A Pilot Study for a Navajo Textbase

Kip Canfield *(canfield@umbc.edu)*
University of Maryland

Introduction

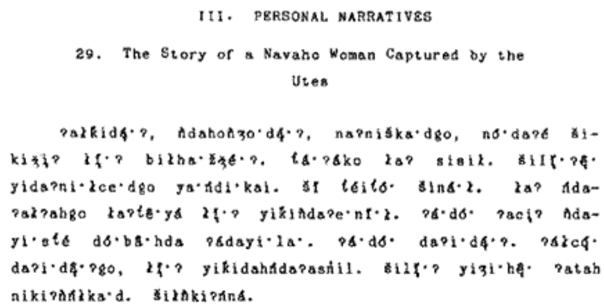
There are a large number of collected and written texts in various Athabascan languages that form a substantial literature that could be used for both scholarship and education. This is especially true of the Navajo language for which there are a large number of written texts, many that are public domain or out of copyright protection. This paper describes and evaluates a project to acquire these texts in electronic format, in the standard orthography, and develop a dictionary lookup tool for use with these texts. Collected texts can take many forms and use many different orthographies. For this pilot study, the Navajo texts are typewriter written with a non-standard orthography. The Navajo language has a polysynthetic structure that poses special problems for dictionary lookup.

Methods

The following steps were used for this project and are detailed below:

1. Scanner acquisition of images of the original texts.
2. Optical character recognition of the text images and post-edit.
3. XML encoding of the texts using the Text Encoding Initiative.
4. Use of an XSLT stylesheet for web display of the texts.
5. Development of an automated look-up tool for the lexicon.

Texts collected in 1929 from the book *Navaho Texts* by Sapir and Hoijer (1942) are used for this pilot project. Figure 1 shows a page fragment from this work that uses a non-standard orthography. Figure 3 shows that same fragment with the standard orthography after acquisition.


III. PERSONAL NARRATIVES

29. The Story of a Navaho Woman Captured by the Utes

Figure 1. A page image fragment from Navaho Texts

After the image of a page has been scanned, it must be recognized using optical character recognition (OCR). For this project, the open-source *Gamera* system was used which is written in the Python language. *Gamera* allows arbitrary characters to be trained using an implementation of the k-nearest neighbor algorithm whose weights are optimized using a genetic algorithm. Figure 2 shows the Gamera interface that allows iterative classification of characters from actual text images and supports training until the error rate is acceptable.

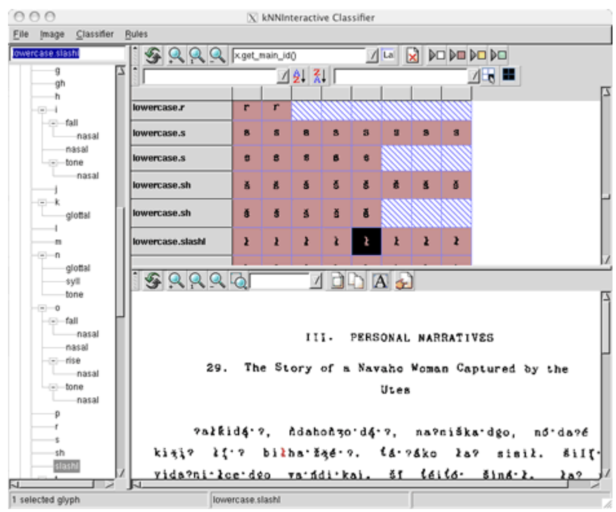


Figure 2. A Gamera Screen shot

The output configuration for a particular project requires Python programming to define the character mappings. The output is a mapping from the recognized characters of the text image to text in the Times New Roman Navajo font. The Text Encoding Initiative (*TEI*) is used for encoding that output. A sample of *Navaho Texts* that has been encoded using *TEI* is shown in Figure 3. It is transformed to HTML using the XSLT stylesheet that is available at the *TEI* website, augmented with a CSS file that includes the Navajo font.

The final step in the workflow is to develop and use a lookup tool for the lexicon that allows a user to click on a word and see the correct dictionary page or easily navigate to it. A major work for the Navajo lexicon is the *Analytical Lexicon* by Young and

Morgan (1992). There is also a project to put the *Analytical Lexicon* (*AL*) on-line that is partially completed and available at <http://www.speech.cs.cmu.edu/egads/navajo/>. The dictionary lookup tool developed here tries to map a verb stem parsed from a word to a page (URL) in this on-line *AL*. The problem is that a morpheme (the stem) must be extracted from the complex morphology of the verb for lookup which is typically a difficult task for users.

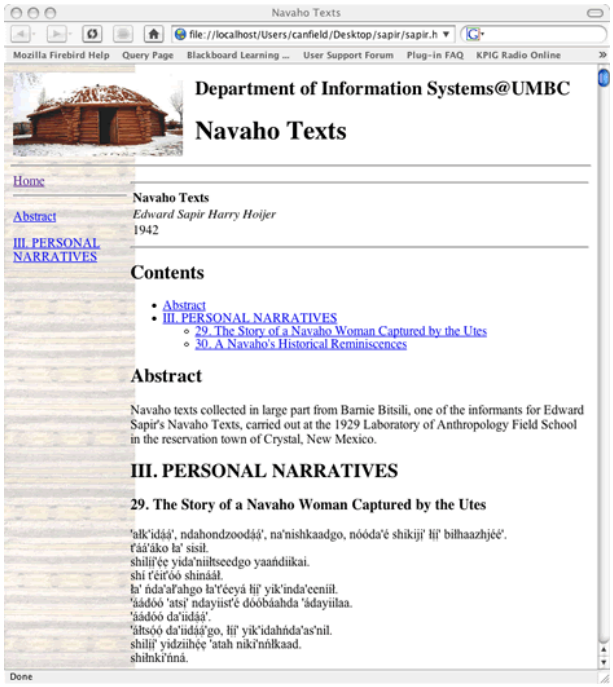


Figure 3. The TEI Encoding

A very simple example of a morphological parser for dictionary lookup is presented here. Pseudo-code for the algorithm is shown below. The algorithm is implemented in the Perl programming language.

1. Get the word
2. Look in a list for direct word lookup (no parsing)
3. If found, display the lexical entry
4. Otherwise:

   (a) Parse the word (assume it is a verb)

   (b) Match the longest common substring to a list of all stem shapes

   (c) Score each match

   (d) Rank the matches by score

   (e) Link each stem match to the URL for the corresponding root in the AL

For step 4a, each substring of the verb is compared to a list of all stem shapes. A simple score is attached to each match in step 4c where:

```
score = (index position of the substring)
* (length of the substring)
```

This privileges matches that are towards the end of the word and longer substring matches. The ranked matches are displayed with the recommended one being the one with the highest score. Example output from the batch version of the Perl program that shows a correct parse is shown below.

- `Word=na'nishkaadgo :`
- `nish - (12) - <http://www.speech.cs.cmu.edu/egads2/navajo/entry?nish>`
- `kaad - (28) - <http://www.speech.cs.cmu.edu/egads2/navajo/entry?kaad>`
- `na' - (0) - <http://www.speech.cs.cmu.edu/egads2/navajo/entry?na%27>`
- `ni - (6) - <http://www.speech.cs.cmu.edu/egads2/navajo/entry?ni>`
- `The recommended stem is kaad`

The highest scored match (28) is the correctly recommended stem. Note that the URL to the on-line *AL* contains still another encoding for Navajo characters (a custom Latin1 mapping that is also URL encoded) and the Perl program must also translate between the standard orthography and this custom mapping.

Results

The scanning procedure is very simple and does not require specialized equipment. The process of scanning does not require any special linguistic expertise and can be carried out as a batch job that produces the image files. The OCR training and classification process using the *Gamera* system is fairly straightforward and with the output programming pieces pre-done for a project, it can be performed by domain experts. The author found that using a training level with an approximately 15% error rate, he could do all acquisition steps of the workflow in under 20 minutes for a physical page and batch pre-scanning would have significantly reduced this time. The *TEI* encoding is simple and ensures that the textbase will be archival.

A formal evaluation of the dictionary lookup tool was performed. A sample of the first 300 words of the text shown in Figure 1 was selected and the Perl program parser was run against this sample in batch mode. This resulted in a list of 300 outputs such as that above. The author then checked each of the 300 parses for accuracy. Three categories were used to evaluate this output: correct, incorrect, and non-verb. The non-verb

category was used for adverbials, nouns, etc. that do not transparently map to verb stems. The result of this evaluation was that the parser was 92% accurate for Navajo verbs with a breakdown of: correct=124, incorrect=10, and non-verb=166. 45% of the sample is non-verb. The remaining problem is how to generally discriminate between verbs and non-verbs for lookup.

Conclusions

The workflow introduced here appears to be useful for domain experts that are trying to create on-line textbases for Athabascan literature. Ultimately these textbases could be used in the schools for language and cultural studies since they are easily implemented as web pages. The dictionary lookup tool goes at least some distance in solving the long-standing problem of helping users to navigate the complex Navajo lexicon. The link to the on-line *AL* is a simple example of cross-project interaction in the computational humanities. An updated programmatic interface to the *AL* would be a significant one.

Bibliography

Sapir, Edward, and Harry Hoijer, eds. *Navaho Texts.* Iowa City: Linguistic Society of America, 1942.

Young, Robert, William Morgan Sr., and Sally Midgette. *Analytical Lexicon of Navajo.* Albuquerque: University of New Mexico Press, 1992.