MITRE: DESCRIPTION OF THE *ALEMBIC* SYSTEM AS USED IN MET

John Aberdeen, John Burger, David Day, Lynette Hirschman, David Palmer, Patricia Robinson, and Marc Vilain

The MITRE Corporation
202 Burlington Rd.
Bedford, MA 01730

{aberdeen, john, day, lynette, palmer, parann, mbv}@mitre.org

Alembic is a comprehensive information extraction system that has been applied to a range of tasks. These include the now-standard components of the formal MUC evaluations: name tagging (NE in MUC-6), name normalization (TE), and template generation (ST). The system has also been exploited to help segment and index broadcast video and was used for early experiments on variants of the co-reference identification task. (For details, see [1].)

For MET, we were of course primarily concerned with the foundational name-tagging task; many downstream modules of the system were left unused. The punchline, as we see it, is that *Alembic* performed exceptionally well at all three of the MET languages despite having no native speakers for any of them among its development team. We were one of only two sites that attempted all three languages, and were the only group that exploited essentially the same body of code for all three tasks.

RULE SEQUENCES

The crux of our approach is the use of rule sequences, a processing strategy that was recently popularized by Eric Brill for part-of-speech tagging [2]. In a rule sequence processor, the object is to sequentially relabel a body of text according to an ordered rule set. The rules are evaluated in order, and each rule is allowed to run to completion only once in the course of processing. The result is an iteratively-improved labelling of the source text. In the name-tagging task, for example, the process begins with an approximate initial labelling, whose purpose is simply to find the rough boundaries of names and other MET-relevant forms, such as money. This rough labelling is then improved by applying a rule sequence. Individual rules then refine the initial rough boundaries, determine the type of a phrase (person, location, *etc.*), or merge fragmented phrases into larger units. See Figure 1 below.

The rules themselves are simple. The two below come from the actual sequence for Spanish MET.

```
(def-phraser
  label          NONE
  l-word-1       lexeme "asociación" ...
  label-action   ORGEX)

(def-phraser
  label          ORGEX
  right-1        lexeme "de"
  right-2        phrase NONE
  bounds-action  merge)
```

Consider how these rules apply to the string "Asociación de Mutuales Israelitas Argentinas". First, the initial labelling breaks the string into components on the basis of part-of-speech taggings:

```
<none>Asociación</none> de
<none>Mutuales Israelitas Argentinas</none>
```

The first rule searches for organizational head nouns, *e.g.*, "asociación" and others, and marks any matching phrase as an organization (ORGEX in our local MET dialect). This yields the partial relabelling:

```
<orgex>Asociación</orgex> de
<none>Mutuales Israelitas Argentinas</none>
```

The second rule applies to isolated organization head phrases, and merges in their complements:

```
<orgex>Asociación de
Mutuales Israelitas Argentinas</orgex>
```

MET-SPECIFIC DEVELOPMENT

In the course of MET, we ported the *Alembic* name tagger to all three of the target languages. We did so with essentially no guidance from native speakers of any of these languages. For Spanish, two of us collaborated to develop a rule sequence by hand; to this task, one of us brought two semesters of college Spanish, and the other brought fluency in French. With help from a good dictionary and atlas, we were able to understand the training texts well enough to grasp their critical semantics, or as much of the semantics as was needed for the purpose of name tagging. For Japanese, one of us taught himself to read Kanji at a fifth-grade level, and developed a name-tagging sequence through repeated scrutiny of

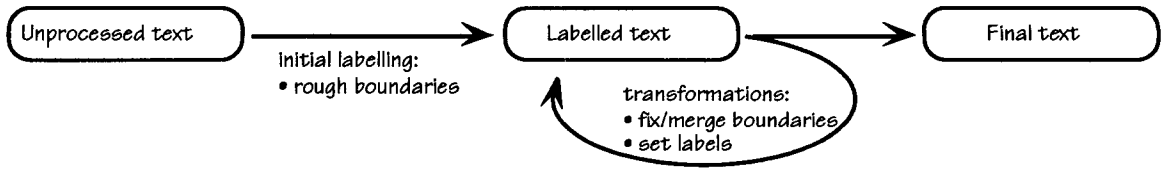

Figure 1: Brill's rule sequence architecture as applied to phrase tagging.

the training texts. It is important to note that our lone Japanese-MET developer had only passing understanding of the texts he was reading. The development process for him consisted largely of Kanji pattern-matching (as opposed to *bona fide* reading). Finally, for Chinese, we had not even the limited reading ability available for Japanese. Aside from date and money patterns, the entirety of the Chinese rule sequence was acquired through a machine learning process.

Besides these rule sequences, several language-specific extensions were required to port *Alembic* to MET. As we needed to segment Chinese and Japanese texts into separate tokens we adapted the NEW-JUMAN tagger/segmenter for Japanese, and the NMSU segmenter for Chinese. In addition, our Spanish system exploited a Spanish part-of-speech tagger that we had developed previously.

RESULTS

The preliminary nature of the MET task precludes formulating a full assessment of our system's performance. Nevertheless, we are pleased with our early results. *Alembic* either exceeded or came near matching its performance on the English name-tagging task in MUC-6. The chart in Fig. 2 shows the relative rankings of the four languages (solid bars indicate training, and shaded ones formal testing).

These results show gaps between training and testing performance, especially in the two Asian languages. Part of these differences can be attributed to inconsistencies that were eventually detected in the final test data. This may account for much of the 10% training-to-testing gap in Chinese. Indeed, on a held-out development test set, Chinese performance was virtually identical to that on the development training set; the learning procedure had thus acquired a very predictive model of the development data overall. However, since the tagging conventions on the formal test set were not wholly consistent with those in the training set, the performance of the model could only be expected to decrease in the final evaluation. For Japanese, a similar problem arose because refinements to the guidelines over the course of MET development were not reflected in the development data set. Since our Japanese developer could not actually read most of the Japanese material, he could only interpret changes to the guidelines in so far as they were incorporated in the training set. As the guidelines and training set drifted further apart, this led increasingly to the same inconsistencies we experienced with Chinese.

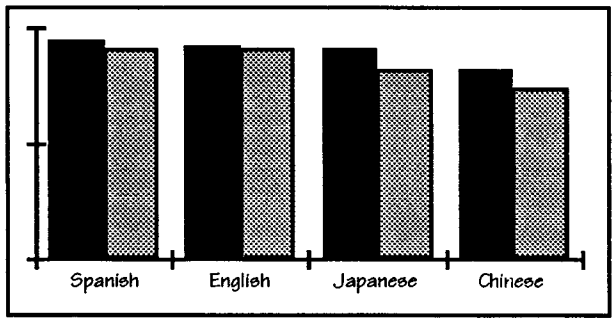

Figure 2: Name tagger rankings, by language.

We should not let these error analyses obscure *Alembic*'s achievements, however. The system garnered commendable scores on all three languages, despite its developers having at best passing linguistic fluency—and in one case no language knowledge at all. We think this success is due to several factors. First, the inherent speed of the system (25,000-30,000 words per minute) enables a rapid-evaluation methodology. For manual engineering, this allows changes in the model to be implemented and tested efficiently. Second, *Alembic* supports the developer through a growing suite of tools, chief among them the phrase rule learner. Finally, we owe the bulk of the system's success to the underlying framework with its emphasis on sequences of simple rules.

REFERENCES

[1] Aberdeen, J., Burger, J., Day, D., Hirschman, L., Robinson, P., & Vilain, M. (1995) "MITRE: description of the *Alembic* system used in MUC-6." In Sundheim, B. (ed.), *Prcdgs. of the Sixth Msg. Understanding Conference.* Columbia, MD.

[2] Brill, E. (1993). *A corpus-based approach to language learning.* Doctoral Diss., Univ. of Pennsylvania.