'. Space segmented scripts may be removed of clitic punctuation that appear within a word such as "her's" as an option. This program does so outright, and includes the removal of commas.

Ideographic systems can be segmented at iteration of each character yet there are still national language segments within the ideographic group. These capturing patterns are done through regular expression of text as an extremely efficient manipulation of strings in a digitized encoding.



**Figure 1.3** Procedural Schema

Figure 1.3 expresses the general pipeline of count and calculation. The program then continues to contextualize the words of the corpus data and count the results. The default setting places the context at a range of one to the right of the node. Contextualization includes the isolation of sentences to comprehend an ending node which lacks a right hand context. This task is made more difficult in scripts that use an assembly of terminal symbols. In most occidental scripts, a period stops the sentence fully. This program collects the exclamation and question mark to be treated as full stops. Colons, semi-colons, and select other punctuations may be considered in the same class. Within the oriental literature, the convention from antiquity was for the reader to assume the ending of a sentence or provide a larger space than was usual as shown in Figure 1.2. Due to this, Chinese, Korean, and Japanese all have contrastive terminal punctuations left behind from their adoption of occidental customs during modernization, which was a chaotic process. These encodings remain in common use. As a result, corpus linguists should be warned when identifying a sentence from such a script through automation.

At this stage, the data is open for general utility. Sophisticated statistics can use this model of contextualized count to retrieve relational geometry within the structure of the corpus given the parameter of contextualization. The internal corpus statistics are pre-calculated. This means that a user will always be able to access the results of a pre-packaged language. User provided corpora are given orthographic statistics after calculation in successive steps. Both the array of unicode expressions and their count are presented to the user to make decisions on what to include.

**evaluation**

The interface is simple and requires only an assumption of which language is being fed as a source as well as the basic use of English. So long as a teacher or learner has either a text file source or folder, the program will segment and run analytical methods to present results. The remaining output can be interpreted by a teacher or student for classroom or personal use in breaking down the contents of the corpus. The information within the given corpora can be used as a list of frequent bigrams. Future iterations may include variable expansion of the context to enclose multiple words and a left span option. A feature for a stop list should also be considered.

New users will find the program easy to use because of the precise goal within the software to provide collocational information. The ability for the process to adapt to the input will compensate for assumptions linguists make when interacting with users without technical training. The ability to filter words that hold external graphemes does most of the work in focusing the efforts of the inquirer, yet there may be a future option to reduce a diacritic vowel into the Latin set. This method of assembly is referred to as 'literalism', namely reconstituting and regulating a corpus at the level of direct interaction with the encoding. Investigators wanting to reproduce this method should be aware that much of the acquired encoding from a wild corpus may have schemes outside of unicode or right-to-left decoding. The presence of these anomalies does make implementing isolation and replacement of conventioanlly mapped encodings more difficult.

The program can be utilized offline without serious need for computation particularly in the manner of reference and utilizing the internal corpus. The size is light in comparison to comparable tools.