(15) SELECT (d.det) (-1C (adj) LINK T:PreviousFinal LINK 0C (n.count)) (0 ("<རང>") LINK T:IsWord) (1 ("<མི>") LINK T:IsWord) (2 ("<འདུག་?>"r) LINK T:IsWord) ;

As noted in the previous section, special care must be taken with abbreviated syllables. The genitive case marker, for instance, manifests as the abbreviated syllable འི, as in (5) above, or as one of several standalone syllables. Therefore, a rule such as the following which specifies ལས་ as a noun rather than ablative case if preceded by a genitive must accommodate both standalone (16) and abbreviated case (17).

(16) SELECT (n.count) (-1 ("<(གི་|ཀྱི་|གྱི་)>"r) LINK T:IsWord) (0 ("<ལས>") LINK T:IsWord) ;

(17) SELECT (n.count) (T:PrevAbGen) (0 ("<ལས>") LINK T:IsWord) ;

(18) TEMPLATE IsAbGen = 0 ("<.+འི>"r) ; TEMPLATE PrevAbGen = T:PrevFinal LINK T:IsAbGen ;

In other cases, syllable-based tagging obviates the need for specific rules relating to abbreviated syllables. For example, (19) removes noun tags from ས་, provided that it is not preceded by end of sentence punctuation or by an intersyllabic tsheg. In addition to being a freestanding word meaning "earth", ས་ also marks agentive case when attached to open syllables, as in (4) above.

(19) REMOVE n.xxx (-1 tshegless) (0 (case.agn) LINK 0 ("<ས་?>"r)) ;

(20) SET shad = ("<[།༎]>"r) ; SET tshegless = ("<.*[^་]>"r) - shad ;

No such rule is needed in the syllable-based rule tagger. If ས་ is attached to the preceding syllable, then that syllable will be tagged SS or ES, and the hypothesis that ས་ means "earth" will simply not arise.

4 Future directions

The syllable-based tagger adds complexity to the word-based tagger without improving its overall performance. So why use it?

In a traditional pipeline approach, tokenization or segmentation precedes part-of-speech tagging, with the output of the former process feeding the latter. This has the disadvantage that errors made at earlier stages in the pipeline propagate to later stages. The success of the pipeline is effectively limited by the quality of its initial component. So a Tibetan POS-tagger can only be as good as the word segmentation system that precedes it.

Another common limit of traditional pipelines is that different components often require that the data be represented in different ways. In practice, this can be an obstacle to component interaction. By bringing the data requirements of the word segmenter and the POS-tagger in line with each other, we hope to facilitate the development of more cooperative NLP components, including a joint approach to segmentation and tagging.[4]

References

Eckhard Bick and Tino Didriksen. 2015. CG-3 - Beyond classical constraint grammar. *Proceedings of the 20th Nordic Conference on Computational Linguistics (NODALIDA 2015)*, pages 31-39.

Edward Garrett, Nathan Hill and Abel Zadoks. 2014. A rule-based part-of-speech tagger for Classical Tibetan. *Himalayan Linguistics*, 13(1):9–57.

Hans van Halteren. 1999. Performance of taggers. In Hans van Halteren (ed.), *Syntactic Wordclass Tagging.* Springer: Netherlands, 81-94.

Huidan Liu, Minghua Nuo, Longlong Ma, Jian Wu, and Yeping He. 2011. Tibetan word segmentation as syllable tagging using conditional random field. *25th Pacific Asia Conference on Language, Information and Computation,* 168–177.

Hwee Tou Ng and Jin Kiat Low. 2004. Chinese part-of-speech tagging: One-at-a-time or all-at-once? Word-based or character-based? *Proceedings of EMNLP.* Barcelona, Spain.

Nianwen Xue. 2003. Chinese word segmentation as character tagging. *Computational Linguistics and Chinese Language Processing*, 8(1):29-48.

[4] The Classical Tibetan corpus, as well as the various rule taggers described in this paper, are available on the following GitHub site: https://github.com/tibetan-nlp