UNIVERSITY OF TARTU
Institute of Computer Science
Computer Science Curriculum

Sriyal Himesh Jayasinghe

An Evaluation of Sinhala Language NLP Tools and Neural Network Based POS Taggers

Master's Thesis (30 ECTS)

Supervisor : Kairit Sirts , Ph.D

Tartu 2019

ABSTRACT

An Evaluation of Sinhala Language NLP Tools and Neural Network Based POS Taggers

Abstract : Part Of Speech tagging is a fundamental problem in the NLP domain and Part Of Speech taggers are used to address this challenge. Though Rule based, probabilistic or deep learning approaches can be used to develop a Part Of Speech tagger, deep learning based Part Of Speech taggers have shown better results. All the Part Of Speech tagging researches that have been carried out so far for the Sinhala language have been done using rule based and probabilistic approaches. This research focuses on developing and evaluating deep learning based Part Of Speech taggers using LSTM network for the Sinhala language.In this research we trained 5 deep learning based Part Of Speech tagging models on two different data sets and evaluated the results of those models. The evaluation results have shown that deep learning based Part Of Speech taggers can be used for Sinhala language and their performance is better than the existing rule based or probabilistic Part Of Speech taggers.

Keywords : Natural Language Processing, Part Of Speech, POS tagging, Evaluation, Rule based approach, Stochastic approach, Deep learning,

Singala keele NLP tööriistade hindamine ja närvivõrgul põhinevad POS-sildistajad (ühestajad).

Abstraktne: PoS sildistamine on fundamentaalne probleem, NLP domeenis ja PoS silidistajaid (ühestajaid) kasutatakse selle väljakutse lahendamiseks. Kuigi reeglipõhist, tõenäosuslikku või süvaõppe lähenemisviisi saab kasutada, PoS-sildistaja (ühestaja) väljatöötamiseks, aga süvaõppel põhinevad PoS sildistajad (ühestajad) on paremaid tulemusi näidanud. Kõik senimaani läbi viidud singala keele PoS-sildistamise uuringud, on läbi viidud kasutades reeglipõhist ja tõenäosuslikku meetodit. See uurimistöö keskendub süvaõppel põhinevate PoS-sildistamise (ühendamise) arendamisele ja hindamisele, kasutades singala keele jaoks LSTM-võrku. Selle uurimistöö käigus koolitasime viite (5) süvaõppele tuginevat PoS-sildistamise (ühendamise) mudelit, kahel erineval andmekogumil ja hindasime nende mudelite tulemusi. Hindamistulemused on näidanud, et süvaõppel põhinevaid PoS-sildistajaid (ühestajaid), saab singala keele jaoks kasutada ja nende jõudlus on parem, kui olemasolevad reeglipõhised või tõenäosuslikud PoS-sildistajad (ühestajad).

Märksõnad: Loomulik keele töötlemine, PoS (keeleosa), POS-sildistamine (ühestamine), hindamine, reeglipõhine lähenemisviis, stohhastiline lähenemine, süvaõppimine.

CERCS: P176 Tehisintellekt

List of Abbreviation

POS	Part Of Speech
OOV	Out Of Vocabulary
NLP	Natural Language Processing
LTRL	Language Technology Research Laboratory
NLPC	National Languages Processing Center
SVM	Soft Vector Machine
HMM	Hidden Markov Model
CRF	Conditional Random Fields
LSTM	Long Short Term Memory
UD	Universal Dependencies
CoNLL	Computational Natural Language Learning

Table of Content

Introduction

In this chapter author wishes to present the reader the reasons and motivation that led to undertake this research the goals expected to achieve by carrying out this research.

Problem Domain

Part Of Speech (POS) tagging is a fundamental problem in the Natural Language Processing (NLP) domain. As highlighted by Màrquez and Rodríguez (1998) POS tagging revolves around assigning each word of a text with the proper morphosyntactic tag taking the context of the word appearance into consideration. POS taggers are used in the NLP domain to address this challenge. As highlighted by Stanford Natural Language Processing Group (2019) a POS tagger is a piece of software that reads text in some language and assigns parts of speech to each word. Since POS taggers can be used as an input layer to other NLP tasks such as sentimental analysis, question answering and named entity resolution many researches are being carried out bring out ever improved POS taggers.

Hasan, UzZaman & Khan (2007) have highlighted three primary approaches that can be applied when developing POS taggers .

They are as follows

- Rule based approach - predict the POS for a word based on a set of pre defined rules.
- Stochastic (probabilistic) approach- predict the POS for a word taking the probability of a tag sequence occurring.
- Deep learning approach- predict the POS for a word using deep neural network models.

Sinhala, the native language of the Sinhalese ethnic group is used by a population of over 16 million in Sri Lanka (Sri Lanka. Department of census and statistics, 2012, p.4). Sinhala Language belongs to the Indo-European language tree (Kanduboda, 2011) like the Hindi, Bengali and Urdu languages. But compared to the languages from the same geographical continent the amount and the depth of the researches conducted in all NLP tasks for Sinhala language is very minimum (Wijesiri *et al.* , 2014).

Existing Sinhala POS taggers and limitations

Though for languages such as English POS taggers using various techniques are introduced, only a handful of researches have been carried out for POS taggers in Sinhala language. All the researches so far have been carried out for the Sinhala language POS tagging are based on stochastic approaches or rule based approach.

Herath & Weerasinghe (2004), Jayaweera & Dias (2011), Jayaweera & Dias (2012), Jayasuriya & Weerasinghe (2013), Jayaweera & Dias (2014), Jayaweera & Dias (2015) and Jayaweera & Dias (2016) have proposed Hidden Markov Model (HMM) based POS taggers for the Sinhala Language. The test accuracies of the above mentioned researches have been reported between 60% to 91.5%.

Gunasekara, Welgama & Weerasinghe (2016) have proposed a hybrid POS tagger by combining HMM and rule-based models. This research has managed to produce an accuracy of 72%.

A research done by Dilshani *et al* (2017) have proposed a POS tagger for the Sinhala Language using the Support Vector Machine (SVM) approach with a reported accuracy of 84.68%.

Fernando and Ranathunga (2018) have proposed a POS tagger for the Sinhala language, which reports an accuracy of 87.14% using the Conditional Random Fields (CRF) approach.

With the above mentioned researches it can be seen that all the researches carried out for Sinhala POS taggers have been based on stochastic and rule based approaches. When observing the results of the researches done on POS tagging for other languages it can be seen that deep learning methods have managed to produce better accuracies compared to stochastic or rule based approaches.

Universal Dependencies (UD) is a community project to develop cross-linguistically consistent treebanks annotation for human languages (Universal Dependencies, 2014). Though there are treebanks available for more than 70 humans languages, a treebank for Sinhala language is not available at the moment (Universal Dependencies, 2017a).

Since there is no UD treebank available, Sinhala language has been overlook by the POS tagger libraries which compete at the Computational Natural Language Learning (CoNLL) shared tasks challenge (Zeman *et al,* 2018) as well. The POS tagger libraries which compete at the CoNLL shared tasks challenge are considered to provide cutting edge environments to train custom deep learning POS taggers.

Goal of the research

As it can be seen that

1. there have been no attempt made on developing a POS tagger using the deep learning method for the Sinhala language

2. POS tagger models of the Sinhala language from the libraries of the CoNLL shared task are missing

this research attempts to train and evaluate several deep learning based POS taggers from the libraries which compete at the CoNLL shared task.

Literature Review

In this chapter the author present a review of the various researches carried out on different NLP technologies of the Sinhala language, brief introduction to the chosen libraries from the CoNLL shared task challenge to develop POS taggers for the Sinhala language.

Researches done on Sinhala NLP technologies

Under the researches carried out on various Sinhala NLP technologies the author wishes to discuss about the researches done on morphological analyzers, named entity recognizers and parsers.

Morphological Analyzers

In the NLP domain morphological analyzers are used to decompose a given word into its combining parts taking the context of the word appearing into consideration.

The early foundation for a Sinhala morphological analyzer has been laid by the work of Herath *et al* (1989) and Herath *et al* (1992) by presenting linguistic analysis of Sinhalese grammar and laying down a modular unit structure for a Sinhala morphological analyzer.

Hettige & Karunananda (2006b) has published a rule based Sinhala morphological analyzer which they claim was to be embedded with a English to Sinhala machine translation system that they were developing. This work has not presented any testing results of the work done nor a code to try out the said solution. Hettige & Karunananda (2011) has published a work done for a Sinhala to English machine translator. In this work the authors have highlighted the importance of their morphological analyzer as the morphological generator sits between the Sinhala sentence composer and the translated English words. The authors haven't published major testing results other than mentioning that the accuracy of morphological generator is 96%. Since the testing data or implementation of the said solution isn't available it's impossible to carryout any local testing of the published solution.

Hettige, Karunananda & Rzevski (2012) have published an ontology based work done on a Sinhala morphological analyzer. This work too is claimed to be done for a English to Sinhala machine translation system and as an feature to manage the scalability of the proposed system they have introduced multi-agent architecture. This system has been tested with a test set of 300 words and has produced an accuracy of 96%.

Welgama, Weerasinghe & Niranjan (2013) have proposed a morphological analyzer using morpheme segmentation algorithm and they have reported an accuracy of 51.38%. Fernando & Weerasinghe (2013) has proposed another rule based morphological analyzer for Sinhala verbs with an accuracy of 67.27%. Dilshani & Dias (2017) have proposed another morphological analyzer for Sinhala verbs but results of their work is not publicly available.

Named Entity Recognizers

Named entity recognition revolves around the task of identifying named entities from an unstructured text and classifying them into to pre defined classes.

The first work on named entity recognition for Sinhala language has been done by Dahanayaka & Weerasinghe (2014) where they have developed a Conditional Random Fields model. Since this is the first attempt of a named entity recognition for the Sinhala language they have developed another model on Maximum Entropy to compare their Conditional Random Fields model. The features used in this work were context word, words around the context word and word suffixes. They had trained the model with a data set of 68205 words and tested with a dataset of 5902 words and have reported a precision value of 81.71%, a recall value of 51.34% and a F-measure score of 63.06%

Senevirathne *et al.* (2015) have published another work done using a Conditional Random Fields model. For this research the authors have used a large dataset with 222362 words compared to the work done by Dahanayaka & Weerasinghe (2014). Additionally they have introduced new features namely Context word, length of the word, first word and context word to their model. This work has reported a precision value of 78.36%, a recall value of 66.13% and a F-measure score of 71.73%

Manamini *et al.* (2016) have published another work for a named entity recognizer for the Sinhala language. They have adopted the approach of Dahanayaka & Weerasinghe (2014) by having a Conditional Random Fields model as the baseline model and Maximum Entropy model as the base line. By reviewing work done on other languages this research has introduced a set new features to make the model more accurate and stop over-fitting. The introduced features are frequency of the word, word frequency, first and last word of a sentence, POS tag, gazetteer lists, clue words, outcome prior and cutoff features to expand the feature set set by Senevirathne *et al.* (2015). This model has been trained with a corpus of 110000 words and after performing a 10-fold cross validation the CRF model has produced 40.1%, 29.8% and 34.1% as overall precision, recall and F1 values respectively.

Parsers

Since Parsers act as a computational representation of the grammar of a natural language, indepth knowledge of language grammar is a must for a successful parser. Work done by Liyanage *et al.* (2012) and Kanduboda & Prabath (2013) has set the linguistic background of the Sinhala language required for a Sinhala parser. Hettige & Karunananda (2006a) has published a work about a design and implementation of a Sinhala parser which acts as a component of a machine translation system. In their publication they have highlighted 10 grammar rules the parser works upon. Since the publication more towards publishing the work done on the machine translator they have given less prominence to the parser component. As a result they haven't published any testing or evaluation results nor any implementation of their work is published other than mentioning that they have used Prolog and Java environments. Carrying forward with this work the same authors have done another publication for a computational grammar model for Sinhala to English machine translation (Hettige & Karunananda, 2011). In this publication they have given in-depth explanation about the architecture and the set of rules defined in their proposed parser for overall translator. This proposed parser has been developed based on the context-free grammar production rule concept and the parser has been extended to support 85 rules for nouns and 18 rules for verbs. As with their previous publication they haven't published any substantial test results of the parser other than mentioning the accuracy of their morphological generator. Liyanage *et al* (2012) has published a work done using the context-free grammar rule which covers 10 simple sentence structures.

Choosen Models

The following models were choosen to experiment train a deep learning based Sinhala POS tagger.

1. Stanford NLP library (Stanford NLP, 2019) - Stanford NLP parser is a very famous NLP library among the NLP community and they have performed exceptionally well at the CoNLL-U shared tasks.

2. NLPCube library (NLPCube, 2019) - NLP-Cube pipe line too has performed well at XPOS tagging of the CoNLL-U shared task.

3. ICSPAS (ICS-PAS, 2019) - ICSPAS or known as COMBO is a NLP pipe line which consists of a tagger, lemmatizer and dependency parser.

4. UDPipe Future (UDPipe-Future, 2019) - UDPipe Future is a open python library to train POS taggers. UDPipe Future managed to score the best score in the 2018 CoNLL-U shared task 2018 competition.

5. UDPipe (UDPipe, 2019) - UDPipe is a NLP pipeline designed and developed Charles University of the Czech republic.

Corpora and Word Embeddings

Corpora

The following corpora were used in this research.

1. Language Technology Research Laboratory corpus (Language Technology Research Laboratory, 2016a)

2. National Languages Processing Center corpus (National Languages Processing Centre, 2019a)

Language Technology Research Laboratory (LTRL) corpus is generated by the Language Technology Research Laboratory of University of Colombo Computer Science Department (Language Technology Research Laboratory, 2016b) and has been used as the corpus in work done by Jayasuriya and Weerasinghe (2013), Jayaweera and Dias (2014) and Gunasekara, Welgama & Weerasinghe (2016).

National Languages Processing Center (NLPC) corpus is generated by the National Languages Processing Center of University of Moratuwa (National Languages Processing Centre, 2019b) and has been used as the corpus in work done by Fernando *et al* (2016), Dilshani *et al* (2017) and Fernando and Ranathunga (2018).

Since both corpora had been manually tagged both contained human errors. Additionally both were not formatted according to the ConLLU format. As a result several pre-processing steps had to be carried out. After carrying out the pre-processing steps it was identified that the LTRL corpus contained 91210 word-tag pairs and the NLPC corpus contained 253711 word-tag pairs.

When analyzing the two corpora it was identified that the NLPC corpus is built by taking the LTRL corpus as the baseline and as a result NLPC corpus contained all the sentences of the LTRL corpus.

The two corpora have used two different POS tag sets. Though the LTRL tag set guidelines were taken as the baseline, the NLPC has taken deeper linguistic characteristics of the Sinhala language into consideration to generate a new tag set for their corpus. (Fernando *et al*, 2016, p.03). These factors have made the NLPC corpus to have a greater depth and coverage in the number of tokens and the tag utilization compared to the LTRL corpus.

Language Technology Research Laboratory Corpus

This corpus (Language Technology Research Laboratory, 2016b) is built from Sinhala newspaper article extracts covering areas arts, sports, politics religion and common knowledge. The data set consists of 21 text files where each file contained varying number and length of text representations.

Prepossessing of the corpus

The below table shows the issue of the raw data set and the mitigation steps that were carried out.

Issue	Mitigation steps
Some words were not tagged	Identified such words through a python script and manually tagged the word with the correct POS tag
Tags not present in the tag set were identified	Identified such tags through a python script and manually tagged the word with the correct POS tag
Inconsistencies with the tags used for same word were identified	Manually inspected such words and tagged them with the correct POS tag
Wrong formatting of word-tag pair	Identified such wrong formatting through a python script and manually corrected the format
Wrong usage of punctuation marks	Manually inspected such punctuation marks and corrected them
Not presented in CoNNL-U format	Converted the cleaned data through a python script to the CoNLL-U format.

Analysis of the cleaned corpus

After carrying out the pre-processing steps the cleaned corpus contained 91210 word-tag pairs distributed among 4367 sentences. The 91210 words in the corpus were made out of one or many occurrences of 16372 unique words. The total number of unique words in the whole corpus is calculated at 17.95%.

The table below shows the composition of the full corpus in terms of frequency of frequencies of unique words.

No. of words	1	2-10	11-50	51-100	101-200	201-500	501-1000	1001-2000	> 2001
No. of occurrences	9214	5886	1042	124	69	25	9	2	1
Percentage	56.28%	35.95%	6.36%	0.75%	0.42%	0.152%	0.054%	0.012%	0.006%

Training, development and testing sets

The cleaned corpus was divided into training, development and testing sets as mentioned in the table below.

Set Type	No Of sentences	No Of Word-tag Pairs	Percentage of word-tag pairs against the cleaned corpus
Training set	3879	80336	88.08%
Validation set	269	5432	5.96%
Testing set	219	5442	5.96%%

Analysis of training, development and testing sets

Further analysis were carried out to identify unique word composition of the three sets and number of Out Of Vocabulary (OOV) words of the test and validation sets .

Unique word composition - The below table shows the number and the percentage of unique words.

Set Type	No Of Word-tag Pairs	No Of Unique Words	Percentage of Unique Words
Training set	80336	14726	18.33%
Validation set	5432	2253	41.48%
Testing set	5442	2134	39.21%

Out Of Vocabulary (OOV) analysis

Further analysis was carried out to estimate the number of words that are not in the training set but in the testing set and validation set (OOV words). The table below shows the Out-of-the-bag analysis of the testing set and validation set against the training set.

Set compared against	No of OOV words	Percentage of OOV words	No of OOV unique words	Percentage of OOV unique words
Validation set	1244	22.90%	880	39.06%
Testing set	1134	20.84%	820	38.43%
Testing and validation sets combined	2378	21.87%	1646	43.81%

LTRL Tag Set

The corpus has used 29 POS tags (Language Technology Research Laboratory, 2016b) to label the words. The below table shows the composition of the tags in the training, development and testing sets.

Tag	Description	Training set	Validation set	Testing set
NNM	Common Noun Masculine	3415	287	186
NNF	Common Noun Feminine	335	18	12
NNN	Common Noun Neuter	17987	1519	1446
NNPA	Proper Noun Animate	3270	253	160
NNPI	Proper Noun Inanimate	5522	457	584
PRP	Pronoun	2248	103	88
VFM	Verb Finite Main	2233	158	120
VNF	Verb Non Finite	4171	222	204
VNN	Verb Non Finite Noun	2171	166	162
VP	Verb Particle	6489	339	379
NVB	Noun in Kriya Mula	3017	143	162
JVB	Adjective in Kriya Mula	703	62	24
JJ	Adjective	4831	186	176
RB	Adverb	635	41	30

RP	Particle	3932	149	158
CC	Conjunction	1585	68	92
DET	Determiner	1713	160	142
POST	Postposition	4721	350	395
QFNUM	Number Quantifier	1527	134	176
FRW	Foreign Word	192	1	118
SYM	Symbol	1	0	0
“	Left Quote	407	26	35
”	Right Quote	407	26	35
(	Left Parenthesis	85	15	24
)	Right Parenthesis	85	15	24
,	Comma	1128	77	111
:	Middle-sentence Punctuation	320	48	26
.	Sentence-final Punctuation	3879	269	219
?	Undefined	3327	140	154
Total		80336	5432	5442

National Languages Processing Center Corpus

This corpus (National Languages Processing Centre, 2019b) is built from Sinhala newspaper article extracts and official documents and has been manually tagged. This corpus compromised of a single file which contained text representations of varying lengths.

Prepossessing of the dataset

Though this corpus compared to the LTRL corpus contained far lesser number of human mistakes still the below mentioned pre-processing steps had to be carried out.

Issue	Mitigation steps
Some words were not tagged	Identified such words through a python script and manually tagged the word with the correct POS tag
Wrong usage of punctuation marks	Manually inspected such punctuation marks and corrected them
Not presented in CoNNL-U format	Converted the cleaned data through a python script to the CoNLL-U format.

Analysis of the cleaned corpus

After carrying out the pre-processing steps the cleaned corpus contained 253711 word-tag pairs distributed among 11319 sentences. The 253711 words in the corpus were made out of one or many occurrences of 33050 unique words. The total number of unique words in the whole corpus is calculated at 13.02%. The table below shows the composition of the full corpus in terms of frequency of frequencies of unique words.

No. of words	1	2-10	11-50	51-100	101-200	201-500	501-1000	1001-2000	> 2001
No. of occurrences	17983	11847	2500	377	202	100	26	13	2
Percentage	54.41%	35.85%	7.56%	1.14%	0.611%	0.303%	0.079%	0.039%	0.006%

Training, development and testing sets

The cleaned corpus was divided into training, development and testing sets as mentioned in the table below.

Set Type	No Of sentences	No Of Word-tag Pairs	Percentage of word-tag pairs against the cleaned corpus
Training set	9840	223680	88.16%
Validation set	688	15004	5.92%
Testing set	791	15027	5.92%%

Analysis of training, development and testing sets

Further analysis were carried out to identify unique word composition of the three sets and number of Out Of Vocabulary (OOV) words of the test and validation sets .

Unique word composition - The below table shows the number and the percentage of unique words.

Set Type	No Of Word-tag Pairs	No Of Unique Words	Percentage of Unique Words
Training set	223680	30089	13.45%
Validation set	15004	4896	32.45%
Testing set	15027	5007	33.32%

Out Of Vocabulary (OOV) analysis

Further analysis was carried out to estimate the number of words that are not in the training set but in the testing set and validation set (OOV words). The table below shows the Out-of-the-bag analysis of the testing set and validation set against the training set.

Set compared against	No of OOV words	Percentage of OOV words	No of OOV unique words	Percentage of OOV unique words
Validation set	1825	12.16%	1409	28.78%
Testing set	2203	14.66%	1620	32.35%
Testing and validation sets combined	4028	13.41%	2961	36.05%

NLPC Tag Set

Though the tag set has defined 38 POS tags in the tag description (National Languages Processing Centre, 2016c) the corpus has used only 30 POS tags to label the tokens. The below table shows the composition of the tags in the training, development and testing sets.

Tag	Description	Training set	Validation set	Testing set
NNC	Common Noun	55596	4055	3992
NNP	Proper Noun	23152	1104	1434

PRP	Pronoun	6321	367	442
QUE	Questioning Pronoun	89	6	3
NDT	Deterministic Pronoun	73	2	1
QBE	Question Based Pronoun	142	30	13
VFM	Verb Finite	5919	352	439
VP	Verb Particle	15793	1037	1121
VNN	Verbal Noun	6390	495	409
AUX	Modal Auxiliary	1362	95	124
VNF	Verb Non Finite	11540	640	693
NCV	Noun in Compound Verb	4301	196	222
JCV	Adjective in Compound Verb	2857	171	212
RRPCV	Particle in Compound Verb	3808	220	153
JJ	Adjective	15981	1302	1152
NNJ	Adjectival Noun	5828	386	364
RB	Adverbs	2391	155	101
POST	Postposition	16534	1109	1062
CC	Conjunction	3400	211	158
RP	Particle	4690	566	657
NIP	Nipatha	4094	219	180
DET	Determiner	5340	331	362
CM	Case Maker	2043	100	108
NVB	Noun in Sentence Ending	777	30	39
NUM	Number	5056	354	250
ABB	Abbreviation	1852	131	72
FS	Full Stop	9840	688	791
PUNC	Punctuation	7964	576	459
FRW	Foreign Word	195	48	5

UNK	Undefined	82	28	9
Total		223680	15004	15027

Word Embedding

As argued by Liu, *et al* (2015) word embedding captures both semantic and syntactic information of words to be frequently used in NLP tasks. Since the models that are expected to build using the above explained corpus are neural network based models a suitable word embedding model had to be selected.

Though there are several pre trained word embedding models available for other languages only FastText word embedding models are available for Sinhala language. There are two FastText models available for the Sinhala language and below table provides an evaluation of the two models

Model	Vector Size Used	No of Words Captured	File Size
Grave et al. (2018)	300	808044	1.8GB
Bojanowski et al. (2017)	300	79030	209.3 MB

When choosing a pre trained word embedding model, a key point that should be considered is to choose a model which has a low OOV ratio when compared against the corpus used. The below table shows the OOV analysis of the two pre trained FastText models when compared against the two corpus.

Corpus	Model	No of OOV words	OOV words ratio	No of OOV unique words	Unique OOV words ratio
LTRL Corpus	Grave et al. (2018)	2804	3.07%	1819	11.11%
	Bojanowski et al. (2017)	9239	10.12%	5826	35.59%
NLPC Corpus	Grave et al. (2018)	6075	2.39%	4590	13.88%
	Bojanowski et al. (2017)	26127	10.30%	15062	45.57%

When analyzing the OOV results of the two word embedding models it can be seen that the Grave et al. (2018) model has a lower OOV ratio for both the corpus. Though the memory utilization of this model is far greater when taking the accuracy of the POS models into consideration it was decided to use the Grave et al. (2018) model as the word embedding model.

Testing and Evaluation

This chapter presents the reader with the results of the testing and evaluation of the models trained.

Overall Accuracy

The below table highlights the overall accuracy, average precision,recall and F1 score of the models for the two corpora.

Model Name	LTRL Corpus				NLPC Corpus			
	Accuracy	Precision	Recall	F1Score	Accuracy	Precision	Recall	F1Score
Stanford Model	76.75%	70.93%	73.23%	70.45%	**90.89%**	82.60%	81.70%	81.10%
ICSPAS Model	**80.94%**	79.20%	**84.08%**	**80.81%**	90.36%	78.46%	81.11%	78.96%
NLPCube Model	80.43%	79.97%	81.66%	80.18%	89.98%	**83.67%**	**85.04%**	**83.71%**
UDPipe Model	77.41%	79.34%	81.57%	79.71%	88.29%	80.60	83.29%	80.79%
UDPipe Future	80.26%	**80.27%**	82.13%	80.60%	90.05%	81.35%	80.70%	79.06%

Jayasuriya and Weerasinghe (2013), Jayaweera and Dias (2014) and Gunasekara, Welgama & Weerasinghe (2016) have used the LTRL corpus for their researches. When evaluating the above results it can be seen that all the models trained on the LTRL corpus have the produced better accuracies when compared against the above mentioned works. When comparing LTRL corpus trained models with each other it can be seen that ICPAS model has managed to produce the best accuracy, recall and precision values. Still it can be seen that NLPCube and UDPipe Future models too have performed as good as the ICSPAS model and as a result just the overall accuracy, precision and recall will not be sufficient to ICSPAS is the best model for LTRL corpus.

Fernando *et a*l (2016), Dilshani *et al* (2017) and Fernando and Ranathunga (2018) have used the NLPC corpus for their researches. As with the models trained on the LTRL corpus it can be seen that the models trained on the NLPC corpus too have managed to produce better accuracies than the results published by the above mentioned researches. When comparing the NLPC trained models with each other it can be seen that the NLPCube model has managed to produce the best precision, recall and F1 scores. Still with the above results it can be seen that the Stanford, UDPipe future models too have performed as goos as the NLPCube model.

As a result it was decided to analyse the OOV accuracies of the models.

OOV Accuracies

The below table highlights the OOV and non OOV accuracies of the models for the two corpora.

	LTRL Corpus		NLPC Corpus	
	Non OOV Accuracy	OOV Accuracy	Non OOV Accuracy	OOV Accuracy
Stanford Model	81.42%	58.99%	**92.94%**	78.94%
ICSPAS Model	82.80%	**73.89%**	92.28%	**79.16%**
NLPCube Model	82.52%	72.49%	92.05%	77.89%
UDPipe Model	81.82%	60.67%	91.62%	68.90%
UDPipe Future	**83.07%**	69.57%	92.69%	74.67%

It can be seen that the OOV accuracies of these models are higher than the reported OOV accuracies of the previous researches done using both the corpora. When comparing the OOV accuracies it can be seen that the ICSPAS model has the best OOV accuracy among the models trained using both the corpora. Even with the non OOV accuraies it can be seen that the ICSPAS model has performed well.Since this effort is a mutli class classification effort it was decided to evaluate individual label precision and recall values as well.

LTRL Label Analysis

The below table highlights label wise precision and recall of the models trained from the LTRL corpus.

Tag	Stanford Model		ICSPAS Model		NLPCube Model		UDPipe Model		UDPipe Future Model	
	Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall
NNM	67.12%	87.10%	76.14%	**89.24%**	**79.02%**	87.10%	75%	82.26%	73.215	88.17%
NNF	NA	0%	52.63%	**83.33%**	47.05%	66.66%	69.23%	75%	**77.77%**	58.33%
NNN	78.75%	78.97%	**84.48%**	78.28%	79.13%	**81.81%**	77.06%	80.15%	82.04%	80.22%
NNPA	55.86%	74.38%	**79.54%**	**87.5%**	72.10%	85.625	61.62%	71.25%	73.12%	85%
NNPI	77.56%	53.23%	**90.95%**	**58.56%**	88.21%	55.56%	82.49%	36.30%	87.43%	53.59%
PRP	**81.48%**	75%	80.95%	77.27%	80.23%	**78.41%**	78.16%	77.27%	80.23%	78.40%
VFM	69.86%	**85%**	**73.48%**	80.83%	68.84%	79.16%	71.75%	78.33%	69.06%	80%
VNF	80.70%	90.16%	**86.83%**	87.25%	84.40%	**90.20%**	81.74%	87.75%	85.58%	87.25%
VNN	96.69%	90.12%	96.68%	90.12%	**97.20%**	85.80%	95.45%	90.74%	**96.02%**	89.50%
VP	81.08%	87.07%	86.51%	**89.71%**	**86.92%**	89.45%	81.90%	88.39%	86.56%	88.39%
NVB	64.95%	**77.77%**	**68.51%**	76.54%	67.80%	74.07%	68.02%	72.22%	64.29%	**77.77%**
JVB	18.18%	8.33%	16.12%	20.83%	18.18%	16.66%	**28.13%**	37.5%	**22.22%**	16.66%
JJ	46.44%	85.22%	43.79%	**88.06%**	**52.74%**	81.81%	46.50%	86.93	47.42%	83.52%
RB	**57.70%**	50.0%	44.11%	50%	57.14%	40%	52.94%	**60%**	41.66%	50.0%
RP	78.23%	95.56%	73.02%	**99.36%**	**79.58%**	96.20%	76.11%	96.84%	77.57%	96.20%
CC	**97.70%**	92.39%	77.48%	**93.48%**	81.13%	**93.48%**	81.13%	**93.48%**	81.13%	**93.48%**
DET	92.64%	88.73%	89.44%	**89.44%**	**92.70%**	**89.44%**	89.44%	**89.44%**	90.07%	**89.44%**
POST	92.98%	70.38%	**94.59%**	70.88%	90.16%	71.90%	86.85%	71.89%	88.27%	**72.41%**
QFNUM	86.82%	82.39%	**96.93%**	**89.72%**	93.29%	86.93%	93.46%	81.25%	95.71%	88.63%
FRW	NA	0%	82.95%	90.68%	75.97%	**99.15%**	82.14%	77.97%	**84.55%**	88.13%
“	88.88%	22.85%	**100%**	**100%**	97.22%	**100%**	97.14%	97.14%	100%	97.14%
”	55.73%	97.14%	**100%**	**100%**	**100%**	97.14%	97.14%	97.14%	97.22%	**100%**
(	**100%**	**100%**	88.88%	**100%**	**100%**	91.66%	**100%**	**100%**	**100%**	**100%**
)	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**

,	99.09%	98.19%	**100%**	98.19%	**100%**	88.28%	**100%**	**100%**	**100%**	99.09%
:	72.22%	**100%**	86.66%	**100%**	**100%**	**100%**	**100%**	96.15%	**100%**	**100%**
.	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**	99.54%	**100%**	**100%**	**100%**
?	45.37%	60.39%	46.95%	**64.94%**	**50.26%**	62.99%	48.39%	54.44%	48.39%	58.44%

When analyzing the label wise precision and recall of the models trained using the LTRL corous it can be seen that the ICSPAS model has scored the best precision value on 13 labels and the best recall value on 16 lables. The second best results on label wise precision and recall has been earned by the NLPCube model with best precision score value on 12 labels and best recall score value on 10 labels. The model that has performed poorly on label wise precision and recall has been the Stanford model and in the case of NNF and FRW labels the Standford model has performed rather poorly with classifying all NNF and FRW labels incorrectly leading the true positive and false negative values to be zero thus calculating the precision and recall impossible.

NLTC Label analysis

The below table highlights label wise precision and recall of the models trained from the NTLC corpus.

Tag	Stanford Model		ICSPAS Model		NLPCube Model		UDPipe Model		UDPipe Future Model	
	Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall
NNC	87.64%	**91.85%**	**90.94%**	87.70%	90.18%	87.02%	86.40%	88.05%	88.51%	89.52%
NNP	**91.61%**	78.45%	89.83%	**82.65%**	89.67%	79.91%	86.60%	69.46%	89.55%	77.12%
PRP	**98.86%**	98.41%	97.55%	**99.10%**	97.55%	**99.10%**	96.26%	**99.10%**	97.55%	99.05%
QUE	**100%**	33.33%	NA	0%	66%	66%	75%	**100%**	NA	0%
NDT	NA	0%	NA	0%	NA	0%	NA	0%	NA	0%
QBE	55.55%	**38.46%**	**83.33%**	**38.46%**	62.5%	**38.46%**	55.55%	**38.46%**	66.66%	30.76%
VFM	92.72%	**95.67%**	93.45%	94.30%	92.99%	93.62%	90.94%	93.84%	**95.32%**	92.71%
VP	92.10%	**94.65%**	91.47%	93.75%	**92.79%**	94.20%	92.38%	93.13%	92.57%	94.46%

VNN	**91.11%**	90.22%	88.45%	88.01%	89.80%	**90.46%**	87.06%	85.57%	87.77%	89.19%
AUX	97.56%	96.77%	98.36%	96.77%	98.36%	96.77%	**98.41%**	**100%**	98.34%	95.96%
VNF	**92.05%**	88.60%	88.90%	**89.03%**	89.71%	86.87%	87.03%	88.16%	89.93%	88.88%
NCV	**74.57%**	79.28%	64.13%	83.78%	73.91%	**84.23%**	68.68%	81.98%	72.04%	82.43%
JCV	**76.34%**	**89.66%**	70.98%	85.37%	75.94%	84.91%	73.01%	77.83%	70.08%	80.66%
RRPCV	**88.51%**	**85.62%**	83.67%	80.39%	81.51%	77.78%	80.36%	**85.62%**	84.97%	83.06%
JJ	**85.50%**	83.42%	83.70%	84.72%	81.04%	**85.32%**	80.96%	84.20%	80.75%	84.90%
NNJ	63.30%	70.60%	59.49%	76.65%	59.14%	**79.94%**	57.47%	66.76%	**63.68%**	70.33%
RB	**91.58%**	86.14%	82.40%	88.12%	69.40%	**92.07%**	76.52%	87.12%	81.98%	90.10%
POST	**97.59%**	95.57%	96.13%	**95.95%**	92.79%	95.76%	95.91%	95.10%	96.40%	**95.95%**
RP	99.39%	**99.39%**	**99.69%**	98.47%	99.23%	97.72%	99.27%	97.71%	99.23%	98.47%
NIP	95.14%	97.77%	97.74%	96.11%	96.70%	**97.78%**	95.97%	92.77%	97.15%	94.44%
DET	**99.16%**	**98.61%**	97.54%	**98.61%**	98.61%	98.34%	96.72%	97.79%	98.61%	98.34%
CM	98.16%	99.07%	97.30%	**100%**	99.08%	**100%**	94.73%	**100%**	**99.07%**	99.07%
NVB	**81.08%**	76.92%	67.35%	87.61%	60.38%	82.05%	62.71%	**94.87%**	58.62%	87.18%
NUM	96.76%	**95.6%**	95.54%	94.40%	**98.26%**	90.8%	97.10%	80.4%	97.82%	90.0%
ABB	92.11%	**97.22%**	97.22%	97.22%	93.33%	97.22%	97.05%	91.66%	**97.22%**	97.22%
FS	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**	99.12%	**100%**	**100%**	**100%**
PUNC	**100%**	**100%**	99.78%	**100%**	**100%**	**100%**	**100%**	98.47%	98.49%	**100%**
FRW	41.67%	**100%**	41.67%	**100%**	**62.5%**	**100%**	41.66%	**100%**	41.66%	**100%**
UNK	50%	11.11%	NA	0%	**100%**	**55%**	50%	11.11%	100%	11.11%

When analyzing the label wise precision and recall of the models trained using the NTLC corous it can be seen that the Stanford model has scored the best precision value on 15 labels and the best recall value on 13 lables. Further more it can be seen that NLPCube models has performed next best in the models trained from the NTLC corpus. The ICSPAS model has performed rather poorly with thee QUE and UNK labels with calculating the precision and recall values of those labels impossible. Though UDPipe model has not won many best precision and recall place positions it can be seen that the model has performed at a consistent level.

Accuracies on Training and validation set

Analysis of the training and validation set accuracies were carried out to identify any overfitting tendencies of the models. The below table highlight the training and validation accuracies of the models for the two corpora

Model		LTRL Corpus	NLTC Corpus
Stanford Model	Training set accuracy	85.65	92.67%
	Validation set accuracy	80.70%	92.24%
	Test set accuracy	76.75%	90.88%
ICSPAS Model	Training set accuracy	89.07%	92.92%
	Validation set accuracy	81.86%	92.06%
	Test set accuracy	80.94%	90.36%
NLPCube Model	Training set accuracy	89.07%	93.17%
	Validation set accuracy	81.60%	91.66%
	Test set accuracy	80.43%	89.97%
UDPipe Model	Training set accuracy	97.68%	98.47%
	Validation set accuracy	78.46%	90.46%
	Test set accuracy	77.41%	88.29%
UDPipe Future Model	Training set accuracy	95.07%	94.71%
	Validation set accuracy	81.27%	91.71%
	Test set accuracy	80.26%	90.05%

With the above results it can be seen that all most all the models tend to have a tendency to overfit with the LTRL model with the UDPipe model showing a high over-fitting. Though all the models trained with the NLTC corpus seems to have a low over fitting tendency compared with the LTRL corpus UDPipe model has shown a very high over fitting tendency compared to the the models. This over fitting tendency of the UDPipe model will have to be considered if it's considered for future researches.

Conclusion

This research focused on filling the gap that was there due to no attempt had been made to experiment with a deep learning based POS tagger for Sinhala language. The research initiated with providing a brief introduction into POS tagging and available POS tagger methods followed by a justification to carry on with the research by providing a brief review of the available POS taggers for the Sinhala language. The literature review chapter was focused on carrying out a review of the available NLP technologies for the Sinhala language and providing an introduction to the models expect to trained. Next chapter presented an overview of the corpora used in this research. Testing and evaluation chapter provided the testing and analysis results of the trained model. With the results of the testing and evaluation of the trained models it was identified that the models produced much better accuracies when compared against the previous researches done. Though the accuracies of the trained models were above expectation it was identified that some models are not fully competence to perform as fully pledge taggers due to their low or moderate precision and recall values estimated for individual labels of the corpus. Additionally it was identified that the models tend to over-fit with the LTRL corpus and UDPipe model tend to over fit on both the corpora. Number of instances for some of the labels were not sufficient enough for the models to fully converge to those labels as well. As future enhancement the following steps can be taken

1. The models have been trained using the default network parameters and a research can be taken up in the future to identify the optimal hyper parameters for the models

2. The same models can be further trained with larger corpora.

3. A research can be carried out to build a hybrid model by combing the trained models with the past researches conducted.

References

Manamini, S.A.P.M., Ahamed, A.F., Rajapakshe, R.A.E.C., Reemal, G.H.A., Jayasena, S., Dias, G.V. and Ranathunga, S., 2016, April. Ananya-a named-entity-recognition (ner) system for sinhala language. In 2016 Moratuwa Engineering Research Conference (MERCon) (pp. 30-35). IEEE.

Bojanowski, P., Grave, E., Joulin, A. and Mikolov, T., (2017). Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics, 5, pp.135-146.

Dahanayaka, J.K. and Weerasinghe, A.R., 2014, December. Named entity recognition for Sinhala language. In 2014 14th International Conference on Advances in ICT for Emerging Regions (ICTer) (pp. 215-220). IEEE.

Dilshani, W.S.N. and Dias, G. (2017). A Corpus-Based Morphological Analysis of Sinhala Verbs. The Third International Conference on Linguistics in Sri Lanka, ICLSL 2017. Department of Linguistics, University of Kelaniya, Sri Lanka. P60

Dilshani, N., Fernando, S., Ranathunga, S., Jayasena, S., and Dias, G. (2017). A Comprehensive Part of Speech (POS) Tag Set for Sinhala Language. The Third International Conference on Linguistics in Sri Lanka, ICLSL 2017. Department of Linguistics, University of Kelaniya, Sri Lanka.

Fernando, S. and Ranathunga, S., 2018, May. Evaluation of different classifiers for sinhala pos tagging. In 2018 Moratuwa Engineering Research Conference (MERCon) (pp. 96-101). IEEE.

Fernando, S., Ranathunga, S., Jayasena, S. and Dias, G., 2016, December. Comprehensive part-of-speech tag set and svm based pos tagger for sinhala. In Proceedings of the 6th Workshop on South and Southeast Asian Natural Language Processing (WSSANLP2016) (pp. 173-182).

Fernando, N. and Weerasinghe, R., 2013. A morphological parser for sinhala verbs. In Proceedings of the International Conference on Advances in ICT for Emerging Regions.

Gunasekara, D., Welgama, W.V. and Weerasinghe, A.R., 2016, September. Hybrid part of speech tagger for sinhala language. In 2016 Sixteenth International Conference on Advances in ICT for Emerging Regions (ICTer) (pp. 41-48). IEEE.

Grave, E., Bojanowski, P., Gupta, P., Joulin, A. and Mikolov, T., (2018). Learning word vectors for 157 languages. arXiv preprint arXiv:1802.06893.

Hasan, F.M., UzZaman, N. and Khan, M., 2007. Comparison of different POS Tagging Techniques (N-Gram, HMM and Brill's tagger) for Bangla. In Advances and innovations in systems, computing sciences and software engineering (pp. 121-126). Springer, Dordrecht.

Herath, S., Ikeda, T., Ishizaki, S., Anzai, Y. and Aiso, H., 1992. Analysis system for Sinhalese unit structure. Journal of Experimental & Theoretical Artificial Intelligence, 4(1), pp.29-48.

Herath, S., Ikeda, T., Yokoyama, S., Isahara, H. and Ishizaki, S., 1989, October. Sinhalese morphological analysis: a step towards machine processing of Sinhalese. In TAI (pp. 100-107).

Herath, D.L. and Weerasinghe, A.R., 2004, December. A stochastic part of speech tagger for Sinhala. In Proceedings of the 06th International Information Technology Conference (pp. 27-28).

Hettige, B. and Karunananda, A.S., 2006a, August. A parser for sinhala language-first step towards english to sinhala machine translation. In First International Conference on Industrial and Information Systems (pp. 583-587). IEEE.

Hettige, B. and Karunananda, A.S., 2006b, December. A Morphological analyzer to enable English to Sinhala Machine Translation. In 2006 International Conference on Information and Automation (pp. 21-26). IEEE.

Hettige, B. and Karunananda, A.S., 2011, September. Computational model of grammar for English to Sinhala Machine Translation. In 2011 International Conference on Advances in ICT for Emerging Regions (ICTer) (pp. 26-31). IEEE.

Hettige, B., Karunananda, A.S. and Rzevski, G., 2012. Multi-agent System Technology for Morphological Analysis. Proceedings of the 9th Annual Sessions of Sri Lanka Association for Artificial Intelligence (SLAAI), Colombo.

ICS-PAS (2019). COMBO is jointly trained tagger, lemmatizer and dependency parser.
Available at: < https://github.com/CoNLL-UD-2018/ICS-PAS > [Accessed 10 May 2019]

Jayasuriya, M., and Weerasinghe, A. R. (2013). Learning a stochastic part of speech tagger for Sinhala. In Proceedings of the International Conference on Advances in ICT for Emerging Regions. (pp. 137-143). IEEE.

Jayaweera, A.J.P.M.P. and Dias, N.G.J., 2011. Part of Speech (POS) tagger for Sinhala language, Proceedings of the Annual Research Symposium 2011, Faculty of Graduate Studies, University of Kelaniya, pp 81

Jayaweera, A.J.P.M.P. and Dias, N.G.J., 2012. Evaluation of Stochastic Based Tagging Approach for Sinhala Language, Proceedings of the Annual Research Symposium 2012, Faculty of Graduate Studies, University of Kelaniya, pp 89.

Jayaweera, A. J. P. M. P., and Dias, N. G. J. (2014). Hidden Markov Model Based Part of Speech Tagger for Sinhala Language. arXiv preprint arXiv:1407.2989.

Jayaweera, A.J.P.M.P. and Dias, N.G.J., 2015. Unknown Words Analysis in POS tagging of Sinhala Language. arXiv preprint arXiv:1501.01254.

Jayaweera, M. and Dias, N.G.J. 2016. Comparison of Part of Speech taggers for Sinhala Language. In proceedings of the 17th Conference on Postgraduate Research, International Postgraduate Research Conference 2016, Faculty of Graduate Studies, University of Kelaniya, Sri Lanka. p 35

Kanduboda, A.B., 2011. The Role of Animacy in Determining Noun Phrase Cases in the Sinhalese and Japanese Languages. ことばの科学, 24, pp.5-20.

Kanduboda, A. and Prabath, B., 2013. On the usage of Sinhalese differential object markers object marker/wa/vs. object marker/ta. Theory and practice in language studies, 3(7), p.1081.

Language Technology Research Laboratory (2016a). *Downloads.* electronic dataset. Language Technology Research Laboratory - UCSC Sinhala Tagged Corpus, Available at: < http://ltrl.ucsc.lk/download/1304/?uid=ba77fe0dde > [Accessed 04 February 2019]

Language Technology Research Laboratory (2016b). *Home.* Language Technology Research Laboratory - UCSC, Available at: < http://ltrl.ucsc.lk/ > [Accessed 04 February 2019]

Liu, Y., Liu, Z., Chua, T.S. and Sun, M., (2015). Topical word embeddings. In Twenty-Ninth AAAI Conference on Artificial Intelligence.

Liyanage, C., Pushpananda, R., Herath, D.L. and Weerasinghe, R., 2012, March. A computational grammar of sinhala. In International Conference on Intelligent Text Processing and Computational Linguistics (pp. 188-200). Springer, Berlin, Heidelberg.

National Languages Processing Center (2019a). *Sinhala-POS-Data.* electronic dataset. National Languages Processing Center - news- verified- final level.txt, Available at: < https://github.com/nlpc-uom/Sinhala-POS-Data/blob/master/news-%20verified-%20final%20level.txt > [Accessed 10 July 2019]

National Languages Processing Center (2019b). *Home.* National Languages Processing Cente - University of Moratuwa, Available at: < https://www.mrt.ac.lk/web/nlp > [Accessed 10 July 2019]

National Languages Processing Center (2019c). *Sinhala-POS-Data.* electronic dataset. National Languages Processing Center - Tagging Guide.pdf, Available at: < https://github.com/nlpc-uom/Sinhala-POS-Data/blob/master/Tagging%20Guide.pdf > [Accessed 10 July 2019]

NLP- Cube (2019). Natural Language Processing Pipeline - Sentence Splitting, Tokenization, Lemmatization, Part-of-speech Tagging and Dependency Parsing Available at: < https://github.com/adobe/NLP-Cube > [Accessed 10 May 2019]

Màrquez, L. and Rodríguez, H., 1998, April. Part-of-speech tagging using decision trees. In European Conference on Machine Learning (pp. 25-36). Springer, Berlin, Heidelberg.

Senevirathne, K.U., Attanayake, N.S., Dhananjanie, A.W.M.H., Weragoda, W.A.S.U., Nugaliyadde, A. and Thelijjagoda, S., 2015, December. Conditional Random Fields based named entity recognition for sinhala. In 2015 IEEE 10th International Conference on Industrial and Information Systems (ICIIS) (pp. 302-307). IEEE.

Sri Lanka. Department of census and statistics (2012). *Census of Population and Housing 2012 - Population by Province.* Available at: < http://www.statistics.gov.lk/pophousat/cph2011/index.php?fileName=SriLanka&gp=Activities&tpl=3 > [Accessed 1 August 2019]

Stanford NLP (2019a). Official Stanford NLP Python Library for Many Human Languages

Available at: < https://github.com/stanfordnlp/stanfordnlp > [Accessed 10 May 2019]

UDPipe (2019). UDPipe Available at: < http://ufal.mff.cuni.cz/udpipe#download > [Accessed 10 May 2019]

UDPipe-Future (2019). NCoNLL 2018 Shared Task Team UDPipe-Future

Available at: < https://github.com/CoNLL-UD-2018/UDPipe-Future > [Accessed 10 May 2019]

Universal Dependencies. 2014. Introduction. [ONLINE] Available at: < https://universaldependencies.org/docs/introduction.html. [Accessed 02 February 2019]

Universal Dependencies. 2017a. Current UD Languages. [ONLINE] Available at: < https://universaldependencies.org/#current-ud-languages [Accessed 02 February 2019]

Universal Dependencies. 2017b. CoNLL-U Format. [ONLINE] Available at: < https://universaldependencies.org/format.html. [Accessed 02 February 2019]

Welgama, V., Weerasinghe, R. and Niranjan, M., 2013. Evaluating a machine learning approach to sinhala morphological analysis. In Proceedings of the 10th International Conference on Natural Language Processing, Noida, India.

Wijesiri, I., Gallage, M., Gunathilaka, B., Lakjeewa, M., Wimalasuriya, D., Dias, G., Paranavithana, R. and De Silva, N., 2014, January. Building a wordnet for sinhala. In Proceedings of the Seventh Global WordNet Conference (pp. 100-108).

Zeman, D., Hajič, J., Popel, M., Potthast, M., Straka, M., Ginter, F., Nivre, J. and Petrov, S., 2018, October. CoNLL 2018 shared task: multilingual parsing from raw text to universal dependencies. In Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies (pp. 1-21).

Appendix

Trained models

1. The trained models using the LTRL corpus, datasets, python scripts used to clean the data, calculate the accuracies, precision, recall, F1 score can be found in the following google link

https://drive.google.com/open?id=1zW3s2wXNVqGYQvYdTtnNF8Z2tCUxD9v4

2. The trained models using the NTLC corpus, datasets, python scripts used to clean the data, calculate the accuracies, precision, recall, F1 score can be found in the following google link

https://drive.google.com/open?id=1jnPdXzVSwQIlw8QKY3kr_uxqD30P6guW

Licence

Non-exclusive licence to reproduce thesis and make thesis public

I, Sriyal Himesh Jayasinghe,

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to reproduce, for the purpose of preservation, including for adding to the DSpace digital archives until the expiry of the term of copyright, An Evaluation of Sinhala Language NLP Tools and Neural Network Based POS Taggers,

supervised by Dr. Kairit Sirts.

2. I grant the University of Tartu a permit to make the work specified in p. 1 available to the public via the web environment of the University of Tartu, including via the DSpace digital archives, under the Creative Commons licence CC BY NC ND 3.0, which allows, by giving appropriate credit to the author, to reproduce, distribute the work and communicate it to the public, and prohibits the creation of derivative works and any commercial use of the work until the expiry of the term of copyright.

3. I am aware of the fact that the author retains the rights specified in p. 1 and 2.

4. I certify that granting the non-exclusive licence does not infringe other persons' intellectual property rights or rights arising from the personal data protection legislation.

Sriyal Himesh Jaysinghe

14/08/2019