Named Entity Recognition for Nepali: Data Sets and Algorithms

Nobal Niraula
NowaLab
Madison, Alabama, USA
nobal@nowalab.com

Jeevan Chapagain
The University of Memphis
Memphis, Tennessee, USA
jchpgain@memphis.edu

Abstract

Named Entity Recognition (NER) task involves locating Named Entities (NEs) in free text and classifying them into predefined categories such as Person Name, Location and Organization. Although the NER task has been studied widely in resource-rich languages, it has not been studied thoroughly for Nepali, a resource-poor language. In this paper, we present the systematic study of NER for Nepali language with clear Annotation Guidelines obtaining high inter-annotator agreements. The annotation produces EverestNER, the largest human annotated NER data set for Nepali which has 24,587 entities in total. It has 308,353 tokens corresponding to 15,798 sentences which are annotated into five categories: Person, Location, Organization, Date and Event. We split the EverestNER data set into EverestNER-train and EverestNER-test. These standard data sets, therefore, become the first benchmark data sets for evaluating Nepali NER systems. We release the EverestNER benchmark data sets to facilitate the research in Nepali language

1 https://github.com/nowalab/everest-ner

NER is a fundamental task in NLP. It can be used stand-alone for mining knowledge from text. It also enables several downstream applications such as Information Retrieval (Guo et al. 2009), Question Answering (Molla, Van Zaanen, and Smith 2006), Machine Translation (Babych and Hartley 2003), Summarization (Li et al. 2020), etc. Thus, research in NER is a widely studied topic in resource-rich languages such as English and German (Sang and De Meulder 2003; Li et al. 2020). However, NER in resource-poor language Nepali, the main focus of this paper, has not been studied thoroughly. Few initial efforts have been made but they are limited by the coverage of named entities, requirements for hand crafted rules, and lack of standard benchmark data sets (Bam and Shahi 2014; Dey, Paul, and Purkayastha 2014; Singh, Padia, and Joshi 2019).

Table 1: A sample output of our NER model which extracted Event (EVT), Person (PER), Date (DAT), Organization (ORG) and Location (LOC).

<table>
<thead>
<tr>
<th>Token</th>
<th>Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>सशҘ œहरӆ  غالبवस</td>
<td>B-EVT</td>
</tr>
<tr>
<td>ओलӄ</td>
<td>I-EVT</td>
</tr>
<tr>
<td>आज</td>
<td>B-DAT</td>
</tr>
<tr>
<td>सशҘ œहरӆ</td>
<td>B-ORG</td>
</tr>
<tr>
<td>हलचोक</td>
<td>B-LOC</td>
</tr>
<tr>
<td>गर्दै</td>
<td>O</td>
</tr>
</tbody>
</table>

NER is a widely studied topics for a long time. Li et al. (Li et al. 2020) provide an excellent survey on the NER covering its history, approaches, resources and challenges. Initial methods relied on hand-crafted rules or supervised machine learning methods (e.g. Support Vector Machine (SVM) (Hearst et al. 1998), and Conditional Random Field (CRF)
(Lafferty, McCallum, and Pereira 2001)) that required feature engineering. CRF is the most commonly used approach that obtained state-of-the-art performance on several benchmark data sets (Li et al. 2020). Recent deep learning based approaches automatically learn features directly from raw input. Learning features using Bidirectional Long Short-Term Memory (BLSTM) for CRF layer at the top to decode tags yielded state-of-the-art results in both generic as well as domain-specific entity extraction tasks (Huang, Xu, and Yu 2015; Ma and Hovy 2016; Niraula, Kao, and Whyatt 2020; Wunnava et al. 2018). Such settings allowed training NER systems in end-to-end fashion, directly from the raw text. Many recent approaches leverage contextualized embeddings such as BERT (Devlin et al. 2018a) and ElMo (Peters et al. 2018). Multilingual BERT models now include Nepali (Devlin et al. 2018b) & (Conneau et al. 2019). However, these models have not been tested for NER tasks in Nepali.

In this paper, we present the first systematic study of NER for Nepali language in news articles. Our main contributions include:

• **Annotation Guideline:** We develop a clear annotation guideline for consistently marking named entities and obtaining high quality human annotated corpus.

• **Coverage:** We consider five named entities, the largest coverage of NEs for Nepali to date: Person, Location, Organization, Event and Date.

• **Benchmark Data Sets:** We release the first standard benchmark data set, EverestNER, to develop and compare Nepali NER systems at our GitHub address: https://github.com/nowalab/everest-ner. We provide human annotations both in character and word levels so either type of NER systems can be experimented using these data sets.

• **End-to-end NER Models:** We evaluate the state-of-the-art end-to-end BLSTM and BERT-based NER systems using the EverestNER data sets.

**Related Work**

There are very few NER works available in literature for Nepali (Bam and Shahi 2014; Dey, Paul, and Purkayastha 2014; Singh, Padia, and Joshi 2019). The first one is by Bam and Shahi (Bam and Shahi 2014) that utilized SVM to detect Person, Organization, Location and Misc categories. They used word features as well as gazetteers including person, organization, location, middle name, verb, designation and others. The SVM model was trained in one vs rest classification setting. This first known model for NER in Nepali has a number of shortcomings: (a) it is not clear how the authors generated the training data set. The quality of the training data set is not reported either, (b) their work did not take context information into account while training the model, and (c) computing features like this hinders the scalability of the NER work.

The second work is by Dey and Purkayastha (Dey, Paul, and Purkayastha 2014) who used Hidden Markov Model with n-gram technique for extracting POS tags. POS-tags with common noun, proper noun or combination of both are combined together, then uses a gazetteer list as a look-up table to identify the named entities. Results are reported for 750 sentences corresponding to Person, Location, Number, Organization, Currency, and Quantifier. This is a very primitive work which does not describe how the training examples are obtained and used. The proposed methodology is very vague.

The third work, and is the most recent one, is by Singh et al. (Singh, Padia, and Joshi 2019) who studied NER in Nepali is the closest work to our work. The authors first collected sentences from daily newspapers and annotated three types of entities: Person, Location and Organization. The authors then trained multiple neural models such as BiLSTM, BiLSTM-CNN, BiLSTM-CRF, and BiLSTM-CNN-CRF with different word embeddings and found that BiLSTM-CNN model performed the best. Although this work provides a data set (NepaliNER) publicly, the authors have not provided the annotation guideline nor conducted the human evaluation of the annotated corpus e.g. inter-rater reliability, making the data set less reliable. Furthermore, it does not provide separate train and test data sets, making it harder for other NER systems to be evaluated and compared against. In contrast, we have clear annotation guidelines, high inter-rater agreements, train and test data sets for benchmarking NER systems, and experiments with latest transformer models. Our EverestNER data set has 15,798 annotated sentences and is more than four times larger than their NepaliNER data set which has 3,603 annotated sentences. EverestNER contains more than twice as many entities (24,587 entities) as NepaliNER (11,183 entities). Furthermore, EverestNER covers 5 entity types compared to 3 entity types in NepaliNER.

**Corpus Preparation**

We considered news articles as they frequently use named entities. We crawled 1,000 January 2019 news articles from setopati.com, a popular news portal in Nepal. Four articles were discarded because they were very long compared to others. The articles cover different domains such as Politics, Sport, Art, Society, Economics, Blog and Literature.

**Data Preparation for Annotation**

We applied very basic cleaning on the crawled articles e.g. removing HTML tags etc. and extracted sentences from them. Sentence extraction is a straightforward task in Nepali as sentences typically end with an ending mark ‘।’ and other punctuation marks such as ? and !.

A typical approach in NER annotation is to tokenize a sentence first and assign labels to the tokens corresponding to the predefined categories. This is challenging for morphologically rich languages such as Nepali. Words have different word forms corresponding to different aspects such as number, gender, honor, and tense. In addition, different suffixes can attach to the same word creating diverse word forms. This creates a challenge for tokenization resulting in different outputs for different tokenization schemes. For this reason, we did not tokenize the sentences for the annotation task. We rather marked Named Entities at character level using
<table>
<thead>
<tr>
<th>NE</th>
<th>Guidelines</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>PER</strong></td>
<td>Proper names of people including first names, last names, individual or family names, fictional names and unique nicknames. Generational markers such as Jr. and IV are included. <strong>DO NOT MARK</strong> honorific titles such as titles (डा), relation names (आमा, बापा), pronouns (तू, ते, सही, जी), reflexive pronouns (आफ), name prefixes (जी, जीमान्, नी), and royal titles (राजा, रानी, युवराज) and Sir (सर)</td>
<td>(a) First names: e.g. वणकमल, नारायणकाजी (b) Family names: e.g. महरा, शाही (c) Generational markers: जीमान् and नी (d) Aliases, nicknames: e.g. प्रतिप, बाबा, परी (e) Combinations of 1-4: e.g. नारायणकाजी जीमान् (f) Fictional/mythological characters: e.g. रावण, चौराहा</td>
</tr>
<tr>
<td><strong>LOC</strong></td>
<td>All man-made structures and politically defined places like the names of countries, rivers, and railway stations are marked as LOC. <strong>DO NOT MARK</strong> a generic reference to a location or a nationality e.g. मन, समुद्र, अमेरिका, नेपाली</td>
<td>(a) Buildings: e.g. पौड़े घर, एपोलो अंतरिक्ष (b) Cities, towns, city districts: e.g. माईत्र, बसु, ललितपुर (c) Continents: e.g. एशिया (d) Countries, states: e.g. काठमाडौं, समुद्र (e) Geographical areas: e.g. मानहरु, चाईना (f) Parks: रारा राय खेत, गोदावरी (g) Planets, celestial objects: e.g. चन्द्रमा, गौतम (h) Seas, lakes, rivers: e.g. उम्रा, संगमरमर, गंगा</td>
</tr>
<tr>
<td><strong>ORG</strong></td>
<td>The name of a company, media group, team, political party or any other entity created by a group of people.</td>
<td>(a) Commercial companies: e.g. नेपाल टेलिकोम, गोल (b) Commissions: e.g. खनिज विभाग (c) Communities/groups of people: e.g. लिंग (d) Education &amp; scientific institutes: e.g. राष्ट्रीय अनुसंधान केन्द्र (e) Judicial systems: e.g. काठमाडौं जिला आदालत, सरकारी आदालत (f) Law enforcement organizations: e.g. अनुसंधान विभाग, नेपाली सेना (g) News agencies and stations: e.g. काठमाडौं ट्विटर (h) Political parties: e.g. नेपाली कांग्रेस (i) Public administration: e.g. प्रराष्ट्रीय विभाग, नौसेना विभाग (j) Sport leagues and clubs: e.g. आईसीएफ, नेपाल क्रिकेट संघ (k) Banks: e.g. सानिध्य बैंक, अमेरिकन बैंक</td>
</tr>
<tr>
<td><strong>EVT</strong></td>
<td>Named events and phenomena including natural disasters, hurricanes, revolutions, battles, wars, demonstrations, concerts, sports events, etc.</td>
<td>(a) Expos: e.g. बौद्ध धर्म, गोलियों एवं आग (b) Ex- plicitly marked events e.g. टेलिकोमको धार्मिक सादृश्य, स्वतंत्र विद्वान विषय (c) Sporting Leagues e.g. विश्वकप, सालिंग, एफ वान (d) Hurricanes e.g. चौराहा लड़ाई, माओवाद्धरु</td>
</tr>
<tr>
<td><strong>DAT</strong></td>
<td>Date or period of 24 hours or more, including day, week, month, certain named period, season, year, etc. Age is also included in this category whether it is a noun, adjective, or adverb phrase. Numerical values can be spelled out or expressed using digits.</td>
<td>(a) Full or partial date: १५ काठमाडौं २०७६, असार १५ (b) Duration: हजार वर्ष, माघ १२ दिन और १५ (c) Age: ३५ वर्ष, ३५ वर्ष का (d) Season: वसंत रीति, चित्रित (e) Day and month: आठवारा वैशाख</td>
</tr>
</tbody>
</table>

**Table 2: Annotation guideline for EverestNER data set**

BRAT Annotation Tool (Stenetorp et al. 2012). This allows us to experiment with different tokenization schemes as well as develop character-based algorithms for NER. A sample annotation at character level is shown in Figure 1.

![Character-level annotation](image)

**Figure 1:** Character-level annotation for Named Entities using BRAT tool: [(31, 37, ‘PER’, ‘झारी’), (40, 47, ‘LOC’, ‘अमेरिका’)]

**Annotation Targets and Process**

In this study, we considered five different Named Entity Targets namely person (PER), location (LOC), organization (ORG), event (EVT), and date (DAT). Most prior studies, as previously mentioned, only considered PER, LOC and ORG categories.

We adapted the annotation guideline prepared by Ruokolainen et al. for creating Finnish News Corpus for Named Entity Recognition (Ruokolainen et al. 2020). We summarized the annotation guidelines for our annotation targets with their definitions and examples in Table 2. Our full annotation guidelines include detailed examples, exceptions, cases and illustrations. It will be provided as part of the EverestNER corpus.

Once the annotation targets and initial guidelines are finalized, the annotators (the authors of this paper) reviewed and discussed them multiple times prior to the actual annotation. Before annotating records independently, the annotators were given common articles to mark the named entities. We obtained an inter-rater agreement of 0.74 using
Cohen’s Kappa (McHugh 2012), demonstrating substantial agreements between annotators.

**EverestNER Data Sets**

In total, the annotators labeled 28,281 sentences corresponding to 996 news articles to five annotation targets. Out of them, 12,820 (45.33%) sentences had at least one entity, confirming that news articles are rich in named entities.

To make the benchmark data sets, we applied 85-15 split procedure to the news articles using random selection and formed two buckets: one with 85% of articles and another with 15% of articles. We did not apply split procedure in sentence level because we don’t want to have sentences from the same article appear in train and test datasets. To form the data sets, from each bucket, we collected all sentences containing at least one entity and only 25% sentences without any entities. We did not include every sentence without any entity to lower the distribution of O-tags compared to named entity tags. This is on par with the popular Conll-2003 English NER data set where sentences without any entities constitute about 25% of the sentences with entities. This resulted in standard EverestNER-train and EverestNER-test data sets for evaluating Nepali NER systems.

As mentioned, annotations are made at character-level and the data set is readily applicable for developing character-based models. For word-based models, we tokenized the sentences using a list of common Nepali suffixes and assigned token labels based on the annotations. We provided the corpus statistics in Table 3. EverestNER-train contains 13,848 sentences, 268,741 tokens and 21,317 entities. EverestNER-test data set contains 1,950 sentences, 39,612 tokens and 3,270 entities. In total, EverestNER has 15,798 sentences and 24,587 entities. It has four times more annotated sentences and twice as many entities as the previously available data sets in Nepali NER (see the Related Work Section above). In addition, EverestNER contains five entity types and covers more entity types for Nepali to date.

**Named Entity Recognition Methods**

We performed experiments with different types of NER systems as follows:

**Baseline**: Our baseline model is a rule-based system. It collects entities and corresponding targets from the training data set and makes a lookup dictionary to map an entity token span to its target label. Since a token span can have multiple targets depending on the context (e.g. Nepal can be a Person, a Location or an Organization), it assigns the most popular target label to that span (e.g. Nepal -> Location). For prediction, if it finds the longest token span in input that can match exactly with a key in the lookup dictionary, it assigns the corresponding target label for the matched span.

**BERT-based Model**: We constructed a BERT-based NER model using NERDA library (Kjeldgaard and Nielsen 2021) which we call it as BERT-bbmu. It uses bert-base-multilingual-uncased (Devlin et al. 2018b), a multilingual BERT model trained with Wikipedia data on 102 languages including Nepali.

**BLSTM-CRF Models**: We configured different BLSTM-CRF model architectures using NCRF++ library (Yang and Zhang 2018). It can generate word features using Word and Character embeddings as well as external rules. The library can take pre-trained Word and Character embeddings or can itself learn them during training.

**Experimental Setup**

We trained the BERT-bbmu model for 10 epochs, with training batch size 10 and learning rate of 0.0001. For the BLSTM-CRF model, we supplied 300 dimension pre-trained fastText Word Embedding from NPVEC1 (Koirala and Niraula 2021) (processed_tokenized_fast_text) but let the network learn 30 dimensional Character Embeddings itself during training. It was trained for 50 epochs using 4 CNN layers, 0.5 dropout, 0.015 learning rate, SGD optimizer, and 50 batch size.

**Evaluation Criteria**

We used the micro-averaged precision, recall and F1 scores for evaluating NER models using seqeval python package (Nakayama 2018) which is compatible with the CoNLL shared-task evaluation scheme.
Results

We reported the model performances in Table 4. The baseline system obtained the F1 score of 0.62. This is expected as the baseline is simply a look-up system and cannot predict other entities than the entities it knows before. In addition, it cannot infer from the context to appropriately assign the target label. Compared to the baseline system, as expected, we obtained very high performances for BLSTM-CRF and BERT based models with F1 scores of 0.81 and 0.85, respectively. The best performing model was the BERT-bbmu model which obtained precision, recall and F1 scores as 0.87, 0.84 and 0.85 respectively. The contextual representation BERT learns for sentences is critical for NER tasks and is confirmed by our experiments as well. Note that we are the first one to explore the BERT-based NER models for Nepali. Our experiment confirms that we can greatly benefit even from the multilingual BERT for Nepali NER tasks.

To understand how the best performing NER model is performing in individual categories, we reported the precision, recall and F1 scores for individual categories in Table 5. We can see that the model is performing well for all entity types except for the Event category. The Event category was a minority category with just 312 mentions in the training data (see Table 3). All other entity types had over 3,000 mentions for each in the training data. This means the model either did not learn enough context for the Event category compared to the other categories or got affected by the class imbalance problem.

Error Analysis

We studied the mistakes made by the model compared to the gold tags. We observed some common patterns. First, mistakes in tokenization were one of the main reasons for the errors. For instance, ‘Suukyi’ in ‘आङ सान सुक्यी’ (Aung San Suukyi) was incorrectly tokenized into two tokens as सुक्यी (Suu Kyi) due to a common suffix of ‘(ki)’. Because of this, the model did not consider the Key as part of the person name: आङ/B-Person सान/1-Person सु/1-Person की/O. The suffix token की (Ki) overwhelmingly received the O tag compared to the PER tag in the training data set.

Second, it incorrectly assigned labels to the words with multiple senses. For example, instead of marking both tokens as person names in कल्पना खनाल (Kalpana Khanal), it only marked the second term Khanal which is a surname in Nepal: कल्पना/O खनाल/B-Person. This is most likely due to the term कल्पना (Kalpana) which has more than five senses including imagination, creation, thought, assumption and conjecture. Similarly, the model marked नेपाल (Nepal) as a LOC in मिस नेपाल (Miss Nepal) which is not correct as it is a pageant title.

Third, the model did not capture the complete entity span. In few cases, the model was able to discover the annotation errors such as incorrect target for an entity and missed entities.

Conclusions and Future Work

We presented the first systematic study of the Named Entity Recognition problem in Nepali. We started with developing the largest high quality human annotated corpus for training NER systems for Nepali using news articles. We then constructed the EverestNER (EverestNER-train and EverestNER-test) data set, the first benchmark data set for building and evaluating NER systems for Nepali. EverestNER contains 15,798 annotated sentences and 24,587 entities corresponding to five entity types: Person, Location, Organization, Event and Date. With our data sets, it is now possible to develop and compare different NER systems for Nepali. We have released these data sets publicly at GitHub: https://github.com/nowalab/everest-ner.

We developed the end-to-end NER neural models for Nepali using BLSTM-CRF and BERT-based architectures. We trained these models using the EverestNER-train data set and evaluated the models using the EverestNER-test data set. We confirmed that high performing neural NER models can be developed for Nepali using these data sets. To our knowledge, we are the first to experiment with the BERT-based models for NER in Nepali. We discovered that BERT-based NER models for Nepali perform very well even when we utilize the multilingual BERT.

Future work includes experimenting with (a) linguistic features such as POS tags in addition to the features automatically learned by the neural models, (b) different tokenization schemes, (c) character-based NER architectures, and (d) approaches to handle imbalance classes e.g. oversampling minority classes like the Event. Extending this work for Nested Named Entity Recognition is another future task to this end.

Acknowledgments

We would like to acknowledge and thank Nepali Shabdakosh (www.nepalishabdakosh.com) for providing the computation resource and Nepali word list for this project.

References


Li, J.; Sun, A.; Han, J.; and Li, C. 2020. A survey on deep learning for named entity recognition. IEEE Transactions on Knowledge and Data Engineering 34(1):50–70.


