DanfeNER - Named Entity Recognition in Nepali Tweets

Nobal Niraula
Nowa Lab
Madison, Alabama, USA
nobal@nowalab.com

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

Abstract

Twitter allows users to easily post tweets on any subject or event anytime, generating massive amounts of rich text content on diverse topics. Automated methods such as Named Entity Recognition (NER) are required to process the massive tweet data. Processing tweets, however, poses a special challenge as they are informal posts with incomplete context and often contain acronyms, hashtags, misspellings, abbreviations, and URLs due to length constraints. This paper presents the first systematic study of NER in Nepali tweets corresponding to five different entity types: Person Name (PER), Location (LOC), Organization (ORG), Date (DAT), and Event (EVT). We develop DanfeNER, the first human-labeled high-quality NER benchmark data sets for the low-resource language Nepali. DanfeNER contains 5,366 records and 3,463 entities in its train set and 2,301 records and 1,503 entities in its test set. Using this data set, we benchmark several state-of-the-art Nepali monolingual and multilingual transformer models, obtaining micro-averaged F_1 scores up to 81%.

Introduction

Social media platforms allow users to express and share their opinions in real-time on any topic, such as news stories, politics, movies, and events. Twitter, for example, lets users post short messages in the free-form text (tweets) on any topic. Tweets play an important role in modern society as ordinary people can share their voices directly with the public on trends such as natural disasters, sports, social injustice, government policy, and company product and service feedback. Therefore, analyzing and understanding tweets is important for several use cases, such as topic discovery and sentiment analysis. Twitter has millions of users. Analyzing the vast pool of tweets, they generate requires Natural Language Processing (NLP) techniques.

Tweets are rich in information with frequent mentions of named entities (NEs) such as Person Name, Location, Organization, and Event. Extracting NEs from text enables several downstream NLP applications such as question answering (Mollá, Van Zaanen, and Smith 2006), information retrieval (Guo et al. 2009), summarization (Li et al. 2020), machine translation (Babych and Hartley 2003), and target identification in offensive languages (Niraula, Dulal, and Koirala 2021). NEs are extracted from text using a Named Entity Recognition (NER) model, which assigns a predefined category to each token in a sentence. Standard NER categories include Person, Location, and Organization.

Nepali is a language in the Indo-Aryan family. It is the official language of Nepal and several eastern Indian regions, including Sikkim, Darjeeling, and Kalimpong. It is spoken by more than 20 million people, mainly in Nepal and other places worldwide, including Bhutan, India, and Myanmar (Niraula, Dulal, and Koirala 2022).

This paper presents a systematic study of extracting named entities from tweets written in the Nepali language. Although NER for Nepali has been studied in the past, it is focused mostly on extracting named entities in news articles (Niraula and Chapagain 2022; Singh, Padia, and Joshi 2019). News articles are long and contain formal language. In contrast, tweets are short and informal. Due to length constraints, tweets often contain acronyms, abbreviations, hashtags, misspellings, URLs, etc. As they are short, tweets often contain incomplete context. Due to these reasons, it is very challenging to process informal, incomplete, unstructured tweets. It is also shown that NER models developed for formal languages such as News articles do not perform well for tweets (Liu et al. 2011).

Table 1 shows sample Nepali tweets with their English translations. They are of different types, e.g., sarcastic comments (#1 and #2), news headlines (#3 and #4), and reactions to news and events (#5 and #6). Official Nepali is written in Devanagari script (#1, #2, #3, and #4), but due to ease of typing, they are transliterated using Roman letters (#5 and #6). We also observed code-switching tweets between languages, e.g., Nepali-English (#5) and Nepali-Hindi. Code-switching is common in a multilingual population where a speaker switches between languages within a single context (Lipski 1978). A code-switching data set for Nepali tweets has been developed by Maharjan et al. (Maharjan et al. 2015). Besides, as expected, Nepali tweets were very informal and ungrammatical. All these issues pose a special challenge in processing Nepali tweets, e.g., for the NER task.

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

- **Benchmark Data Sets**: We develop DanfeNER, the first standard benchmark data set, to train and evaluate NER

#	Original Tweet and Translation
1	पैसो जति एता रान अनि बैंक मा के को पैसा हुनु ? (*Money is here, that's why banks don't have money.*)
2	**लिङ्देन**^{PER} बाजे नि अब सकिय क्यारे (*Lingden is gone too, it seems.*)
3	बाँदरको समस्याले वैकल्पिक खेतीतर्फ **पाल्पा**^{LOC}का किसान - **kaligandaki Khabar**^{ORG} (*Farmers in Palpa are choosing alternative farming due to monkey problems - Kaligandaki Khabar.*)
4	**यु १९ महिला विश्वकप**^{EVNT} छनोट- **नेपाल**^{ORG}ले **भोलि**^{DATE} **कतार**^{ORG}विरुद्ध खेल्ने (*Nepal plays against Qatar tomorrow for U-19 Women Cricket World Cup.*)
5	Officially break up dherai jasto ko Nepali Cricket prati. (*Most people officially break up with Nepali Cricket*)
6	Kun geet ho yesto? Dai lai suhayo ta hai! :) :) (*Which song is that ? It is a good fit for you brother!*)

Table 1: Sample Nepali tweets with their translations. Named entities are in bold.

systems for Nepali tweets. We release it at our GitHub address [1]. We consider five types of named entities: Person, Location, Organization, Event, and Date. Our annotations are at character and word levels, so different NER systems can be trained with these data sets.

- **NER Models for Nepali Tweets**: We train and benchmark the state-of-the-art transformer models using the DanfeNER data sets. We compare them against the existing news NER models for NER in tweets.
- **Detailed Error Analysis**: We provide a detailed error analysis of NER models for Nepali tweets.

Related Work

NER in Nepali has been studied recently, mainly focusing on news articles but not much study has been done on tweets. One of the early studies used SVM to identify Person, Location, Organization, and Misc categories (Bam and Shahi 2014). It used word features including person, group, location, middle name, verb, designation, and others, as well as gazetteers. Dey et al. (Dey, Paul, and Purkayastha 2014) used Hidden Markov Model with n-gram for extracting POS tags which are then used together with a gazetteer list as a lookup table to identify the named entities. Singh et al. studied NER in Nepali news to discover Person, Location, and Organization using deep neural networks such as BiLSTM, BiLSTMCNN, BiLSTMCRF, and BiLSTMCNN CRF with different word embeddings (Singh, Padia, and Joshi 2019). The authors found BiLSTMCNN performed best among all the models. Most recently, Niraula and Chapagain conducted a detailed study of NER in Nepali news (Niraula and Chapagain 2022). They provided detailed annotation guidelines on marking entities and created EverestNER, a benchmark data set containing train and test data sets for news. The authors evaluated several Neural and transformer models based on EverestNER data sets. They showed that transformer models had state-of-the-art performances for Nepali NER.

The Hindi language is somewhat closer to Nepali. Here we list some works related to Hindi tweets. Vinay et al. proposed different machine learning models like CRF, LSTM, and Decision Tree to recognize the NE in Hindi-English mixed tweets (Sangma, Das, and Majumder). Their system used the word, character, and lexical features to feed onto the models. Their entities are limited to Person, Location, Organization, and Other. Proper annotation guidelines were also not provided for annotating entities. Kushara et al. (Singh, Sen, and Kumaraguru 2018) proposed a method based on hand-crafted features which were passed on to CRF and LSTM RNN to identify the NE in Hindi-English mixed tweets and are only focused on three entities, namely Person, Location, and Organization. The model proposed by Nut et al. (Limsopatham and Collier 2016) for recognizing NE in tweets used the BLSTM model, which could learn the orthographic features automatically without requiring any feature engineering. They used Twitter NER shared task data sets that don't include Nepali tweets.

Finin et al. (Finin et al. 2010) used the Conditional Random Field (CRF) model to evaluate the NER in tweets where data was annotated by Amazon Mechanical Turk and CrowdFlower workers. Their work focused on 3 basic entities: Person, Organization, and Location, limiting coverage for other entities. Wintaka et al. used different machine learning and deep learning methods like CRF and Bidirectional Long Short Term Memory (BLSTM) to recognize named entities (NE) on Indonesian tweets, which contain only 600 tweets, and the entities were also limited to Person, Location, and Organization (Wintaka, Bijaksana, and Asror 2019).

Our DanfeNER data sets contain 7,667 annotated tweets in the Nepali language, with 4,966 entities in total. DanfeNER covers 5 entity types (Person, Location, Organization, Event, and Date) compared to the existing 3 entities type that previous work has covered. DanfeNER also provides training and testing data sets to benchmark models. Our NER methods do not need any feature engineering.

Corpus Preparation

We scrapped tweets from Twitter in June 2022 using Tweepy (Roesslein 2009), an open-source Python library that provides advanced options and filters for searching tweets. We passed on the query as **'lang:ne'** on our search function, which only extracts the tweets that have the Nepali language along with time-stamps, URL, text, user, replies, etc. We did not put any other constraints, so our corpus contains tweets related to different topics like news, entertainment, sports, history, society, economics, and literature. The Twitter API had a rate limit, so we had to collect data in multiple independent requests. We removed the duplicate tweets obtained from the independent requests.

[1] https://github.com/nowalab/DanfeNER

Data Preparation

The extracted tweets were preprocessed using different text preprocessing techniques to remove the noise from tweets like hashtags, mentions, emojis, HTTP links, and special characters like RT. We also removed short tweets that had less than five tokens. We obtained a corpus of 85,418 tweets after these steps.

Annotation Targets and Guideline

For the annotation, we considered five different entity types, namely Person (PER), Location (LOC), Organization (ORG), Event (EVT), and Date (DAT). The tweets were annotated based on the annotation guideline proposed by (Niraula and Chapagain 2022), which contains a proper description of how to annotate each entity and examples for the Nepali language in Nepali.

Annotation Process

We randomized the corpus and loaded it in Label Studio, which is an open-source data labeling tool for text, image, audio, video, and time series data (Tkachenko et al. 2020). We used two native Nepali annotators with advanced degrees. The annotators annotated together some common tweets before annotating independently. The inter-rater agreement among the raters was 0.75, which was calculated using Cohen's Kappa (McHugh 2012), proving significant agreement amongst the annotators.

The typical process in NER annotation is first to tokenize text into tokens and mark tokens corresponding to the predefined entity categories. However, word tokenization remains challenging for morphologically rich languages like Nepali, as words have many word forms corresponding to various characteristics, including number, gender, honor, and tense. Also, attaching different suffixes to the words can create different meanings and forms, making tokenizing the sentences challenging. To overcome such problems, we annotated our tweets at the character level using the Label Studio text annotation tool.

DanfeNER Benchmark Data Sets

We annotated 7,667 tweets in total and obtained 4,996 total entities. We split the annotated records into train and test sets using a 70-30 split procedure and created standard DanfeNER benchmark data sets: DanfeNER-train and DanfeNER-test. It becomes the first benchmark data set for NER in Nepali tweets. Table 2 shows the statistics of the train and test data sets. DanfeNER-train has 5,366 tweets, 92,425 tokens, and 3,463 entities in total. DanfeNER-test has 2,301 tweets, 39,133 tokens, and 1,503 total entities. Person (PER), Location (LOC), and Organization (ORG) are the top-three NER categories we discovered in Nepali tweets. Date (DAT) entities are also frequently found, but Event (EVT) entities are the least used entities in Nepali tweets out of the five NER categories we have considered in this study. We use these benchmark data sets for training and evaluating several machine-learning models in our experiments.

Experiments and Analysis

Models

Transformer models have shown state-of-the-art performances even for Nepali NER tasks (Niraula and Chapagain 2022). Due to their performances, transformers models are being available for Nepali. Monolingual Nepali transformer models are trained from scratch using Nepali text, while multilingual models are trained to combine other languages. We have used five Nepali transformer models for our experiments listed in Table 3. The models include both monolingual and multilingual models. All these models are available in HuggingFace [2].

Npvec1-BERT: nowalab/nepali-bert-npvec1 is our baseline model. This is the first ever known monolingual BERT model for Nepali (Koirala and Niraula 2021). It is a part of NPVec1 which consists of 25 state-of-the-art Nepali Word Embeddings obtained from a comprehensive corpus utilizing Glove, Word2Vec, FastText, and BERT algorithms (Koirala and Niraula 2021).

NepaliBERT: Ranjan/NepaliBERT is a language model based on the BERT model for Nepali. This model was pre-trained using 6.7 million lines of text from the Large Scale Nepali Corpus and OSCAR Nepali corpus and has 82 million parameters. It employed a word-piece tokenizer with a vocabulary size of 50,000 tokens.

NepBERT: amitness/nepbert model was pre-trained using the Nepali CC-100 dataset which contains 12 million sentences, utilizing a Tesla V100 GPU from Google Colab. It has 83.5 million parameters and employs a Byte-level BPE tokenizer with a vocabulary size of 52,000 tokens.

DB-BERT: Sakonii/distilbert-base-nepali model was trained using the Nepali text language modeling dataset, which is a combination of the OSCAR, cc100, and a set of Nepali articles scraped from Wikipedia. The texts in the training set were grouped into blocks of 512 tokens during the training process. The model was trained using the same configuration as the original distilbert-base-uncased model.

BERT-bbmu: bert-base-multilingual-uncased is a transformers model that has been pre-trained using a self-supervised approach on a vast corpus of multilingual data (Devlin et al. 2018). The text was transformed into lowercase and divided into tokens using WordPiece, with a shared vocabulary size of 110,000. The bert-base-multilingual-uncased model is trained in 102 languages.

Experiment Settings

We trained all of the transformer models in the following way: 10 epochs, training batch size 10, and a learning rate of 0.0001. We used Google Colab (Bisong and Bisong 2019) and NERDA Python library (Kjeldgaard and Nielsen 2021) for our experiments.

Evaluation Metrics

We used the micro-averaged precision, recall, and F_1 scores for evaluating NER models using seqeval python

[2] https://huggingface.co

Data	No. Tweets	Tokens	Avg. Len	LOC	ORG	PER	EVT	DAT	Total Entities
Train	5,366	92,425	17.22	923	782	1,061	34	663	3,463
Test	2,301	39,133	17.00	389	356	444	28	286	1,503
Total	7,667	131,558	17.11	1,312	1,138	1,505	62	949	4,966

Table 2: DanfeNER data sets for NER in Nepali tweets.

Notation	Model	Hugging Face Model Id	Tokenizer	Vocab	Train Data	Params
NPVec1-BERT	BERT	nowalab/nepali-bert-npvec1	WP	30000	Wiki, OSCAR, news	22.5M
NepaliBERT	BERT	Rajan/NepaliBERT	WP	50000	LSNC, OSCAR	82M
NepBERT	RoBERTa	amitness/nepbert	BBPE	52000	CC-100	83.5M
DB-BERT	DistilBERT	Sakonii/distilbert-base-nepali	SP	24581	OSCAR, CC-100, Wiki	67M
BERT-bbmu	mBERT	bert-base-multilingual-uncased	WP	105879	Wiki, 102 languages	110M

Table 3: Transformer models: the first four are Nepali monolingual BERT models, and the last one is the multilingual BERT model that includes Nepali. WP, BBPE, and SP refer to WordPiece, Byte-level BPE, and SentencePiece, respectively.

package (Nakayama 2018), which is compatible with the CoNLLshared-task evaluation scheme.

Model	Pre.	Rec.	F_1
NPVec1-BERT	0.63	0.62	0.63
NepaliBERT	0.72	0.69	0.70
NepBERT	0.71	0.69	0.70
DB-BERT	**0.80**	**0.80**	**0.80**
BERT-bbmu	0.76	0.74	0.75

Table 4: Model comparison using micro-averaged F_1 scores.

Model	Pre.	Rec.	F_1	Support
DAT	0.78	0.84	0.81	286
EVT	0.53	0.29	0.37	28
LOC	0.83	0.86	0.84	389
ORG	0.79	0.79	0.79	356
PER	0.81	0.77	0.79	444
Micro Avg.	0.80	0.80	0.80	1,503
Macro Avg.	0.75	0.71	0.72	1,503
Weighted Avg.	0.80	0.80	0.80	1,503

Table 5: Performance evaluation of the best performing DB-BERT model per named entities.

Results and Discussions

We trained all of the transformer models using the DanfeNER-train set and evaluated them on the DanfeNER-test set. Model performances using F_1 micro-average are shown in Table 4. The model with the lowest performance was for NPVec1-BERT with 0.63, 0.62, and 0.63 precision, recall, and F_1 score, respectively. This is likely because compared to other transformer models, NPVec1-BERT has the fewest number of parameters, was pre-trained using just one epoch, and required special preprocessing for input, which we did not respect in this experiment. The best-performing model was DB-BERT with 0.80, 0.80, and 0.80 precision, recall, and F_1 scores, respectively.

We provided the detailed performance report for the best-performing model DB-BERT in Table 5. Compared to Event, the model performed better overall for Location, Date, Person, and Organization types. The F_1 score for the Event was just 0.37. This is because we just had 34 examples in the DanfeNER-train set for Event types compared to hundred of examples for other categories. Events are less frequently mentioned in tweets compared to the other categories. This exact behavior was also reported by another study for NER in Nepali news (Niraula and Chapagain 2022).

Existing Nepali NER Models for Nepali Tweets

Prior research has shown that performances of the NER systems developed for formal text such as news articles significantly drop for detecting NERs in tweets (Liu et al. 2011). It happens due to domain mismatch between formal texts and tweets since tweets have a limited number of words and informal and messy grammar. It would be interesting to see how Nepali NER systems trained in formal language perform on tweets. To that end, we first trained all Nepali transformer models listed in Table 3 on EverestNER-train and evaluated them on the EverestNER-test set, which is created for formal news articles. The results are shown in Table 6. The best-performing transformer model for formal Nepali news was again the DB-BERT with 0.87 F_1 score.

Next, we trained the best performing DB-BERT model in different training settings as shown in Table 7. We trained it with (a) EverestNER-train, formal news train data set (the first row), (b) DanfeNER-train, our train data set for tweets (the second row), and (c) EverestNER-train and DanfeNER-train, combining both formal and informal languages (the third row). Although we have different training sets, we evaluated models only on the DanfeNER-test set. The experiment showed that the existing NER model for news had a F_1 score of 0.69 for tweets. In contrast, the NER model trained on tweet data achieved a F_1 score of 0.80 for tweets, achieving 11% more in the F_1 score than the NER model trained on the news. This justifies that NER in Nepali tweets does require a tweet-specific model. Our model trained using both news and tweets further improved the F_1 score up to 81% for tweets, a slight increase in performance. It, however, improved the macro F_1 score to 0.75, a 3% increase due to the improved performance on the Event category.

Model	**Pre.**	**Rec.**	**F_1**
NPVec1-BERT	0.72	0.71	0.71
NepaliBERT	0.78	0.77	0.78
NepBERT	0.76	0.75	0.75
DB-BERT	**0.88**	**0.87**	**0.87**
BERT-bbmu	0.87	0.84	0.85

Table 6: Models are trained on Nepali news (formal text) and tested on Nepali news (formal text) using EverestNER data sets. Metric used: micro-averaged F_1 score.

Train Data	**Pre.**	**Rec.**	**F_1**
EverestNER-train	0.65	0.73	0.69
DanfeNER-train	0.80	0.80	0.80
DanfeNER-train + EverestNER-train	0.79	0.83	0.81

Table 7: DB-BERT performances under different training data sets. The evaluation was performed using the same DanfeNER-test set. Note that DanfeNER is our data set for Nepali tweets, and EverestNER is the NER data set for formal Nepali news. Metric used: micro-averaged F_1 score.

Error Analysis

We performed a detailed error analysis by comparing the output predicted by a model in the DanfeNER-test set and the corresponding gold labels. We observed six different types of errors and issues listed in Table 8. Code-switching and mixed writing scripts were one of the sources of errors (A). The NER model performed poorly in Nepali tweets that used foreign names such as *Planning Boys*, *B Division*, and *Elon Musk* (B). It is likely because short tweets don't provide enough context, and the foreign terms probably don't get a good representation in the transformer models. The model sometimes had partial prediction (C), most likely due to the limited context available in the short tweets. Language ambiguity is always a challenge in NLP, and we observed those issues as well (D) (e.g. काने, कान and फ्राई, फ्राईडे). Tokenization remains an important challenge for Nepali. We got incorrect predictions due to such cases as well (E). Finally, in some cases, we got errors due to the Hindi language. When we downloaded tweets from Twitter using **lang:ne** filter, it also returned some tweets in the Hindi language. Hindi is also written in Devanagari script like Nepali, and Tweeter's language detection algorithm makes mistakes. Although the writing script is the same, vocabulary and grammar differ between Nepali and Hindi. The NER model for Nepali thus made some errors in the Hindi tweets (F).

Conclusion and Future Work

We presented our DanfeNER system that discovers named entities from Nepali tweets corresponding to five categories: Person, Location, Organization, Event, and Date. We developed high-quality DanfeNER-train and DanfeNER-test data sets to train and benchmark NER systems for Nepali tweets. We released this data at our GitHub repository [3].

We trained several state-of-the-art Nepali monolingual and multilingual transformer models using DanfeNER data sets and achieved the F_1 score up to 81%. We demonstrated through our experiments that NER systems trained for Nepali news perform very poorly for NER in Nepali tweets. We also performed a detailed error analysis of model performances.

Future work includes discovering named entities in transliterated Nepali tweets, i.e., tweets in romanized forms, handling code-switching, and exploring different network architectures.

A. Code-Switching
तर/O म/O त्यो/O रेस/O मा/O छैन/O ’/O Prachanda/~~O~~
Gold: Prachanda/B-PER

B. Foreign Names
झापा/~~B-LOC~~ सँगै/~~O~~ प्लानिङ/~~O~~ ब्वाइज/~~O~~ पनि/O बी/O डिभिजन/O मा/O बढुवा/O /O
Gold: झापा/B-ORG; प्लानिङ/B-ORG ब्वाइज/I-ORG

इलोन/~~O~~ मस्क/~~O~~ ले/O गरे/O कडा/O निर्णय/O
Gold: इलोन/B-PER मस्क/I-PER

C. Partial Prediction
बैठक/O मा/O सिता/~~O~~ देवी/B-PER यादव/I-PER
Gold: सिता/B-PER देवी/I-PER यादव/I-PER

D. Language Ambiguity
मोरंग/B-LOC को/O कता/O काने/~~O~~ पोखरी/~~O~~ तिर/O कान/O को/O प्रति/O मा/O बनाएछन्/O
Gold: काने/B-LOC पोखरी/I-LOC

फ्राई/~~B-DATE~~ गरे/O को/O माछा/O भित्र/O राखेर/O कारागार/O भित्र/O पुर्याइयो/O ब्राउन/O सुगर/O
Gold: फ्राई/O

E. Tokenization
बालेन्द्र/B-PER शाह/I-PER र/O सिसडो/~~O~~ ल/~~O~~ बासी/O बिच/O चर्काचर्/O की/O
Correct Tokenization: सिसडो ल –> सिसडोल, चर्काचर् की -> चर्काचर्की

F. Hindi Language
राष्ट्रीय/B-ORG स्वयंसेवक/I-ORG संघ/I-ORG का/O कार्य/O देश/O हित/O का/O कार्य/O है/O ./O /O महात्मा/O गाँधी/~~O~~
Gold: गाँधी/B-PER

Table 8: Different types of errors and issues. A NER model on DanfeNER-test data makes the predictions.

[3]https://github.com/nowalab/DanfeNER

Acknowledgments

We would like to acknowledge Nepali Shabdakosh - www.nepalishabdakosh.com for providing computing resources for this project.

References

Babych, B., and Hartley, A. 2003. Improving machine translation quality with automatic named entity recognition. In *Proceedings of the 7th International EAMT workshop on MT and other language technology tools, Improving MT through other language technology tools, Resource and tools for building MT at EACL 2003*.

Bam, S. B., and Shahi, T. B. 2014. Named entity recognition for nepali text using support vector machines. *Intelligent Information Management* 2014.

Bisong, E., and Bisong, E. 2019. Google colaboratory. *Building machine learning and deep learning models on google cloud platform: a comprehensive guide for beginners* 59–64.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Dey, A.; Paul, A.; and Purkayastha, B. S. 2014. Named entity recognition for nepali language: A semi hybrid approach. *International Journal of Engineering and Innovative Technology (IJEIT) Volume* 3:21–25.

Finin, T.; Murnane, W.; Karandikar, A.; Keller, N.; Martineau, J.; Dredze, M.; et al. 2010. Annotating named entities in twitter data with crowdsourcing. In *Proceedings of the NAACL Workshop on Creating Speech and Text Language Data With Amazon's Mechanical Turk*.

Guo, J.; Xu, G.; Cheng, X.; and Li, H. 2009. Named entity recognition in query. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, 267–274.

Kjeldgaard, L., and Nielsen, L. 2021. Nerda. GitHub.

Koirala, P., and Niraula, N. B. 2021. Npvec1: Word embeddings for nepali-construction and evaluation. In *Proceedings of the 6th Workshop on Representation Learning for NLP (RepL4NLP-2021)*, 174–184.

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.

Limsopatham, N., and Collier, N. 2016. Bidirectional lstm for named entity recognition in twitter messages.

Lipski, J. 1978. Code-switching and the problem of bilingual competence. *Aspects of bilingualism* 250:264.

Liu, X.; Zhang, S.; Wei, F.; and Zhou, M. 2011. Recognizing named entities in tweets. In *Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies*, 359–367.

Maharjan, S.; Blair, E.; Bethard, S.; and Solorio, T. 2015. Developing language-tagged corpora for code-switching tweets. In *Proceedings of The 9th Linguistic Annotation Workshop*, 72–84.

McHugh, M. L. 2012. Interrater reliability: the kappa statistic. *Biochemia medica* 22(3):276–282.

Mollá, D.; Van Zaanen, M.; and Smith, D. 2006. Named entity recognition for question answering. In *Proceedings of the Australasian Language Technology Workshop 2006*, 51–58.

Niraula, N., and Chapagain, J. 2022. Named entity recognition for nepali: Data sets and algorithms. In *The International FLAIRS Conference Proceedings*, volume 35.

Niraula, N. B.; Dulal, S.; and Koirala, D. 2021. Offensive language detection in nepali social media. In *Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021)*, 67–75.

Niraula, N. B.; Dulal, S.; and Koirala, D. 2022. Linguistic taboos and euphemisms in nepali. *ACM Transactions on Asian and Low-Resource Language Information Processing* 21(6):1–26.

Roesslein, J. 2009. tweepy documentation. *Online] http://tweepy. readthedocs. io/en/v3* 5:724.

Sangma, K.; Das, S.; and Majumder, A. Named entity recognition in hindi-english code-mixed tweets. *Journal homepage: www. ijrpr. com ISSN* 2582:7421.

Singh, O. M.; Padia, A.; and Joshi, A. 2019. Named entity recognition for nepali language. In *2019 IEEE 5th International Conference on Collaboration and Internet Computing (CIC)*, 184–190. IEEE.

Singh, K.; Sen, I.; and Kumaraguru, P. 2018. Language identification and named entity recognition in hinglish code mixed tweets. In *Proceedings of ACL 2018, Student Research Workshop*, 52–58.

Tkachenko, M.; Malyuk, M.; Holmanyuk, A.; and Liubimov, N. 2020. Label Studio: Data labeling software. Open source software available from https://github.com/heartexlabs/label-studio.

Wintaka, D. C.; Bijaksana, M. A.; and Asror, I. 2019. Named-entity recognition on indonesian tweets using bidirectional lstm-crf. *Procedia Computer Science* 157:221–228.