himalayan linguistics

A free refereed web journal and archive devoted to the study of the languages of the Himalayas

Himalayan Linguistics

Extending corpus annotation of Nepali: advances in tokenisation and lemmatisation

Andrew Hardie
Lancaster University, UK

Ram R. Lohani
Yogendra P. Yadava
Tribhuvan University, Nepal

ABSTRACT

The Nepali National Corpus (NNC) was, in the process of its creation, annotated with part-of-speech (POS) tags. This paper describes the extension of automated text and corpus annotation in Nepali from POS tags to lemmatisation, enabling a more complex set of corpus-based searches and analyses. This work also addresses certain practical compromises embodied in the initial tagging of the NNC. First, some particular aspects of Nepali morphology – in particular the complexity of the agglutinative verbal inflection system – necessitated improvements to the underlying tokenisation of the text before lemmatisation could be satisfactorily implemented. In practical terms, both the tokenisation and lemmatisation procedures require linguistic knowledge resources to operate successfully: a set of rules describing the default case, and a lexicon containing a list of individual exceptions: words whose form suggests a particular rule should apply to them, but where that rule in fact does not apply. These resources, particularly the lexicons of irregularities, were created by a strongly data-driven process working from analyses of the NNC itself. This approach to tokenisation and lemmatisation, and associated linguistic knowledge resources, may be illustrative and of use to researchers looking at other languages of the Himalayan region, most especially those that have similar morphological behaviour to Nepali.

KEYWORDS

Nepali, corpus, tagging, lemmatisation, tokenisation, morphology

This is a contribution from *Himalayan Linguistics, Vol. 10(1) [Special Issue in Memory of Michael Noonan and David Watters]: 151–165.*
ISSN 1544-7502

Himalayan Linguistics, Vol. 10(1).
ISSN 1544-7502

Extending corpus annotation of Nepali: advances in tokenisation and lemmatisation

Andrew Hardie
Lancaster University, UK

Ram R. Lohani
Yogendra P. Yadava
Tribhuvan University, Nepal

1 Introduction

Nepali is an Indo-Aryan language spoken by approximately 45 million people in Nepal – where it is the language of government and the medium of much education – and in neighbouring countries (India, Bhutan, and Myanmar). It serves as the lingua franca of an extremely multilingual area of the world; more than 90 languages are spoken within Nepal.[1] Nepali is written in the Devanagari alphabet and has a written tradition extending back to the twelfth century CE. Within the last few years, corpus resources have been developed for Nepali for the first time in the form of the Nepali National Corpus (NNC; see Yadava et al., 2008), developed by the *Bhasha Sanchar* ("language communication") project, also known as *Nelralec*.[2] A variety of Nepali language technology support projects were undertaken within *Nelralec,* including software localisation and font development. This process also created several resources for automated annotation of linguistic analyses of texts and corpora, most notably part-of-speech (POS) tagging. These analytic procedures were applied to, and are available within, the released version of the NNC. In this paper, we describe subsequent work to develop advances on the corpus annotation present in the NNC, with a focus on two forms of analysis in particular – namely, tokenisation (the splitting of text into word-level units of analysis) and lemmatisation (the assignment of every token to a "dictionary headword" or lemma).

After giving an overview of the context of this work (§2), namely the design and composition of the NNC, we will discuss points of the original word-level annotation in the corpus which represent sub-optimal compromises made for practical reasons to address challenging aspects of morphology (§3) – compromises which, we would argue, are no longer necessary in light of the advances in lemmatisation and tokenisation which we present here. We will detail the mechanics of these advances, discussing the software used and the linguistic knowledge resources on which they rely, for tokenisation (§4) and lemmatisation (§5) respectively. In both cases, the software requires

1 According to the 2001 census of Nepal (CBS, 2001).

2 Nelralec: *Nepali Language Resources and Localization for Education and Communication;* funded by the EU Asia IT&C programme, reference number ASIE/2004/091-777. See http://www.bhashasanchar.org

both a set of rules telling it how to deal with most words, and a lexicon which represents a repository of exceptions. Finally, we will present a number of applications of the enhanced automatic annotation within the context of corpus-based research on Nepali (§6).

2 The Nepali National Corpus (NNC): a cursory overview

Although our concern in this paper is to detail advances in the annotation of Nepali texts and corpora, we will in this section present an overview of the composition of the NNC, to give some idea of the context in which this work took place and the research applications that these developments targeted.

Component	Contents	Size in words (approx)
NNC: Core Sample	Written texts sampled as a Nepali match for FLOB and Frown (1990s)	800,000 words
NNC: General Collection	Written texts opportunistically collected, including text from the Web	13,000,000 words
NNC parallel data	Written texts with translations, Nepali-English and English-Nepali	4,000,000 words
NNC spoken corpus	Spoken texts	260,000 words
NNC speech corpus	Audio recordings of sentences for use in text-to-speech applications	6,000 words

Table 1. Overview of the NNC.

The NNC is designed as a multi-modal corpus, containing data from speech and writing. Indeed, several different types of both spoken and written data were collected, such that the NNC as a whole is actually a compendium of corpora each with its own range of potential applications. The main sections of the corpus, and their sizes, are detailed in Table 1. The written part of the NNC is by far the largest, and consists of two corpora designed on a "core-penumbra" basis. That is, a smaller "core" corpus was designed according to an exacting sampling frame, that of the FLOB and Frown corpora of British and American English (Hundt et al. 1998; Hundt et al. 1999). This smaller section is referred to as the *Core Sample (CS)* of the NNC. The primary purpose of the Core Sample was to provide a match to other corpora created from the same sampling frame, including not only Frown and FLOB but also the Lancaster Corpus of Mandarin Chinese (McEnery and Xiao 2004). However, some adaptations to the framework for genres within this sampling frame were made, since some genres existing in English writings (e.g. Western and adventure fiction) were found not to exist in Nepali, due to cultural and other differences between South Asia and Western Europe.

The larger "penumbra" corpus, referred to as the *General Collection (GC),* was collected opportunistically from a wide range of sources such as websites, newspapers, and books, with non-internet texts being gathered in the main directly from publishers and authors. This part of the corpus was intended to allow corpus analyses that depend on a very large corpus (as opposed to a *balanced* and *representative* corpus; see Leech 2007 for some discussion of these notions). For

example, corpus-based lexicography is dependent on the existence of multiple exemplars of each word to be defined and thus, for all but the most common words, requires a dataset of many millions of words.

The NNC has already been applied to the analysis of certain linguistic questions related to the Nepali language (most notably, the role of collocation in the establishment of grammatical categories: see Hardie 2008, Hardie and Mudraya 2009). It has additionally been exploited within the field of Nepali lexicography, leading to the development of a new dictionary of the contemporary language, the *Samkaaliin Nepaalii Sabdakos*.[3] It has been made widely available[4] and its employment for a diverse range of purposes continues.

3 Word-level annotation in Nepali

3.1 The challenge of Nepali morphology

The morphology of Nepali is at once strongly inflectional and strongly agglutinative. As such, it represents a major challenge for word-level corpus annotation. A full account of these issues and how they have been addressed in the initial annotation of the NNC is given by Hardie et al. (2009); here we will give only a brief overview as context for the extensions to be discussed below.

The strongly inflectional nature of the Nepal language is mostly evident in the verbal morphology, as the case, number and gender inflections inherited from Old Indo-Aryan have been almost entirely levelled out in the Nepali nominal system. By contrast, verb inflections mark both subject agreement for number, person, honorificity and (marginally) gender in finite forms, and also a series of different non-finite forms referred to variously in the literature as participles and infinitives. There are also modal inflections (indicative versus optative and imperative) and an inflected passive, causative, and negative.

In addition to highly productive compounding of independent roots to form new lexemes, agglutinative processes play a role in the grammar of both nouns and verbs in Nepali, in contrast to the inflectional system. Several additional verbal categories are marked by compounding of non-finite lexical verbs with finite auxiliary verbs. Many tense-aspect combinations are indicated in this way. Examples from the paradigm of *garnu* "do" include forms such as *garthyo* "(he) used to do", *garcha* "(he) does", *garnecha* "(he) will do", and *garirahyo* "(he) continued to do" – where *thiyo* and *cha* are finite forms of *hunu* "be", and *rahyo* is a finite form of *rahanu* "stay". Forms of *hunu* in particular are used as compounded auxiliaries to indicate most Nepali tense forms. It is also through this kind of compounding that the higher levels of verb honorificity are marked.

The elements that are typically compounded to nouns, and in some cases to pronouns, adjectives and other similar categories, are the markers of semantic and/or grammatical role that, in the literature on Nepali, are variously referred to as case markers or postpositions (the latter being the term that we adopt). These include ergative *le,* accusative-dative *laaii,* genitive *ko/kaa/kii,* and locative *maa*. Similar in form, but distinct in function, is the collective/plural marker *haruu*. A single noun root can be followed by multiple postpositions in a range of configurations.

Finally, it should be noted that in many cases, there is inconsistency in current and recent Nepali orthographic practice concerning whether some of the agglutinative elements discussed

3 See http://www.nepalisabdakos.com

4 See http://www.bhashasanchar.org/aboutnnc.php . Distribution of the NNC is now handled by the *Language Technology Kendra* organisation (http://www.ltk.vacau.com).

above – especially postpositions – are to be written as a single word with the element they are attached to or not. For example, instances of *maa* attached to a preceding noun and instances with an intervening space may both be observed. The general standard, albeit one far from universally followed, has been to write them together; however, quite recently there appears to have been a move towards the practice of writing them separately.

3.2 Compromises in the original NNC part-of-speech annotation scheme

In developing a POS tagging scheme for the first release of the NNC (Hardie et al., 2009), referred to as the Nelralec Tagset, certain compromises were made in order to make the task more tractable. In particular, full consistency between the treatment of nouns and verbs in the area of *tokenisation* was sacrificed.

Tokenisation is a necessary first step to POS tagging: the very act of assigning grammatical tags to a text assumes that appropriate units have been identified to which tags can be given. This is not necessarily a straightforward process. Even in a language such as English, where the notion of the word or *token* as a linguistic unit corresponds very closely to the orthographic (whitespace-bounded) word, some adjustment to the tokenisation is undertaken prior to POS analysis. For instance, the wordform *don't* is typically tokenised as *do n't:* the resulting two-token sequence can then be given the same tags as *do not,* enabling the difference between these two realisations of the negative particle to be abstracted away in corpus searches and frequency counts if necessary. In a language such as Nepali, with the extensive grammatical agglutination noted above, there are three possible ways to approach tokenisation:

- Perform no tokenisation (apart from the basic separating-out of punctuation), and tag each multi-element token according to all the grammatical distinctions indicated by the elements.
- Perform no tokenisation, but tag each multi-element token in a simplified manner based on a subset of the grammatical distinctions that are present.
- Perform extensive tokenisation, splitting apart multi-element words into strings of tokens which can then be individually tagged.

It might be questioned why the third option, of splitting apart multi-element words, is even under consideration in this context. The reasons are twofold. Firstly, and crucially, elements within a Nepali compound are often (inflectionally and in some cases functionally) identical to units that can be found elsewhere in texts. For example, the elements of verb *garnecha* "(he) will do", namely *garne* and *cha,* both occur as independent words (a participle of *do* and a finite form of *hunu* "to be", respectively). Secondly, and as noted above, the use of spaces in Nepali orthography is not fully consistent. That means that there are cases where these elements are already, in effect, "pre-split" before analysis begins; splitting the other cases then leads to consistency. Both these issues would reduce the precision/recall and accuracy of computerised corpus searches for, and frequency counts of, different linguistic phenomena – whereas consistency in encoding makes it possible for such analyses to be substantially more thorough and rigorous. So while this third strategy does involve making changes to the original word divisions of the corpus text, there are strong advantages in terms of the tractability of subsequent analysis.

The other two strategies above have drawbacks based on the schemas of analysis that they

necessarily imply. The disadvantage of the first strategy in particular is that an attempt to capture in an annotation scheme all the grammatical categories indicated by a potentially long string of compounded elements leads to massive inflation of the tagset. The second strategy avoids this problem, but only at the cost of reducing the resolution of the annotation. So for postpositions in Nepali, a strategy of extensive tokenisation allows a more consistent and comprehensive analysis than the other two strategies. These considerations led us to adopt this third strategy for nouns (and categories with similar behaviour such as adjectives and determiners).

It would, of course, be possible to treat compounded auxiliary verbs in the same manner – splitting them apart from the verbs they are attached to and tagging them separately. However, a retokenisation solution is not as straightforward for verbs as for nouns and postpositions, since there are some forms within some verbal compounds that are not identical to a free-standing element. For instance, the verb form *huncha* "is" is composed of *hun* and *cha,* but while *cha* is also a freestanding element, *hun* is not (the independent equivalent would be the root form *hu* "be"). Moreover, the number and the complexity of different elements which would need to be separated out is much greater in the case of auxiliary verbs than in the case of postpositions. For this reason, the retokenisation approach was avoided for verbs in the Nelralec Tagset. Instead, the second of the approaches noted above was adopted for verbs: each verb was tagged as a whole, but with simplifying assumptions to prevent the number of categories within the tagset growing massively.[5] While making the task of POS tagging suitably tractable, this introduced an unavoidable inconsistency between the analysis of verbs (no retokenisation, simplified analysis) and the analysis of nouns (retokenisation, full analysis) in the NNC. It also created an obstacle to adding lemmatisation to the annotation of the NNC, namely that while we would ideally annotate the lemmata of all elements of a compound to enable maximal recall in retrieving instances of those lemmata, this was not currently supported by the tokenisation. Thus, when we came to consider extending the annotation, we had a twofold goal: (a) to amend the tokenisation system, and its realisation in the NNC, to reverse this compromise and make it fully consistent by splitting verbal complex elements such as *garnecha, huncha* and *garirahyo* just as noun-plus-postposition elements are split; and (b) to use this improved tokenisation to enable subsequent lemmatisation.

4 Resources for (re)tokenisation

The creation of our extended tokenisation system for Nepali was focused on the development of certain necessary linguistic knowledge resources. The software that implements the Nepali POS tagger is the *Unitag* system (Hardie 2004, 2005), a modular tagger which incorporates a language-independent configurable tokeniser. This *Unitokenise* module operates in two passes across each text file. In the first pass, it inserts an explicit token break at every whitespace, at every punctuation mark, and at every XML element – in effect, applying "dumb" tokenisation. In the second pass, it moves across the tokenised text adjusting the token breaks in accordance with a list of *tokenisation rules*. These are a language specific resource, external to the program. They are defined as a pattern around a token break; if the pattern is matched then the rule is applied. The rule may be to "split"

5 Many thousands of different verb tags would be required to capture all the tense-aspect, mood, voice, person, number, gender, and honorificity distinctions made via compounding of auxiliary verbs in Nepali. The amount of manually-annotated data required to train a POS tagging system based on tag n-gram frequencies, such as a Markov model (see El-Beze and Merialdo 1999), is proportional to the n^{th} power of the size of the tagset, so such a large number of distinctions would clearly not be practical.

(insert a token break) or to "merge" (remove a token break). The process is cyclical (that is, newly-created tokens are checked to see if any of the retokenisation rules can apply to them) so multiple tokens can be either split off from a single orthographic word, or merged together.[6] For instance, the splitting-off of the locative postposition *maa* in the original version of the Nepali tokeniser used in the tagging of the NNC was controlled by the following rule:

```
split #|मा
```

The | indicates the position of the token break; the # is a wildcard indicating "any string of characters to the end or beginning of the word". Like all resources in the tokeniser and tagger, and the NNC itself, the rules use the Devanagari script and the Unicode character set. So this rule could be paraphrased as "if the word ends in *maa* (मा), add a token break before the *maa*". To handle the tokenisation of postpositions, around 50 such rules were required. The tokeniser also uses an additional resource, an exceptions lexicon. This contains all wordforms which should *not* be split or merged according to the specified rules, even though they match the pattern. For example, the word *laamaa* "Buddhist priest" ends in *maa,* but the *maa* here is not locative: it is simply part of the noun root. Thus, *laamaa* is listed in the exceptions lexicon, and will never be altered by the splitting-rule given above.

A much more extensive set of rules (and accompanying exceptions lists) were required to handle the retokenisation of verbs. Whereas postpositions are generally invariant in form or have few variants (for instance, the plural *haruu* has the variant spelling *haru*), auxiliary verbs – as noted above – have many different forms inflected for person, number, gender and honorificity. The first step in assembling retokenisation resources for the verbs was to address the splitting-off of the various forms of *hunu* "be", the tense-aspect marking auxiliary. A list of forms from the paradigm of *hunu* was created and used to generate a batch of splitting rules. However, other verbs also form verbal compounds. Like many Indo-Aryan languages, Nepali has a series of "light" or "vector" verbs which, while they do not mark tense-aspect combinations as the auxiliary *hunu* does, are formally similar to auxiliaries. They thus also need to be tokenised separately, so that both the light/vector verb and the main verb can receive a POS tag and, ultimately, a lemma. In contrast to the case of *hunu,* to exhaustively list all forms of all possible light/vector verbs would be much very difficult relying on intuition alone; thus, the NNC corpus data was exploited.[7] A list of all tokens tagged as verbs in the original NNC annotation was generated, ordered by frequency. This list was scrutinised to identify complex verbs with a frequency above 50 in the whole NNC.[8] Manually decomposing these highly-frequent complex verbs produced a list of vector forms, from which an-

6 The output of the tokeniser contains only the words of the text in their retokenised form, together with an inter-token trace which indicates where a split has taken place (which may or may not be preserved by subsequent processing, according to the requirements of the task at hand). The original orthographic form of the words is not present in the output, but is of course preserved in the input data, and a typical approach would be to archive/distribute a corpus in both the original format and the fully-annotated (tokenised, POS tagged, lemmatised) format.

7 All corpus analyses undertaken in the creation of the linguistic knowledge resources described here were accomplished using the *CQPweb* corpus analysis software (see Hardie, forthcoming).

8 The cut-off point of 50 was determined by the limitations of analyst time available for the development of the tokeniser's linguistic resources. However, this is very considerable coverage – the whole corpus contains 2,255,827 tokens tagged as verbs, accounted for by 71,688 types; of these 3,177 types, with a cumulative frequency of 1,994,787 tokens, have a frequency of 50 or more. So types representing 88% of the verb tokens in the corpus were included in the manual analyses that produced the tokenization rules.

other batch of rules could be generated. This corpus-based approach ensured that, although complete coverage of all possible vector verb-forms could not be achieved, all highly frequent vectors were addressed, thus covering the vast majority of cases. Finally, all simple verbs with a frequency higher than 50 that could possibly occur within a complex verb form were listed (again from the NNC data), and rules generated for these forms.

Such an enhanced rule set necessarily requires a longer list of exceptions. Some exceptions were added to the lexicon as they were encountered in manual analysis of the data. However, a more systematic approach was needed. Firstly, the full list of retokenisation rules was run across a subset of the NNC Core Sample that, in the process of developing the POS tagset and tagger, had been manually analysed. The tokeniser's actions – that is, the list of words split apart – were recorded, and any of this output that was incorrect was used to generate an entry in the exceptions lexicon. Secondly, collocation searches on the words immediately preceding the split-off elements were used to identify exceptions.[9] The logic behind this was as follows. If a collocation search for a separated token (such as *maa* or some form of *hunu*) yields a frequent collocate in the position one-token-right of the search node, and that collocate is infrequent or absent in the rest of the corpus, then there is a reasonable probability that that "collocate" is actually part of a word that has been split, but should not have been. This process yielded a much expanded list of exceptions, which, again, are empirically known to provide the highest possible coverage across the contents of the corpus.

In their final state, the retokenisation resources consist of around 2,400 tokenisation rules and an exceptions lexicon of around 3,800 items. The number of rules is high because so many contain lexically-specific material; we take no stance on whether or not these kinds of tokenisation rule reflect accurately how any part of the linguistic knowledge of human speakers is actually structured. Many rules have multiple exceptions, so in theory the list of exceptions should be substantially longer than the list of rules; but the disparity in size of the two lists reflects the priority given to rules/exceptions whose operation is highly frequent, to maximise coverage. This revised tokenisation system was run on the tagged NNC data, and POS tags assigned to the newly-generated tokens, prior to lemmatisation.

5 Resources for lemmatisation

5.1 The goal of lemmatisation in Nepali

Given the nature of Nepali morphology outlined above, the goal of lemmatisation was to map each inflected form to a single "basic" form. In the case of adjectives and determiners with gender/number inflection, the "basic" form was deemed to be the masculine singular (the root plus suffix *o*). For verbs, it was deemed to be the infinitive (the root plus suffix *nu*). For other words which do not inflect (many nouns and adjectives, most postpositions) each wordform is its own lemma. These decisions were taken in accordance with Nepali grammatical and lexicographical practice – for instance, in Nepali grammars the infinitive is the citation form of the verb, and it is the infinitive

9 A *collocation* is generally defined in corpus-based analysis as a co-occurrence relationship between one word (a node) and another (the collocate) that is significantly more frequent in the near vicinity of the node than it is elsewhere in the corpus (but see McEnery and Hardie 2011, chapter 6 for discussion of complexities of this definition). The window of "near vicinity" is typically three to five words to the left and to the right of the node word. In CQPweb, the window is configurable, and in this analysis, we ran collocation searches solely for collocates one word to the right of the node.

that is listed in dictionaries.

Since Nepali texts often contain variant spellings, it was deemed that the lemma of a variant spelling should be the more standard form. So, for instance, the plural marker *haruu* has a variant *haru;*[10] the target lemma for both was *haruu*. The same approach was taken to morphophonemic variants. The placeholder PUNC was the target lemma for all punctuation marks.

5.2 Unilemma software

As with tokenisation, a *Unitag* module was used for lemmatisation which at the level of the software is language-independent but which requires appropriate linguistic knowledge resources. This module, *Unilemma,* is an extension to the *Unitag* system originally described by Hardie (2004, 2005).

Unilemma follows a three-step procedure to lemmatise each token. First, the word is looked up in a lemmatisation lexicon; if it is found, the specified lemma is assigned. Failing that, in the second step the system runs through a list of lemmatisation rules; if any of the rules applies, that rule is used to generate a lemma form. If neither of these procedures produces a lemma, then as a fallback procedure, the unaltered wordform is assigned as a lemma. Although the system is different in most details of implementation to the tokenisation system, the general outline of the linguistic knowledge resources is the same – a set of rules that operate on most words, together with a lexicon containing those cases which do not comply with the principles those rules embody (as well as, for performance reasons, especially frequent forms).

5.3 Lemmatisation lexicon

Lemmata for all the most frequent words in the language were defined in the lexicon, including most closed-class words. Furthermore, lemmatisation of spelling variants and morphophonemic variants was accomplished by listing them within the lexicon. As with tokenisation, the lexicon's contents were targeted at highly frequent items to achieve the broadest coverage possible. To accomplish this, a corpus-derived frequency list for each part-of-speech category was extracted and examined. However, although the lists of words to be included in the lexicon were created semi-automatically, the specification of the actual lemma was in all cases done manually. The lemma lexicon finally contained around 4,000 items.

Unilemma is able to understand basic conditional entries in the lemma lexicon. If a wordform is ambiguous between parts-of-speech, it is possible that its lemma may vary depending on what part of speech it has. For example, the word *din/dina,* a homonym (दिन) in Devanagari, may be a form of the verb *dinu* "give" or it may be the noun *din* "day". The lexicon entry for this word is as follows:

दिन दिनु#V दिन#N

In this case, the # separates a possible lemma from a POS-tag criterion. For any given instance of *din/dina,* if it has been given a noun tag in context (beginning with N), it will receive the lemma *din*. If it has been given a verb tag (beginning with V) it will receive the lemma *dinu*. Thus,

10 Here and throughout, doubled <u> and <i> in our transliteration indicates the distinction between the long and short Devanagari vowel characters, a distinction which is not actually reflected in Nepali phonology.

the lemmatisation depends on the previous POS analysis.

It should be noted that the POS analysis is not infallible (the accuracy rate of the Nepali instantiation of *Unitag* is around 93%)[11] and errors distinguishing in cases like nominal *din* versus verbal *dina* will therefore cascade into the lemmatisation. This should not be seen as a flaw in the lemmatisation resources *per se,* but rather the accepted consequence of using less-than-perfect automated tools for tasks such as POS tagging. There are approaches to corpus annotation where the correctness of the annotation procedure is enhanced by allowing ambiguous output. In the context of POS tagging, this is usually called *k-best* or *n-best* tagging (e.g. by Voutilainen 1999: 10): where the choice of a tag is uncertain, the *n* most likely options are all presented in the output. The same principle can be applied to lemmatisation. So, in the case of *din/dina,* it would be possible for instances of दिन where the noun analysis is preferred but the verb analysis remains feasible to be tagged NN–VI and lemmatised as *din–dinu.* We did not adopt this approach. It has been our experience[12] that such ambiguous annotations complicate utilisation of the corpus by the end user considerably. For instance, with ambiguous annotation, the corpus frequencies of the *din* and *dinu* lemmata would be spread across four items (*din, dinu, din–dinu,* and *dinu–din*) rather than two, with the consequent need for the researcher to account for this somehow in all quantitative analysis of the data. This raises a substantial barrier to usability; put bluntly, when working with corpus annotation, a firm decision which has a known error rate that the researcher can accept or not, depending on their purpose, is more tractable than a mixture of firm decisions (with an error rate) and infirm decisions (also with an error rate, but not the same error rate) in a proportion that may vary from word to word.

5.4 Lemmatisation rules

While many verb forms, including all the most common and all with irregular inflections, have their lemmata listed in the lexicon, the very large number of inflected forms that exist for each verb – even after retokenisation of agglutinated complex verbs – mean that coverage of the language as a whole using just the lexicon was less than ideal. This also applied, to a lesser extent, to inflected adjectives (note that the majority of adjectives in Nepali do *not* inflect for gender/number). To address this, rules for deducing the appropriate lemma for forms not in the lexicon were devised. The rules have three parts: they specify a word template, a POS-tag template, and a transformation. For a given token, if its wordform matches the word template and its tag matches the POS-tag template, then the transformation is applied to deduce the lemma. This is somewhat hard to understand in the abstract, so let us consider a concrete example (with transliteration):

```
#ी   JF   #ी   #ो
#ii  JF   #ii  #o
```

The word-template is the first column, the tag template the second column, and the transformation is defined by the third and fourth columns. A # represents the rest of the word or tag. What this indicates is that if a word ends in *ii,* and has the tag JF (feminine adjective), then a

11 See §5.5 below for details of how accuracy rates were determined.

12 This experience is based on working with such English datasets as the British National Corpus, in which tags such as NN2–VVZ and VVZ–NN1 are used in cases where the tagger was uncertain of the choice between NN2 (plural noun) and VVZ (third person singular verb); the tag placed first is judged more likely.

lemma will be deduced by removing the *ii* (the feminine suffix) and adding *o* (the masculine suffix). Applied to a feminine adjective such as *raamrii* "good", for instance, this would produce the correct lemma *raamro*. Although the element removed by the transformation *may* be (and often is) the same as the suffix sought in the word-template, it need not be.

An example verbal rule is the following:

#यौ	VV#	#यौ	#नु
#yau	VV#	#yau	#nu

The suffix *yau* is the marker for the past tense second person plural (or singular medial-honorific), for example *garyau* "you did" which would map to the infinitive *garnu*. A word ending in *yau* must have a tag beginning in VV for this rule to apply – in the Nelralec Tagset, all indicative finite verb tags begin with VV.

The use of tag templates as well as word templates in lemmatisation rules constrains the lemmatiser from overgenerating – applying its rules in cases where they do not apply, such as nouns which happen to end in a form which is also a verbal suffix – in the absence of a list of exceptions such as was used in the tokeniser. Of course, the tokeniser cannot rely on tags to constrain it, as tokenisation is undertaken prior to tagging. Errors are still possible, as the tagger itself may be mistaken.

The final rule list contains around 150 rules, all but 10 of which relate to verb inflections, both finite and non-finite. This is a shorter set of rules than were used for tokenisation, since the lexical material they contain (different morphological and orthographical forms of the various suffixes) is more general. Even so, however, a relatively extensive list is produced simply because a single "rule" in the sense of a regularity of Nepali grammar will map to multiple "rules" in the formalism used by the lemmatiser.

An example of the final annotation output is given below; note that though a columnar format is used here, it can equally well be expressed as XML or a horizontally-tagged format, and that both POS tags and lemmata are present in the output.

हो	VVYN1	हुनु
,	YM	PUNC
अत्यन्त	RR	अत्यन्त
दुःखद	JX	दुःखद
हुन्	V0	हुनु
छ	VVYN1	हुनु
परिचय	NN	परिचय
।	YF	PUNC

5.5 Evaluation

Corpus annotation systems are typically evaluated quantitatively, that is, by means of statistical measures of how closely their output approximates the desired output, usually measured on a token-by-token basis. It is far from clear that these are optimal measures of performance; as Abney (1997: 121) points out, in many cases we might be more interested in the proportion of *sentences* that are correctly tagged throughout – which will obviously be much lower. Moreover, particularly

for languages like Nepali where the tagger performs extensive tokenisation, a statistic that measures success in tokens is prone to being swayed one way or the other by errors at the tokenisation stage. However, token-based performance measures are the de facto standard form reported in the literature. Two pairs of measures are used: *precision* (percentage of correct analyses out of the total analyses produced) and *recall* (percentage of correct analyses out of the total number of tokens); and *accuracy* (percentage of tokens given the correct analysis) and *ambiguity* (mean number of analyses per token). The reason for the pairs of measures is to allow the performance of n-best annotation, where more than one analysis per token may be produced, to be quantified. As van Halteren (1999: 82) notes, the precision/recall pair is "geared towards the description of ambiguous taggings" and is commonly used in the context of research into information retrieval; accuracy/ambiguity are the more common measures in the corpus linguistic literature. When, as here, the software always produces exactly one analysis per token, ambiguity is always equal to 1 and accuracy is usually cited alone.

To give an example, the accuracy rate of our existing POS tagger, which as mentioned above is 93%, was determined by a standard procedure: of the available manually-tagged data, a 10% subset was held apart as test data, and the rest used as training data. The tagger's performance on an untagged version of the test data, compared to the gold-standard manually-tagged version, gives the accuracy rate. This is expressed as a percentage of tokens given the same tag by manual and automated taggers. Of course, the test and training data both represent edited, published written Nepali; for other types of text, or spoken data, a degradation in the accuracy rate is to be anticipated.

The obvious approach, therefore, would be to evaluate the lemmatiser according to the same procedure. However, this was not directly possible. Since the lemmatisation resources were not generated from training data in the form of a gold-standard annotated corpus, there was no test data on which to base such an evaluation. Manual lemmatisation of a large gold-standard dataset, which would be an extremely labour-intensive and lengthy process, was beyond the scope of our current development project. And while it is possible to estimate the textual coverage of the lemmatisation lexicon and rules, this would not tell us how often the application of lexicon and rules was *correct*. Given these problems, we resolved to undertake a restricted-scale test of the combination of tokeniser, POS tagger and lemmatiser by evaluating its performance on a small sample of running text. Of course, the performance of the lemmatiser cannot be assessed independently of that of the tokeniser and tagger, as the output of the tagger is the input to the lemmatiser.

The test data were drawn from the NNC Core Sample. Three samples of approximately 2,000 words each (prior to tokenisation) were selected, for a total of 475 sentences. Each sample represented a different broad genre: one fiction, one literary non-fiction, one non-literary non-fiction; all were texts that had not previously been analysed manually in the process of training the POS tagger. Each token of this sample was manually checked to determine whether or not it had been accurately analysed. All errors of whatever kind were recorded. The sample data totalled 8,763 tokens after retokenisation. The figures for the different kinds of errors, and their combinations, are shown in Table 2.

Error Type	Count	Percentage
All tokens	8763	100.0
Error in tokenisation only	7	< 0.1
Error in POS tagging only	76	0.9
Error in lemmatisation only	255	2.9
Error in tokenisation, POS and lemmatisation	108	1.2
Error in POS tagging and lemmatisation	69	0.8
Total tokens with any error	*515*	*5.9*

Table 2. Error rates in manually-checked sample data.

The overall accuracy rate of 94.1% for the annotations as a whole is higher than the 93% measured for the POS tagger alone *prior* to the new additions to the tokenisation; this is probably because many of the newly separated elements are highly frequent auxiliary verbs whose tagging (and lemmatisation) is straightforward (such as *cha* "is"). The accuracy of the lemmatiser is 95.1%, but if we try to separate the effects of the lemmatiser from the other components by considering *only* tokens where the input to the lemmatiser did not contain an error, then the lemmatiser's accuracy rate becomes 97%. So as we would have predicted, errors in tokenisation or tagging tend to cascade through and result in erroneous lemmatisation as well. Moreover, it was noticeable that many of the errors involved the same words being tagged incorrectly every time they occur; this suggests that, with relatively little additional effort, the error rate could be reduced further by addressing these recurrent problems directly. In sum, then, although a small-scale test like this can never be relied on to produce wholly precise accuracy statistics, a performance in the region of 94-95% is roughly in keeping with the success rates most commonly reported in the literature for token-level corpus annotation.

6 Applications for the lemmatised NNC

The resources described in the preceding sections have been applied to the NNC to produce a "second version" of the data, which is the version currently made available to researchers working with the corpus. The lemmatisation, and the extended tokenisation on which it depends, have numerous applications. At the most basic level, searches in the corpus are now much more flexible and comprehensive. For example, the use of lemmatisation to link variant spellings with their standard forms allows a complete set of instances to be retrieved by searching for the lemma – rather than searching for each variant form separately or using a complex regular-expression query, as would previously have been necessary. Similarly, many kinds of analyses can be undertaken at a more abstract level using the lemmatisation. For instance, searching for forms of the progressive aspect (such as *garirahancha* "(he/she) is doing") would have required all possible forms of the auxiliary *hunu* "be" to be spelt out explicitly (in *garirahancha*, *cha* is the form of *hunu*). With the lemmatised corpus, search patterns can be specified in terms of lemmata, which is much more efficient. For example, using the CEQL[13] query language in the CQPweb software, together with POS tags and lemmata, forms of the progressive such as *garirahanchu, garirahanthyo* and so on can be retrieved

13 Common Elementary Query Language, created by Stefan Evert.

with this query:

```
_V* {rahanu} {hunu}
```

(with, of course, Devanagari strings for the lemmata; note that in CEQL syntax, {braces} indicate a lemma query and _underscore indicates a POS tag query, so this query searches for "any verb, followed by a form of *rahanu,* followed by a form of *hunu*").

Such abstracted searches are a tool of convenience, but could in theory be accomplished without lemmatisation, albeit with some effort. However, some analyses absolutely depend on the lemmatisation. An extension of Hardie's (2008) work on collocational patterns around Nepali postpositions looks at patterns around auxiliary verbs.[14] For corpus software to generate correct collocation statistics for a verb such as *hunu,* it needs to be able to identify, group and quantify the contexts of *all* the different forms that that verb might take. The lemmatisation is indispensible for this purpose.

7 Conclusion

In this paper, we have illustrated how lemmatisation of a language such as Nepali, which contains many morphological challenges for automated analysis, can be approached. In particular, we have focused on the linguistic knowledge resources required by the tokenisation and lemmatisation modules of the *Unitag* software – namely tokenisation rules and lists of exceptions, a lemmatisation lexicon, and lemmatisation rules. Such a combination of rules on the one hand, and specific wordlists on the other, is liable to be required by any such system, although of course the precise format of these resources will depend on the nature of the software in question. We have illustrated an approach to resource creation that is partly automated, by using corpus-derived frequency lists and the output of analysis programs to generate the lists of items on which the analyst bases the resources. This approach yields resources with maximised (though not complete) coverage of the language. It is our hope that the approach to tokenisation and lemmatisation, and associated linguistic knowledge resources, that we have presented here may be illustrative and of use to researchers looking at other languages of the Himalayan region, most especially those that have similar morphological behaviour to Nepali.

Appendix

The linguistic resources used by the tokeniser and lemmatiser are provided as supplementary materials to this paper:

- The tokenisation rule-list (see 4 above), *n-t-rules-ext.txt*
- The tokenisation exceptions lexicon (see 4 above), *n-t-exceptions-ext.txt*
- The lemmatisation lexicon (see 5.3 above), *n-lemmalex.txt*
- The lemmatisation rules (see 5.4 above), *n-lemmarules.txt*

The Nepali part-of-speech tagging software is available separately at www.lancs.ac.uk/staff/hardiea/nepali/postag.php.

14 The work on postpositions was funded by the UK Arts and Humanities Research Council (AHRC).

References

Abney, Steven. 1997. "Part-of-speech tagging and partial parsing". In: Young, Steve; and Bloothooft, Gerrit (eds.) *Corpus-based methods in language and speech processing,* 118-136. Dordrecht: Kluwer Academic Publishers.

CBS. 2001. *Population of Nepal.* Kathmandu: National Planning Commission.

El-Beze, Marc; and Merialdo, Bernard. 1999. "Hidden Markov models". In: van Halteren, Hans (ed.) *Syntactic wordclass tagging,* 263-284. Dordrecht: Kluwer Academic Publishers.

van Halteren, Hans. 1999. "Performance of taggers". In van Halteren, Hans (ed.) *Syntactic wordclass tagging,* 81-94. Dordrecht: Kluwer Academic Publishers.

Hardie, Andrew. 2004. *The computational analysis of morphosyntactic categories in Urdu*. Unpublished PhD thesis, Department of Linguistics, Lancaster University. http://eprints.lancs.ac.uk/106/

Hardie, Andrew. 2005. "Automated part-of-speech analysis of Urdu: conceptual and technical issues". In: Yadava, Yogendra P.; Bhattarai, Govinda; Lohani, Ram Raj; Prasain, Balaram; and Parajuli, Krishna (eds.) *Contemporary issues in Nepalese linguistics,* 48-72. Kathmandu: Linguistic Society of Nepal.

Hardie, Andrew. 2008. "A collocation-based approach to Nepali postpositions". *Corpus Linguistics and Linguistic Theory* 4(1): 19-62.

Hardie, Andrew. Forthcoming. "CQPweb – combining power, flexibility and usability in a corpus analysis tool".

Hardie, Andrew; Lohani, Ram Raj; Regmi, Bhim N.; and Yadava, Yogendra P. 2009. " A morphosyntactic categorisation scheme for the automated analysis of Nepali". In: Singh, Rajendra (ed.) *Annual Review of South Asian Languages and Linguistics 2009,* 171-198. Mouton de Gruyter.

Hardie, Andrew; and Mudraya, Olga. 2009. "Collocational patterning in cross-linguistic perspective: adpositions in English, Nepali, and Russian". *Arena Romanistica* 4: 138-149.

Hundt, Marianne; Sand, Andrea; and Skandera, Paul. 1999. *Manual of Information to accompany The Freiburg-Brown Corpus of American English ('Frown').* Englisches Seminar, Albert-Ludwigs-Universität Freiburg. http://khnt.hit.uib.no/icame/manuals/frown/index.htm .

Hundt, Marianne; Sand, Andrea; and Siemund, Rainer. 1998. *Manual of information to accompany the Freiburg-LOB Corpus of British English ('FLOB').* Englisches Seminar, Albert-Ludwigs-Universität Freiburg. http://khnt.hit.uib.no/icame/manuals/flob/index.htm .

Leech, Geoffrey. 2007. "New resources, or just better old ones? The Holy Grail of representativeness". In: Hundt, Marianne; Nesselhauf, Nadja; and Biewer, Carolin (eds.) *Corpus Linguistics and the Web,* pp. 133-149. Rodopi, Amsterdam.

McEnery, Tony; and Hardie, Andrew. 2011, in press. *Corpus linguistics: Method, theory and practice.* Cambridge: Cambridge University Press.

McEnery, Tony; and Xiao, Zhonghua. 2004. "The Lancaster Corpus of Mandarin Chinese: A corpus for monolingual and contrastive language study". In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC) 2004,* Lisbon, May 24-30, 2004, pp. 1175-1178.

Voutilainen, Atro. 1999. "A short history of tagging". In: van Halteren, Hans (ed.) (1999) *Syntactic wordclass tagging,* 9-22. Dordrecht: Kluwer Academic Publishers.

Yadava, Yogendra P.; Hardie, Andrew; Lohani Ram Raj; Regmi Bhim N.; Gurung, Srishtee; Gurung, Amar; McEnery, Tony; Allwood, Jens; and Hall, Pat. 2008. "Construction and annotation of a corpus of contemporary Nepali". *Corpora* 3(2): 213-225.

Andrew Hardie
a.hardie@lancaster.ac.uk

Ram R. Lohani
ramlohani@gmail.com

Yogendra P. Yadava
ypyadava@gmail.com

SUPPLEMENTARY MATERIAL

The following supplementary material is available online at:
http://www.linguistics.ucsb.edu/HimalayanLinguistics/articles/2011/HLJ1001H.html

- n-lemmalex.pdf [106 pp.]
- n-lemmarules.pdf [5 pp.]
- n-t-exceptions-ext.pdf [53 pp.]
- n-t-rules-ext.pdf [66 pp.]