Shallow Parsing for Nepal Bhasa Complement Clauses

Borui Zhang
George A. Smathers Libraries
University of Florida
boruizhang@ufl.edu

Abe Kazemzadeh
Graduate Programs in Software
University of St. Thomas
abe.kazemzadeh@stthomas.edu

Brian Reese
Institute of Linguistics
University of Minnesota
breese@umn.edu

Abstract

Accelerating the process of data collection, annotation, and analysis is an urgent need for linguistic fieldwork and documentation of endangered languages (Bird, 2009). Our experiments describe how we maximize the quality for the Nepal Bhasa syntactic complement structure chunking model. Native speaker language consultants were trained to annotate a minimally selected raw data set (Suárez et al., 2019). Embedded clauses, matrix verbs, and embedded verbs were annotated. We apply both statistical training algorithms and transfer learning in our training, including Naive Bayes, MaxEnt, and fine-tuning the pre-trained mBERT model (Devlin et al., 2018). We show that with limited annotated data, the model is already sufficient for the task[1] . The modeling resources we used are largely available for many other endangered languages. The practice is easy to duplicate for training a shallow parser for other endangered languages.

1 Introduction

Nepal Bhasa (also known as Newari or Newar Language) is an endangered, low-resource language mainly spoken by the indigenous community in Kathmandu Valley, Nepal. The native speaker population has declined from 1,041,090 (Shrestha, 1999) to 860,000 from 1991 to 2011. The current project comprises two interconnected goals. First, we aim to better understand Nepal Bhasa complementation structures in order to explore broader cross-linguistic generalizations. Second, we leverage the available low cost resources and our expertise in language, linguistics, and data science, to build complementation prediction models to facilitate our research on Nepal Bhasa complementation.

2 Linguistic motivation

Every natural language has complementation structures where a clause is embedded within a larger clausal constituent by a clause embedding predicate. Clausal syntactic structure and lexical semantics of clause embedding verbs, therefore, are two major focuses in research on complementation (Moulton, 2009; Bresnan, 1972). We study the Kathmandu Nepal Bhasa dialect (cf. Genetti (2009) for Dolakha Newar) which is generally head-final (OV language). However, embedded complement clauses include both head-final and head-initial complementizers, as shown in (1) and (2) respectively.[2]

(1)	Sitā-na	[CP	Rām-na	oṃ	nala
	sitaana		ramna	ong	nala
	Sita-ERG		Ram-ERG	mango	eat.PST

dhakā/dhayā]	dhā-u.
dhakaa/dhayaa	dhaau
C/C	say-PST

‘Sita said **that** Ram ate mangos.’

(2)	Sitā-na	dhā-u	[CP	**ki**	Rām-na
	sitaana	dhaau		ki	ramna
	Sita-ERG	say-PST		C	Ram-ERG

oṃ	nala].
ong	nala
mango	eat.PST

‘Sita said **that** Ram ate mangos.’

[1] The data and code used in this paper are available at github.com/boruizhang/newa

[2] We follow Leipzig glossing conventions: ERG for ergative, PST for past tense, and C for complementizer.

Proceedings of the Fifth Workshop on the Use of Computational Methods in the Study of Endangered Languages, pages 61 - 67
May 26-27, 2022 ©2022 Association for Computational Linguistics

Kathmandu Nepal Bhasa has four surface forms of complementizers: two head-final *dhakā*/*dhayā*, head-initial *ki*, and null head (Zhang, 2021). In studying clausal complementation (CP) in this language, one would ideally want access to as much language data as possible exemplifying the relevant CP embedding structures. However, in our experience the speed of data collection and the vocabulary range are often limiting factors in traditional linguistic fieldwork. These limitations negatively impact the development of research on endangered languages given the lack of quantitative evidence to test and confirm theoretical claims. Zhang (2021) suggests that different complementizers in Nepal Bhasa contribute different syntactic and semantic properties to a CP even though the surface forms are seemingly interchangeable based on the general meaning of the sentence as in the examples shown in (1) and (2). The data from the first author's Nepal Bhasa fieldwork shows potential morphological restrictions on matrix verbs that may be related to complementizer selections.

Large, structured corpora collected from non-fieldwork study can help validate theoretical linguistic claims. Such corpora provide a wider range of verb forms and more realistic distributions of complementizer uses. Clause-level structural information can be used to retrieve embedded CP constituents (e.g. Universal Dependencies relations or labeled constituent structure parse trees as in the UPenn Treebank (McDonald et al., 2013; Marcus et al., 1993)). However no such resources exist for Nepal Bhasa. Building structured corpora is costly and time-consuming for small research groups. Therefore, the current research investigating Nepal Bhasa CPs is aimed to explore the extent to which NLP techniques can help with accelerating the process of linguistic fieldwork annotation.

3 Data and Annotation

Structured corpora provide naturalistic data with syntactic and semantic annotations and provide an important resource for linguistic research and language documentation (Hovy and Lavid, 2010; de Marneffe and Potts, 2017). Building annotated corpora for endangered languages is particularly beneficial, as linguistic insights are systematically shown in the data, which are directly reusable and can be improved by adjoined efforts over time. However, there is no annotated public corpus resource of Nepal Bhasa currently available. However, the Open Super-large Crawled Aggregated coRpus (OSCAR) (Suárez et al., 2019), a 20TB corpus covering 166 natural languages, does include 5.7MB (16694 sentences) of unannotated Nepal Bhasa data. Two native speaker consultants assisted in the annotation process for the project, providing their language expertise on identifying embedded clauses and verbs in a small, pre-selected set of sentences. We worked closely on reviewing annotation work to improve the annotation quality. A faster annotating work speed was observed in the later annotation sessions. We then used this annotated data to train shallow parsers to predict embedded clauses in Nepal Bhasa.

The study focuses on the CPs that are headed by *dhakā* (head-final) and *ki* (head-initial). Table 1 outlines the pre-processing steps undertaken before annotation, including removing non-Devanagari characters, aligning one sentence per line, and removing sentences that had less than three words in one line, resulting in a total of 16603 clean sentences left for CP extraction.[3] 684 sentences were found containing the keyword *dhakā*, and 2660 sentences were found that contained the keyword *ki*. *Dhakā* is a good morphological cue for the detection of complement sentences in the data, while *ki* is ambiguous between being a complementizer or a phrasal conjunction ('and'). Out of the 3344 sentences (680+2660) that potentially contain CPs, 200 *dhakā*-sentences and 100 *ki*-sentences were randomly selected for the annotation task. See Appendix **??** for the written annotation guidelines.

Among the 300 sentences, 6 true embedded CPs were found in the 100 *ki*-sentences, and more than 190 true embedded CPs were found in the 200 *dhakā* sentences. After the manual annotation, the annotated sentences were converted to the CoNLL-2003 shared NER task format using IBO labels (Abney, 1991), as shown in Table 2.

[3] Data will be made available in Github repo.

Devanagari script data	Sentences
OSCAR-2019 raw sentences	16694
# actual working sentences	16603
# Keyword *‘dhakā’*	684
# Keyword *‘ki’*	2660
Annotated	
# total manually annotated	300
# total identified non-embedded	101

Table 1: Nepal Bhasa OSCAR corpus status

Tags	Tokens
#I	2332
#O	1933
#B	208

Table 2: Annotation level distribution

Because chunks are by definition non-overlapping sequences of tokens, the models we present below are unable to recognize recursive structures (e.g., a CP embedded in another CP). We found few instances of such structures in the data and chose to label only the embedding CP in such instances.

4 Learning methods

We implemented three CP chunking models for Bhasa Nepal: Naive Bayes, maximum entropy, and mBERT via NERDA. For Naive Bayes and maximum entropy models, we utilized both bigram and trigram features (two and three word token contexts with one and two tag/label contexts, respectively) and we tried predicting labels in forward and reverse directions (using preceding and following n-gram contexts, respectively). For the labels used as features, we used predicted labels as the features when testing to prevent leakage of the true labels into the prediction.

We used the NERDA Python library (Kjeldgaard and Nielsen, 2021) to train a neural chunking model. The package offers an easy to use interface for the NER task with fine-tuning of pretrained large models for any low-resource language.

We fine-tuned the pretrained cased mBERT model ‘bert-base-multilingual-cased’ for our experiment (Devlin et al., 2018). Nepal Bhasa is reported as being included in the mBERT training process. The data is split into training, validation, and testing sets with a ratio of 7:2:1. The average training time is 3 to 4 minutes with GPU. The hyper-parameter settings 11 epochs, 10 warmups, 7 batches proved to be the best among different trials. A systematic observation is that a larger batch size, 10 for example in this case, significantly lowers the accuracy, which (Keskar et al., 2016) suggest in their study of deep learning structures.

5 Results and discussion

Figure 1 and Table 3 show the token accuracy and chunk precision, recall, and F1 scores. For all the metrics except recall, increasing the n-gram context from bigram to trigram improved the maximum entropy models, whereas the extra trigram context decreased the performance for naive Bayes across all metrics.

We hypothesized that reversing the order of processing would be beneficial for head-final languages like Nepal Bhasa, since the embedded clause appears before the main verb (order: S CP V) in the default position, in contrast to the head-initial word order (order: S V CP) of languages like English. However, the result did not show any benefit to performance, even decreasing the performance (these were omitted from Figure 1 but shown in Table 3). This suggests that using right-to-left processing only may not be an appropriate learning order regardless of the headedness of a natural language. Even the head-final CPs as in (1) show a ‘backward’ relation between the main verb and the complementizer head of the embedded clause; the remaining sentence elements do not maintain backward relations for every pair.

The mBERT-based NERDA models show comparable performance to the maximum entropy models with trigram features. The NERDA model showed improved accuracy (96% vs. 91% but slightly lower F-score (69% vs. 72%), with NERDA showing higher recall than maximum entropy (77% vs. 69%) but lower precision (63% vs. 75%). We manually reviewed the predictions for every entry in the test set. The NERDA fune-tuning models predict the right edge of the CP with 100% accuracy, and the left-edge with 69% accuracy. The prediction accuracy result matches the language typological feature of

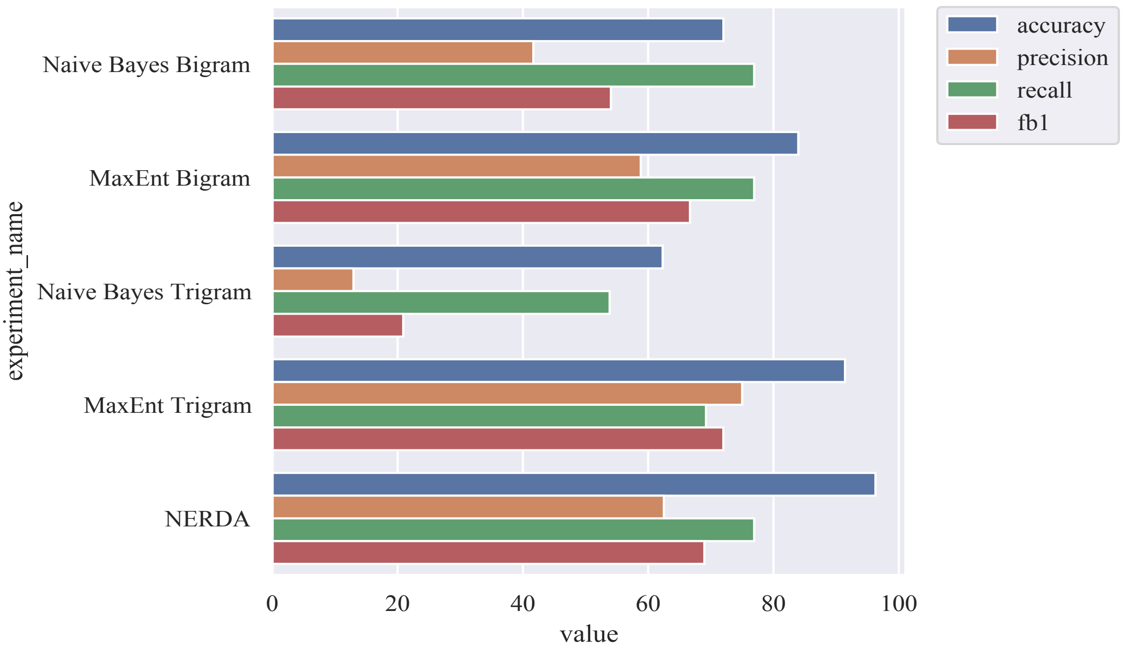

Figure 1: Tag-level accuracy and chunk-level precision, recall and F1 score for top-performing models. Table 3 contains more details of other models.

experiment_name	toks	phrases	corr.	acc.	prec.	rec.	fb1
Naive Bayes Unigram	268	13	1	63.81	2.17	7.69	3.39
MaxEnt Unigram	268	13	0	52.24	0.00	0.00	0.00
Naive Bayes Bigram	268	13	**10**	72.01	41.67	**76.92**	54.05
MaxEnt Bigram	268	13	**10**	83.96	58.82	**76.92**	66.67
Naive Bayes Trigram	268	13	7	62.31	12.96	53.85	20.90
MaxEnt Trigram	268	13	9	91.42	**75.00**	69.23	**72.00**
Naive Bayes Bigram Backward	268	13	1	62.69	7.69	7.69	7.69
MaxEnt Bigram Backward	268	13	6	90.67	40.00	46.15	42.86
Naive Bayes Trigram Backward	268	13	0	56.34	0.00	0.00	0.00
MaxEnt Trigram Backward	268	13	0	60.07	0.00	0.00	0.00
NERDA	242	13	**10**	**96.28**	62.50	**76.92**	68.97

Table 3: Experimental results: number of tokens (toks), number of chunks (phrases), number of correct chunks (corr.), token accuracy (acc.), chunk precision (prec.), chunk recall (rec.), and chunk f-score (fb1).

Nepal Bhasa being head-final. As previously discussed, head-final complementizers syntactically appear on the right periphery of the clause, before the main verb, and therefore the right edge of the clause is more predictable due tothis strong linguistic cue. This could also show the benefit of the mBERT model's bidirectional transformer, which is expected to be good at capturing both head-final CP and other components with different headedness.

In contrast, the difficulty in predicting left CP boundaries may reflect corpus distributional facts. First, the head-initial complementizers are more rarely used in the data set than the head-final ones, even though both kinds are grammatical in this language. Second, other linguistic components, such as noun phrases and adverbials, may occupy the left periphery of an embedded CP structure.

Considering the small size of the training data, the accuracy of the model heavily depends on the training data, as shown by the 10-fold cross validation results in Table 4.

Additionally, we provide counts for matrix (embedding) verbs for the entire annotated data set. (See full list in Appendix B.) Our ongoing linguistic fieldwork data suggests a morphological restriction in Nepal Bhasa matrix verbs in complementation constructions: aspectual suffix morpheme नं (a, IPA:[ə]), never appears on embedding verb. The matrix verb distribution shows that the morpheme गु

fold	toks	phrases	corr.	acc.	prec.	rec.	fb1
1	383	16	0	48.56	0	0	0
2	320	16	13	92.50	76.47	81.25	78.79
3	417	16	11	85.37	64.71	68.75	66.67
4	418	16	0	41.15	0	0	0
5	388	16	0	61.86	0	0	0
6	333	17	2	81.08	9.52	11.76	10.53
7	331	19	5	82.78	25	26.32	25.64
8	275	16	11	86.91	64.71	68.75	66.67
9	281	16	15	97.51	93.75	93.75	93.75
10	285	16	11	94.39	55	68.75	61.11
mean	343.1	16.4	6.80	77.21	38.92	41.93	40.32
std	52.3	0.92	5.69	18.72	34.07	35.80	34.84

Table 4: 10-fold cross validation of NERDA model

(u, IPA:[u]), frequently appears in embedding verbs in the corpus which supports the generalization.

6 Conclusion

Our experiments training shallow parsers for Nepal Bhasa complement phrases has shown the potential use of NLP tools in assisting corpus annotation for fieldwork research in endangered languages in general. We successfully achieve some high model performance with the very limited data source (less than 300 manually annotated sentences, 2% of the entire OSCAR Nepal corpus). The procedure may be used as a starting step in developing more structured corpora for fieldworkers.

Furthermore, theoretical linguistic insights also suggest a new perspective to interpret the model performance. For example, we learned that the right boundary of the clause is more predictable than the left boundary of a clause for head-final CPs. This means model performance with the traditional 'IBO' annotation style could show a lower performance than one with an annotation style of 'IEO' ('E': end of the CP clause) for being the exact same model. Therefore, the directions for improving our chunking model performance should not only be seeking higher label accuracy, but also maintaining good linguistic understanding of the language.

Possible future directions can further improve this work. Studies show that annotator expertise has a strong influence on the annotation accuracy and speed (Baldridge and Palmer, 2009). Our language consultants' expertise has grown significantly throughout the experiment. Setting up agreement tests for annotators to review others' annotation work may be helpful to improve future accuracy, although the annotation time might be prolonged and more annotators would be needed. The deep learning NERDA model shows that transfer learning with fine-tuning pre-trained large language model is a promising methodology for low-resource linguistic fieldwork research. However, certain longer sentences were discarded by the training algorithm. Moreover, training models remain independent which makes them easy to share with other fieldworkers, and possibly to combine models to start building more complex structured treebank corpora for low-resource languages.

In additional to transfer learning, active learning featured with actively querying annotators for labels, can provide sufficient information to the annotators without being overwhelmed by a mass of data. More high quality training data can be provided under the productive pipe-line.

Acknowledgements

We thank our Nepal Bhasa native speaker consultants for their time and efforts with providing us the annotation help.

References

Steven P Abney. 1991. Parsing by chunks. In *Principle-based parsing*, pages 257–278. Springer.

Jason Baldridge and Alexis Palmer. 2009. How well does active learning actually work? time-based evaluation of cost-reduction strategies for language documentation. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 296–305.

Steven Bird. 2009. Natural language processing and linguistic fieldwork. *Computational linguistics*, 35(3):469–474.

Joan W Bresnan. 1972. *Theory of complementation in English syntax.* Ph.D. thesis, Massachusetts Institute of Technology.

Marie-Catherine de Marneffe and Christopher Potts. 2017. Developing linguistic theories using annotated corpora. In *Handbook of Linguistic Annotation*, pages 411–438. Springer.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Carol Genetti. 2009. *A grammar of Dolakha Newar*, volume 40. Walter de Gruyter.

Eduard Hovy and Julia Lavid. 2010. Towards a 'science' of corpus annotation: a new methodological challenge for corpus linguistics. *International journal of translation*, 22(1):13–36.

Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. 2016. On large-batch training for deep learning: Generalization gap and sharp minima. *arXiv preprint arXiv:1609.04836.*

Lars Kjeldgaard and Lukas Nielsen. 2021. Nerda. GitHub.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Ryan McDonald, Joakim Nivre, Yvonne Quirmbach-Brundage, Yoav Goldberg, Dipanjan Das, Kuzman Ganchev, Keith Hall, Slav Petrov, Hao Zhang, Oscar Täckström, et al. 2013. Universal dependency annotation for multilingual parsing. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 92–97.

Keir Moulton. 2009. *Natural selection and the syntax of clausal complementation.* University of Massachusetts Amherst.

Bal Gopal Shrestha. 1999. The newars: The indigenous population of the kathmandu valley in the modern state of nepal. *The Journal of Newar Studies*, 2:1.

Pedro Javier Ortiz Suárez, Benoît Sagot, and Laurent Romary. 2019. Asynchronous pipeline for processing huge corpora on medium to low resource infrastructures. In *7th Workshop on the Challenges in the Management of Large Corpora (CMLC-7).* Leibniz-Institut für Deutsche Sprache.

Borui Zhang. 2021. *Clausal Complementation in Nepal Bhasa.* Ph.D. thesis, University of Minnesota.

A Nepal Bhasa complement CP annotation guideline

Please follow the three steps to annotate the sentences in this corpus:
(i) If you find a sentence that has an embedded clause, mark the clause by adding a squared bracket '[ ]' around it.
(ii) Select and add the matrix verb of the sentence to a new line.
(iii) Select and add the embedded verb next to the matrix verb.

If the sentence does not have an embedded clause, add '**' in front of the sentence. If you cannot identify which verb to select, please fill it with a 'UNK' label in the of (ii) and (iii).

Embedded clause annotation example:

(3) स्कुलय् ब्वनेगु इलय् धाःगु खः [कि
Skul bwonegu yilaye dhagukha ki
In-school studying time it's-said [that
छुं नं वस्तुया रंग दइमखु]
chunah bastuya ranga daimakhu
any item color does.not.have
'It's said during school time that all items do not have colors.'
Matrix verb: धाःगु ख (dhagukha)
Embedded verb: दइमखु (daimakhu)

B Nepal Bhasa matrix verb list in the annotation set

Table 5 shows the embedding verbs seen in the corpus.

Matrix verb	Meaning	Count
म्हसीकिगु (mhasiku)	introduce	6
वयाच्वंगु दु (bayachogu)	become	5
न्यनेगु (nyanegu)	ask	4
धाइ (dhai)	say	4
उल्लेख यानातःगु दु (ulekh yanatagudu)	describe	3
थुइकेगु (thuikegu)	understand	3
सुचुकेत (suchuketa)	hide	3
तःगु (tagu)	put	3
खनेदु (khanaedu)	see	2
बियातःगु दु (biyatagudu)	give	2
क्यनेगु (kyanegu)	see	2
धयातःगु दु (dhayatagudu)	say	2
ब्वइ (woi)	show	2
सल्लाह बी (sallaha bi)	advice	2
नियन्त्रणय् कयाः (Niyantran kaya)	take charge	2
जुयाच्वंगु (juyachogu)	happen	2
तायेकाच्वंगु (tayekachogu)	keep	2
धयातःगु (dhayatagu)	say	2
तगु खः (tagu kha)	put	2
यानातःगु (yanatagu)	do	2
कनेगु (kanegu)	make to say	1
बिउगु दु (biyougudu)	give	1
दयेकूगु (dayekugu)	give	1
ज्वनेत (jyoneta)	catch	1
धारणा प्वंकेगु (daharana pwakegu)	pour thoughts	1
यानादीगु दु (yanadigudu)	done	1
पिकयादीगु (pikayadingu)	publish	1

Table 5: Embedding verb distribution