Int. j. inf. tecnol. (September 2020) 12(3):717–729
https://doi.org/10.1007/s41870-020-00480-2

ORIGINAL RESEARCH

Using machine learning to build POS tagger for under-resourced language: the case of Somali

Siraj Mohammed[1]

Received: 5 November 2019 / Accepted: 19 May 2020 / Published online: 3 June 2020

Abstract POS tagging serves as a preliminary task for many NLP applications. It refers to the process of classifying words into their parts of speech (also known as words classes or lexical categories). Somali is a member of the Cushitic languages with limited number of NLP tools for use. An accurate and reliable POS tagger is essential for many NLP tasks like shallow parsing, dependency parsing, sentiment analysis, and named entity recognition. In this paper, we present a statistical POS tagger for Somali language using different machine learning approaches (i.e., HMM and CRF) and neural network model. Our Somali POS tagger outperforms the state-of-the-art POS tagger by 87.51% on a tenfold cross-validation. The key contribution of this paper are (1) building a generic POS tagger, (2) comparing the performances with the existing state of the art techniques, and (3) exploring the use word embeddings for Somali POS tagging.

Keywords Part-of-speech tagger · Machine learning · Neural network · Somali

1 Introduction

Part of Speech (POS) tagging is one of the basic applications of NLP (Natural Language Processing) on any language. It is a process of assigning a tag to every word in a sentence and serves as a preliminary task for carrying out tasks like chunking, dependency parsing, and named-entity recognition on any language. All of these NLP systems must use part of speech tagger as their preprocessor components for their best performance [1–3]. So our work focuses on carrying out POS tagging for Somali. Much of the research in POS tagging has been devoted to resource-rich languages like English and French. African languages like Somali have received far too little attention. Somali language belongs to the lowland East Cushitic family of Afro-Asiatic language. Other languages in the East Cushitic family include Afar, Oromo, Rendille and Boni. Somali language claims an estimated 16 million speakers in Somalia, Somaliland, Djibouti, Kenya, and Ethiopia. Somali Language has been one of the under-resourced languages both in terms of electronic resources and processing tools. Recently, insufficient attempts have been made to develop Somali corpus. A Somali text corpus that has linguistic information is publicly available at http://www.somalicorpus.com/ for public [4]. However, this available of resources not has been used as resource to process NLP tasks like POS Tagger which is becoming a barrier for researches of higher level NLP applications.

Given this circumstance, there is a need to develop a POS tagger for Somali. In this paper, we present an effective POS tagger using different machine learning approaches (i.e., HMM and CRF) and neural network models for under-resourced language-in the case of Somali.

1.1 Somali Language and Writing System

The Somali language is a member of the Cushitic languages which include Oromo and Afar that are spoken in the Horn of Africa. This group of dialects is in turn a member of the Afro-asiatic family of languages such as Mandara, Aramaic, Arabic, Hebrew, Egyptian, and, among

✉ Siraj Mohammed
sirmoh4@gmail.com

[1] Department of Information Technology, College of Engineering and Technology, Jigjiga University, Jijiga, Ethiopia

others [5]. Somali, the national language of Somalia, is also spoken in Djibouti, Eritrea, Ethiopia, and Kenya. Hence, Somali is a regional language that is spoken in the Horn of Africa Region and, in many countries around the world due to the immigration of about 2.5 million Somalis to many parts of the world including UK, US, Canada, Finland, Netherlands, and Sweden. According to [4] Somali language has three major varieties: (1) AfSoomaali or Common Somali, (2) Benaadir and (3) Af-Maay or Maay. A Somali language claims an estimated 16 million speakers in Somalia, Somaliland, Djibouti, Kenya, and Ethiopia.

Different writing system was invented by Somali linguistic scholars such as, Borama (Gadabuursi) alphabet, Somali (Osmanya) alphabet, Kaddare alphabet and, Latin alphabet for Somali. The Somali language did not have an official writing alphabet system, until the former Somalia President Slad Barre, formally introduced the Somali Latin alphabet in October 1972 [6]. The Somali Latin alphabet was invented by Shire Jama Ahmed (Shire Jaamac Axmed) and his system was chosen from among eighteen competing new orthographies [6]. At the same time, Somali was made the sole official language of Somalia. Somali is written today in the Latin script from left to right [7]. For our POS tagging, we have applied Latin alphabet writing system because the Latin alphabet was adopted after 1972. Latin alphabet is described as follows.

’	Bb	Tt	Jj	Xx	Kh kh	Dd	Rr	Ss	Sh sh	Dh dh
Alef	Ba	Ta	Jeem	Xa	Kha	Deel	Ra	Siim	Shiim	Dha
[ʔ]	[b]	[t]	[tʃ]	[ħ]	[x]	[d]	[r]	[s]	[ʃ]	[d]
Cc	Gg	Ff	Qq	Kk	Ll	Mm	Nn	Ww	Hh	Yy
Ayn	Ghayn	Fa	Qaff	Kaaf	Laan	Miim	Nun	Waw	Ha	Ya
[ʕ]	[g]	[f]	[G]	[k]	[l]	[m]	[n]	[w/ʉ:/ u:]	[h]	[j/i:/ I:]
Aa	Ee	Ii	Oo	Uu	Aa aa	Ee ee	Hi ii	Oo oo	Uu uu	
A	E	I	O	U						
Æ/ a	E/ɛ	I/j	ɵ/ɔ	ʊ/ u	Æ:/ a:	E:/ɛ:	ɘ:/ ɔ:	ɞ:/ɔ:	ʊ:/ u:	

Notes

- The Somali alphabet has 26 letters, of which 21 are consonants and the remaining five are vowels, which can be long or short.
- The Somali alphabet uses all the letters of the English alphabet, except p, v and z.
- Tones are marked as follows: the high tone with an acute accent (á), the low tone with a grave accent (à) and the falling tone with a circumflex (â).
- Vowels can contrast breathy voice and harsh voice, and vowel length

1.2 POS Tagging Ambiguity, Tone, Inflection and Derivation in Somali Language

One important challenge for part-of-speech (POS) are lexical ambiguity in which words may have different meanings, inflection and derivation in given language. For instance, look at the following word structure that can have different meanings due to verbal and nominal derivational affixes [8].

- Wuu Dhisayaa ‘he is building’
- Wuu Dhisnayaa ‘he is building for himself
- Wuu hagaajinayaa ‘he is arranging
- Wuu Hagaajisanayaa ‘he is arranging for himself

Besides verbal and nominal derivational affixes, tone can distinguish lexical items, gender, number and case [9]. This is illustrated as follows:

- *qáan* ‘young camels’ versus *qaán* ‘debt’ (two distinct lexical items)
- *ínan* ‘boy’ versus *inán* ‘girl’ (gender)
- *mádax* ‘head’ versus *madáx* ‘heads (number)
- *géri* ‘house.ABS’ versus *gerí* ‘house.GEN’ (case)

These different meanings are caused by tone, verbal and nominal derivational affixes. To a human being, the intended meaning of the above sentence is clear depending on the situations, but for a computer it is far from obvious. The study therefore, presents the research into the design and development of an effective part-of-speech tagger for the Somali language. The purpose of Somalia tagging is to assign the correct tag classes to each word.

The rest of the paper is organized as follows. Section 2 describes related works, Sect. 3 describes the proposed approach, and Sect. 4 explores design, methods, and experimental processes. Section 5 presents experimental evaluations and results. Section 6 presents comparison of the proposed work with the existing works. Finally, the conclusions and future research direction are presented in Sect. 7.

2 Related Work

Tagging is a process that accepts string of untagged word and provides appropriate tag for each of individual words in a sentence [2]. It is a method to categorize words based on their grammatical or syntactic group in a sentence or a corpus. During tagging process, symbols (labels) are assigned to each word in the sentence that tells us the word’s category in the given sentence. These labels are termed as “tags”. Tagging can be done, manually or automatically. Manual tagging is done by hand and correct tag is assigned after group discussions of experts on each

word tags. The requirement of too much time for large amount of corpus and acquisition of knowledge about language grammar and sentence structure are the main problems for manual tagging approach. Tagging can be done automatically by using POS tagger software like Stanford POS tagger.

Various algorithms and methods were introduced to tackle part of speech tagging problem, such as stochastic approach, rule based approach, hybrid approach [10], and Artificial Neural Network. Stochastic method which can also be called statistical method is the method that works based on the statistical information. Any approach that uses probability or statistic information can be grouped under this approach. One of the most commonly known stochastic approaches is hidden Markov model (HMM). The basic idea of HMM is that most likely tag of a given word in sequence of words is chosen by calculating the probability of all possible sequence of tags and then selecting the sequence with the maximum probability [2]. Rule based POS tagging is a dominant approach in computational linguistics and natural language processing that uses large database that contains hand written rules to remove or minimize ambiguities.

This section presents the earlier POS-tagging works conducted in the lowland East Cushitic family of Afro-Asiatic language, which are categorized under the same language branch as the Somali language. To our knowledge, no prior POS tagging research conducted for Somali language. We reviewed the earlier POS-tagging works, which are categorized under the same language branch as the Somali language. Among these categorized language, [11] proposed hidden Markov model (HMM) approach for part-of-speech tagging for Afaan Oromo language using unigram and bigram implementations techniques. The corpus was collected from various public newspapers in Afaan Oromo. The method was tested by training and testing sets using 159 sentences with 1621 words. Experimental results show that the proposed approach is successfully demonstrated and obtained an accuracy of 89.7% [11]. However, this work only considers HMM approach. Also, performance comparison between proposed work and related work is missing. Abraham et al. [12] developed a model for automatic part-of-speech tagging for Oromo language using maximum entropy Markov model (MEMM). This method was tested by training and testing sets using 452 sentences with 6094 words. Experimental results show that the proposed approach is successfully demonstrated and obtained an accuracy of 93.01%. However, the performance comparison between proposed work and related work is missing. Therefore, there is a need to develop POS tagger for Somali. In this paper, we present an effective POS tagger using different machine learning approaches (i.e., HMM and CRF) and neural network models for under-resourced language-in the case of Somali.

3 Proposed approach

Motivation: Much of the research in POS tagging has been devoted to resource-rich languages like English and French. African languages like Somali have received far too little attention. Somali Language has been one of the under-resourced languages both in terms of electronic resources and processing tools. This fact motivates us to develop a POS tagger for Somali.

Contribution: The key contribution of this paper are (1) corpus creation, (2) building a generic POS Tagger, (3) comparing the performances with state of the art techniques(Performance comparison between proposed work and related work), and (4) exploring the use word embeddings for Somali POS tagging. In addition, we contribute the first POS tagger and annotation guidelines for such text and release a new dataset of Somali language.

Methodological steps of the proposed approach: the proposed approach consists of the following steps:

- *Step 1.* Initialization of algorithm parameters. The parameters of Somali POS tagger algorithm are initialized. The critical parameter in both cases is the penalty parameter ḍ. A too small value for Ɓ causes under-fitting and a too large value for ɑ causes over-fitting. In other words, a small value for Ɓ will allow a larger number of training errors, while a large value will minimize training errors. We experimented with cost parameters 0.30 (CRF) and 0.50 (HMM) to give higher accuracies.
- *Step 2.* Reading a sentence.
- *Step 3.* The third step is to tokenize the string or sentence to access the individual word/tag strings.
- *Step 4.* After reading and tokenize a sentence or string, the next step is starting the model (i.e. *HMM, CRF, Neural Networks*) that trained by our datasets to labeling words with their appropriate Part-Of-Speech (e.g., Noun, Verb, Adjective, Adverb, Pronoun). Figure 1 shows the flowchart of the proposed work.

4 Design

4.1 Corpus details

A series of raw text of the corpus was collected from Somalia online newspapers. The corpus includes topics from education, history, culture, politics, health, sport, and

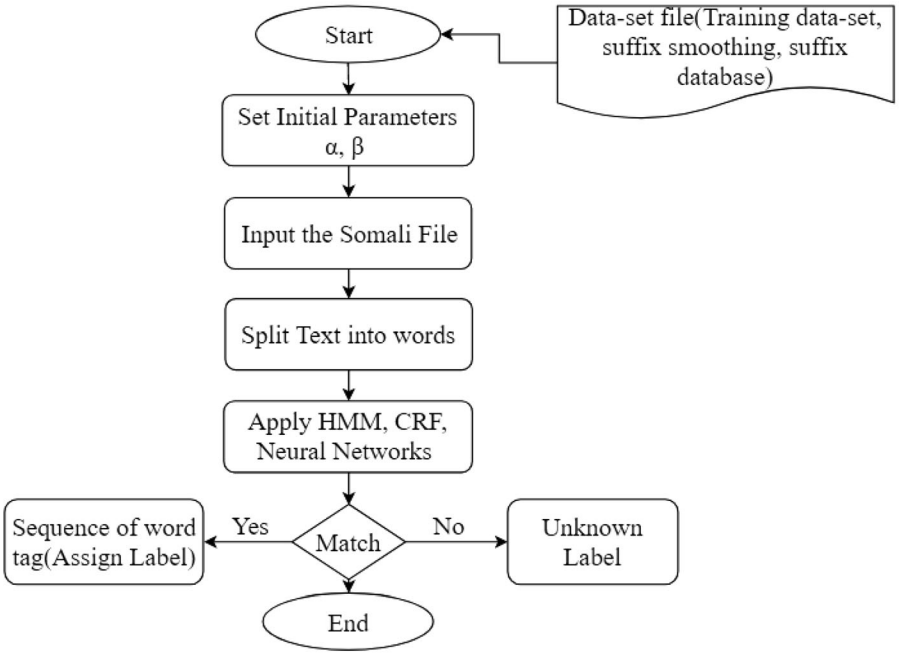

Fig. 1 Flowchart of proposed algorithm

agro-pastoral. Thereafter, pre-preprocessing for cleaning and normalizing the text was carried out. The cleaned corpus was used for POS manual tagging. The manually annotated text contains 14,369 tokens. This corpus is used to test and train the Somali POS tagging and to evaluate tagger performance. It consists of 14,369 tokens (representing 1234 sentences) with 24 different taggers. Figure 2 demonstrates the graphical representation of corpus distributions as follows.

4.2 Preprocessing

In the preprocessing phase the raw text was cleaned and formatted into plain text and XML formats. First, unnecessary punctuation marks and foreign scripts were removed. The corpus was then structured into XML by following TEI corpus encoding standards [13]. During a corpus creation, we had spelling problems because writing system for the Somali language has still not been stabilized. The same word can be written according to people in different ways. This lack of standardization is common for the Somali language. Nimaam [14] show that the word president in Somali language writing system appears like *madaxweyne* or *madaxwayne.* To resolves this problem, for a single word like madaxweyne, we thought the most frequent spelling was the right one. If *madaxweyne* appears 6 times in the corpus and *madaxwayne* 3 times, *madaxwayne* is transforming into *madaxweyne*. For the component

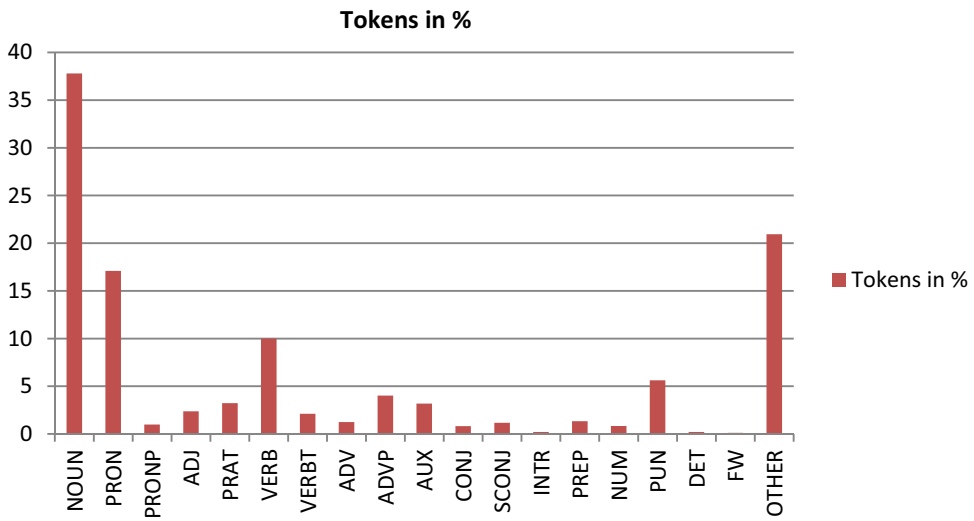

Fig. 2 Graphical representation of corpus distributions

words like *iskumid* and *isku mid*, we have taken the merged orthography like *iskumid*. As such, by minimizing orthographic variations, normalization is expected to clean the corpus.

4.3 The Tagset design

During the design of the corpus and the tagger, a grammar of the Somali language, brown corpus manual, related POS-tagging works conducted in the lowland East Cushitic family of Afro-Asiatic language like Afaan Oromo were reviewed [11, 12, 15–17]. A few Somali linguists have classified Somali part of speech into four only true and fundamental categories. These are substantive, adjective, particle, and verb. But, there are other recognized parts of speech by combinations or forms of the above part of speech such us, nouns, numerals, pronouns, adverbs, relative prepositions, definite article, demonstrative, adjective, possessive pronominal adjective, interrogative and, adjective. Table 1 has the list of all labels of post tags.

5 Methods

We mainly used three kinds of Methods:

- Hidden Markov Model (HMM) Method
- Conditional Random Fields (CRF) Method
- Neural Networks Method

The methods are detailed in the below subsections.

5.1 Hidden Markov Model (HMM)

HMM approach was used for this task since it does not need detail linguistic knowledge of the language as rule based approach. For HMM implementation, we used the open source Viterbi algorithm. Microsoft Visual Studio 2010 was used to develop the proposed system of graphical user interface for only HMM approach. This graphical user interface for only HMM approach is present in Fig. 3 as follows.

As can be seen from Fig. 3, the user interface has different functions. Among them, the left window has the system corpus that loaded from the system, and including user corpus option to enter any corpus or texts from user side. On the middle window we can observe the total words with the number of occurrences, tag counts, tag frequencies and their descriptions. The right pane lists words with tag frequencies, tags with tag frequencies, provides information about what the current tag is, provides lexical and transition probability estimated values.

5.1.1 Experimental Results for HMM Model

Usually, the experiments for POS tagger are starting with partitioning the completed corpus into training and test sets. We used training set for learning, and the test set for assessing the overall performance of the tagger. In our experiments, the whole training data set was partition into ten equivalent sizes (each size is 10% of the total training set). The accuracy of the tagger was tested starting

Table 1 Post tagging with category

S.no.	Category	Type	Tags	Frequency of Tags	Example
1	Noun		NOUN	5439	Bannaankii
2	Pronoun		PRON	1590	Kuwa
3		Proper noun	PRONP	143	Igu, iga, ani, aniga
4	Adjective		ADJ	343	wayn, yar, madow
5	Particle		PART	463	Waa
6	Verb		VERB	309	Hab, karo
7		Transitive	VERBT	304	adkai, ‘adadi
8	Adverbs		ADV	181	hadda, caawa
9	Adverb position		ADVP	579	Ka, uga
10	Auxiliary		AUX	458	Leh, lahayn
11	Coordinating Conjunction		CCONJ	117	an, ama, marka
12	Subordinating conjunction		SCONJ	169	Ee, oo
13	Interjection		INTJ	30	ayo !, ayai !
14	Prepositions		PREP	192	iyo, kor, hoos, dushi,
15	Numeral		NUM	120	Hal, labo, saddex
16	Punctuation		PUNC	808	?, -,.
17	Determiner		DET	31	Kuwa, kii
18	Foreign Words		FW	18	Book, Politics
19	Others		OTHER	3016	–

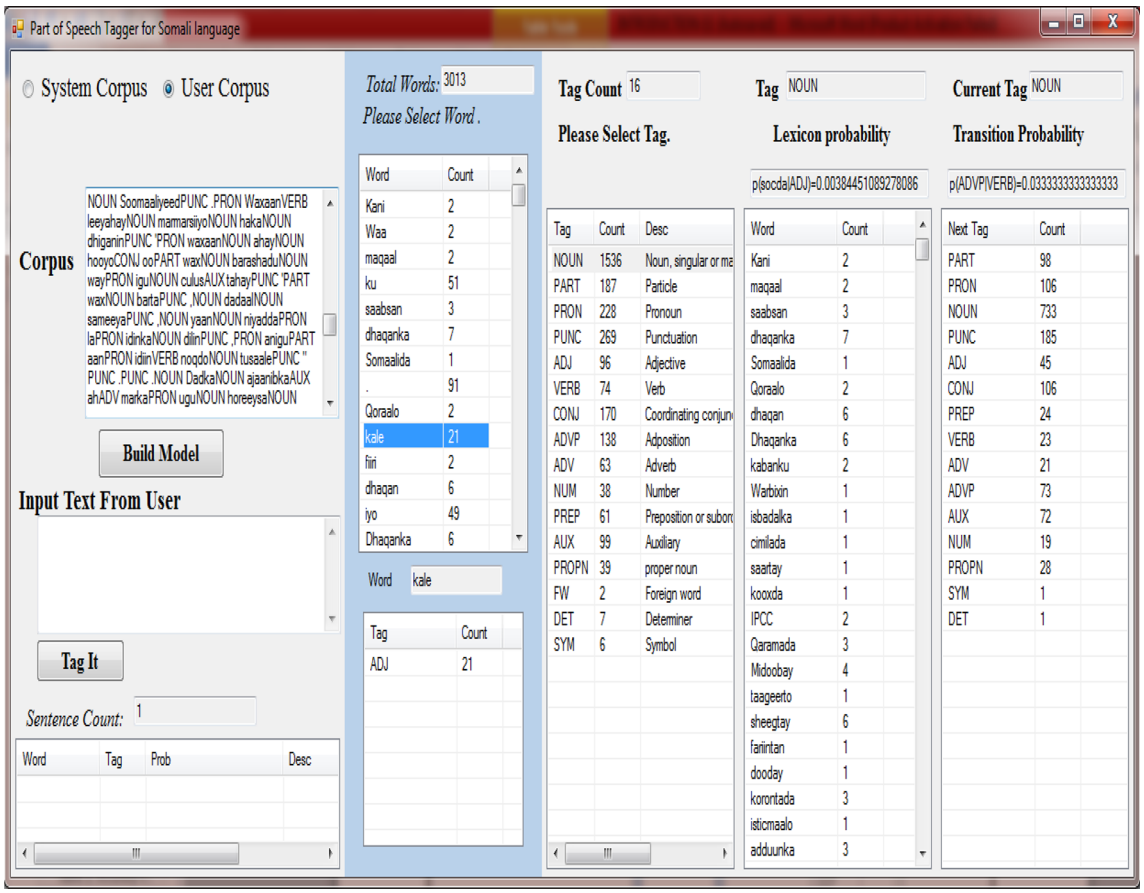

Fig. 3 GUI for HMM POS Tagger

by the first 10% of the data and repeating the process by adding 10% of training data set to the previous data until the entire training corpus is used. For every 10% of training sets added, the accuracy variation is recorded. We evaluated our tagger performance from two different aspects: (1) considering all tokens in the test sets for overall accuracy and (2) considering accuracy for known and unknown words. Table 2 demonstrates the distribution of known and unknown words.

Table 2 shows the data for each fold in terms of the total number of tokens and divided into known and unknown tokens, where the term unknown refers to tokens that are not in any of the nine folds in the other.

5.1.2 Transition Probability

In our case, tags are the unobservable states which produced the observable output i.e. words. The Bigram algorithms used for the tagged corpus and estimate their probabilities by counting occurrences of tags, tag–tag pairs, not tag–word pairs. Hence, as per the bigram assumption, tag *ti* depends on tag *ti-1* which means the probability of a tag depends only on its previous tag. Thus, the probability that a VERB follows an NOUN would be estimated as follows:

$$P(ti = VERB|ti-1 = NOUN) = \frac{\text{Count(NOUN}(ti-1)\text{ and VERB(ti))}}{total\ number\ of\ bi-grams\ starts\ with\ ti-1} \quad (1)$$

where ti is the current tag and ti−1 is the previous tag.

Table 3 presents sample bigram frequencies computed from Somali training corpus. The corpus consists of 14,369 tokens that have only 24 taggers: NOUN, VERB, ADV, ADJ, and so on. The bigram probability estimated values for the conditional probabilities in column 1, 2 and 3 are given in column 2, 4, and 6, respectively. Figure 4 illustrates the graphical transition probability values for taggers.

5.1.3 Lexicon probability

The lexical probabilities (P (w1|t1) can be estimated simply by counting the number of occurrences of each word by category. The lexical probability is the probability that a particular category is realized by a specific word, not the probability that a particular word falls into a particular category. For example, P (Kani | NOUN) is estimated by

Table 2 Statistics for 10 equivalent sizes training data sets

S. no.	Training data percentage	No. of words	Known	Unknown
1	10	1431	1188	243
2	20	2862	2376	486
3	30	4293	3564	729
4	40	5724	4752	972
5	50	7155	5940	1215
6	60	8586	7128	1458
7	70	10,017	8316	1701
8	80	11,448	9504	1944
9	90	12,879	10,692	2187
10	100	14,310	11,880	2430

Table 3 Sample transition probability

Bigram category	Probability	Bigram category	Probability	Bigram category	Probability
P (PART|NOUN	0.63	P (NOUN|VERB)	0.48	P (NOUN|SYM	0.31
P (PRON|NOUN	0.68	P (NUM|VERB	0.44	P (NOUN|DET)	0.26
P (PUNC|NOUN	0.11	P (ADVP|VERB	0.03	P (CONJ|DET	0.08
P (ADJ|NOUN	0.02	P (PRON|VERB)	0.04	P (ADV|ADJ	0.01
P (NOUN|PRON)	0.58	P (PART|PREP)	0.05	P (PROPN|ADJ)	0.01
P (PART|PRON	0.04	P (PUNC|PREP)	0.02	P (AUX|ADJ)	0.05
P (AUX|PRON	0.24	P (CONJ|PREP)	0.05	P (NOUN|NUM)	0.38

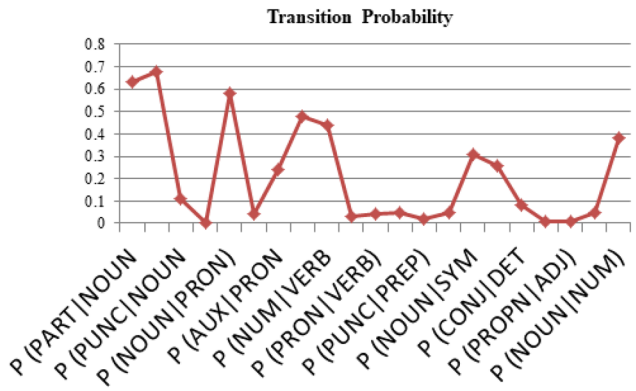

Fig. 4 Graphical representation of transition probability

Count (# times Kani is as NOUN)/Count (# times as NOUN occurs). Table 4 presents sample lexical probability estimated values. For more please see the following equations.

$$P(Kani|NOUN) = \frac{Count(\#timesKaniisasNOUN)}{Count(\#TimesasNOUNOccurs)} \quad (2)$$

The proposed approach has been validated with standard evaluation metrics for different data sets. We use Precision, Recall and F1-Score for evaluation. 5000 words were used to evaluate the proposed approach. We have constructed three test data sets for testing. Table 5 shows the test case values for precision, recall and f1-score.

$$\text{Precision} = \frac{\text{TP}}{TP + FP} \quad (3)$$

$$\text{Recall} = \frac{\text{TP}}{TP + FN} \quad (4)$$

$$\text{F1 - score} = 2 * \frac{(R * P)}{R + P} \quad (5)$$

where TP (True Positive)—assigns correct tags to the given words, FP (false positive)—assigns incorrect tags to the given words, FN (false negative)—not assign any tags to a given words.

The system has an average accuracy of 85.7%. To our knowledge, the accuracy of this system is promising even though we have used the smallest of the corpus that already tagged. In addition to the accuracy results just reported in the previous section, precision, recall, and F-measure can also be used to examine closely the performance of the algorithms with respect to each POS tags. Table 6 shows the average precision, recall, and f-measure for individual POS tags.

In HMM model, punctuation marks (PUNC) have been identified correctly 100% as punctuation marks (PUNC). The algorithm predicted 18.2% of the foreign words correctly as foreign words (FW) and 81.8% are as non- foreign words. Moreover, the proposed approach predicted 87% of the noun words correctly as noun words (NOUN) and 13% are as non-noun words. Figure 5 presents the graphical

Table 4 Sample lexical probability

Words with given lexical probability	Probability	Words with given lexical probability	Probability	Words with given lexical probability	Probability
P (Kani|NOUN)	0.001	P (Yahay|VERB)	0.004	P (#|SYM)	0.0031
P (Somaalida|NOUN)	0.006	P (Ahayd|VERB)	0.003	P (Kii|DET)	0.002
P (Qoraalo|NOUN)	0.011	P (Qaroo|VERB)	0.002	P (Kuwa|DET)	0.001
P (Haboon|NOUN)	0.0006	P (Noqdo|VERB)	0.002	P (Kale|ADJ)	0.014
P (Aniga|PRON)	0.001	P (Laga|PREP)	0.011	P (Badan|ADJ)	0.016
P (Labada|PRON)	0.004	P (Loo|PREP)	0.008	P (Dheer|ADJ)	0.0034
P (Aye|PRON)	0.033	P (Lyadoo|PREP)	0.0052	P (2100|NUM)	0.0014

Table 5 Average precision, recall and, F-measure for HMM model

Test Set	Precision	Recall	F-measure
Set1	0.89	0.84	0.860
Set2	0.93	0.79	0.854
Set3	0.83	0.89	0.858
Average	0.88	0.84	0.857

Table 6 Average precision, recall and, F-measure for individual POS tags

S. no.	Tags	Precision	Recall	F-measure
1	NOUN	87	84.3	85.62
2	PRON	82	81.97	81.98
3	PRONP	83.7	80.39	82.01
4	ADJ	78.4	82.87	80.57
5	PART	85	82.90	83.93
6	VERB	75	73.83	74.91
7	VERBT	79	76.8	74.4
8	ADV	81.20	79.4	80.28
9	ADVP	85	83.4	84.19
10	AUX	86.8	84.6	85.68
11	CCONJ	81.52	84.3	82.88
12	SCONJ	75	81.63	78.17
13	INTJ	83.7	84.78	84.23
14	PREP	79.4	76	77.66
15	NUM	83.98	85.02	84.49
16	PUNC	100	100	100
17	DET	79	76.8	77.88
18	FW	18.2	9.4	12.39
19	Other	85	83.49	84.23

representation of average Precision, Recall and F-measure of individual POS tags.

5.1.4 Error Analysis for HMM Model

Performing transition and lexical experiments, and get output in probabilities are no sufficient to measure the effectiveness of our POS tagger. Performing a confusion matrix can be a better idea to accurate the propose POS tagger model and to know what sorts of errors it makes. To do this, we trained our tagger on 14,369 tokens from the training sets. Then, we tested the proposed POS tagger using 5000 tokens in sentences from the testing set. Based on our experiment, the confusion matrix for the error analysis is presented in Table 7.

As can be seen from the confusion matrix experiment, we have seen that the four most common mistakes are classifying NOUN as PRON, PRON as VERB, ADV as NUN, and NOUN as OTHER. We also see that PUNCT and NUM are always correctly classified as before. In general, the taggers, of the whole 5000 tags about 4451 tags are tagged correct and 549 tags are tagged wrongly. The performance of the tagger is 84.2%. The main reason to get this accuracy values is there is no well-structured corpus for under resourced language like Somali.

As can be seen from the Fig. 6 on the above, the proposed model yields promising result on NOUN, PRONP, AUX, INTJ, NUM, and PUNC part of speech tags but, it has negative impact on the four POS tags: VERB, ADV, FW and OTHER. The main reason getting negative impact on the four POS tags are: (1) rule based approaches not integrated to our model to detect them, (2) no standard corpus for this under-resourced language, and (3) manual tagger might be made error during label words with tags.

5.2 Conditional Random Fields (CRF) Method

Both HMM and CRF modeling methods have been trained and tested on the same dataset using exactly the same features. Parameters have also been selected for both. The parameters of Somali POS tagger algorithm are initialized. The critical parameter in both cases is the penalty parameter ɖ. A too small value for Ɓ causes under-fitting and a

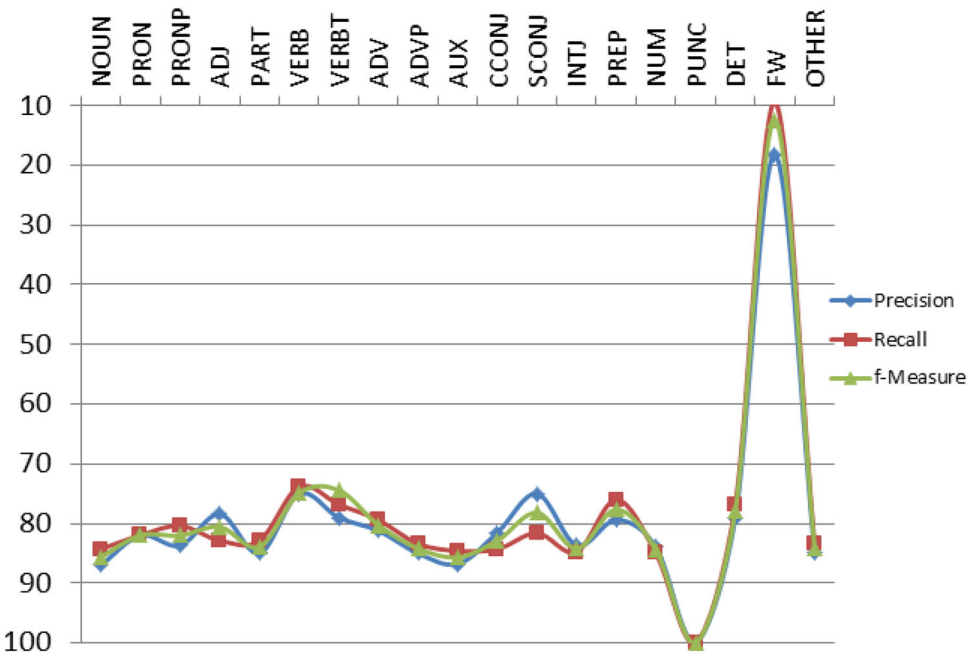

Fig. 5 Graphical representation of average precision, recall and F-measure of individual POS tags

too large value for α causes over-fitting. In other words, a small value for Ɓ will allow a larger number of training errors, while a large value will minimize training errors. We experimented with cost parameters 0.30(CRF) and 0.50 (HMM) to give higher accuracies. In the CRF-based model, we experimented with context features that are similar to the history available to the HMM-based models, especially the affix-augmented model. The feature templates were specified according to the format supported by the CRF++ tool. We used features that captured immediate bigram tag history on both sides of the current word. Average accuracies on tenfold cross-validation are present in Table 8.

On a tenfold cross-validation, CRF achieves an average accuracy of 88.25%, while HMM achieves 85.70% under exactly the same conditions. The average accuracies for both algorithms on known and unknown tokens are shown in Table 8. As can be seen from the table, CRF achieves a slightly higher average accuracy of 80.67% than HMM (80.29%) on unknown tokens, which may lead to the conclusion that CRF model generalizes better. On the other hand, HMM achieves relatively higher on known tokens which explains its slight overall higher accuracy.

5.3 Neural networks method

We used the same set of features as used in CRF model for neural network approach starting with structured perceptron [18]. Structured perceptron was employed using the "seqlearn" library [18]. All the recurrent neural network models were built using simple multilayer perceptron in Keras [19] deep learning framework. Adam [20] optimizer was used as it seems to be well suited for word embeddings experiments. While training the networks, we used approximately 60% of the tagged sentences for training, 30% as the validation set and 10% to evaluate our model. Also, while datasets preprocessing for supervised learning, we fragmented our tagged sentences into three datasets: (1) a training dataset used to train the model, (2) a validation dataset used to modify the parameters of the classifier, and (3) a test dataset used to evaluate the performance of the classifier.

As POS tagging is sequence labeling task [18, 21], we modeled it as a sequence-to-sequence learner. We started with the Vanilla RNN network, which produces an output POS tag for every input word. The vanilla learning model where the fixed length of input vector and output vector sequences is the same. Such a network is the perfect architecture for POS Tagging to a series of given input and produces a series of output vectors. We used Somali word as an input and passed the entire sentence to the neural network. These all experiments were carried out with the help of pre-trained Somali embeddings. We implemented the "encoder–decoder" architecture where the entire sequence of the words or a sentence was represented as a single vector. This vector was then passed onto a decoder which produces an output POS tag for every input word. The proposed architecture for sequence-to-sequence learning using LSTM for POS tagging is present in Fig. 7.

5.3.1 Experimental Results for Neural Network Models

The final tagging performance on the test set is reported in Table 5 that shows the results of different models (SimpleRNN and LSTM) where feature engineering was

Table 7 Confusion matrix of most confused POS tags

	True tag																
	NOUN	PRON	PRONP	PART	ADJ	VERB	AUX	ADV	SCONJ	INTJ	PREP	NUM	PUNC	FW	Other	Total	Performance (%)
Most likely tag																	
NOUN	2454	0	12	0	0	5	0	0	3	0	3	0	0	0	38	2515	97.5
PRON	53	323	0	0	0	72	0	0	0	0	46	0	0	1	10	505	63.9
PRONP	0	0	40	0	0	0	0	0	0	0	0	0	0	0	0	40	100
PART	0	12	0	249	0	15	0	0	0	0	0	0	0	0	5	281	88.6
ADJ	13	0	0	0	148	0	0	0	14	0	5	0	0	0	0	180	82.2
VERB	0	0	0	0	18	112	0	51	0	0	0	0	0	0	0	181	61.8
AUX	0	0	0	0	0	0	59	0	0	0	0	0	1	0	0	334	99.0
ADV	50	0	1	0	0	0	0	100	0	0	2	0	0	0	1	154	64.5
SCONJ	4	0	0	0	0	0	0	0	160	2	0	0	0	0	0	165	96.9
INTJ	0	0	0	0	0	0	0	0	0	51	0	0	0	0	0	51	100
PREP	0	0	11	0	0	0	0	0	0	0	85		0	0	0	96	85.5
NUM	0	0	0	0	0	0	0	0	0	0	0	59	0	0	0	59	100
PUNC	0	0	0	0	0	0	0	0	0	0	0	0	242		0	242	100
FW	0	1	0	0	0	0	0	0	0	1	0	0	0	4	0	18	66.8
OTHER	44	0	0	12	0	0	0	20	0	0	0	0	0	0	103	179	57.5
Total and average																5000	84.2

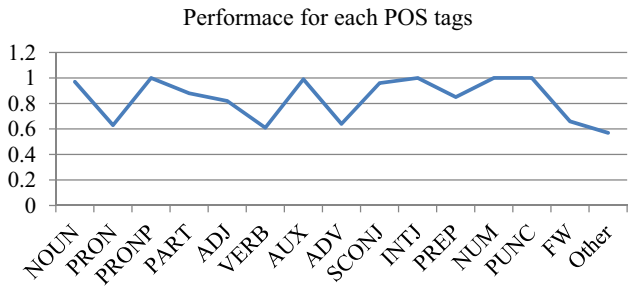

Fig. 6 Tagger performance for each POS tags

Table 8 Average accuracies on tenfold cross-validation (%)

Algorithms	Known	Unknown	Overall
HMM	88.67	80.29	85.70
CRF	88.21	80.67	86.25

required. Table 9 reports the accuracies of neural network models (i.e., SimpleRNN and LSTM) which used non pre-trained word embeddings and/or pre-trained word embeddings. Average accuracies on precision, recall and F1-measure for Neural models are present in Table 9.

All this neural network models was evaluated using precision, recall and F1-measure. On non-pre-trained word embeddings, SimpleRNN achieves an average accuracy of 85.68%, while LSTM achieves 87.78% under exactly the same conditions. On pre-trained word embeddings, SimpleRNN achieves an average accuracy of 89.87%, while LSTM achieves 90.60% under exactly the same conditions.

6 A comparison of the proposed work with the existing works

To the best of our knowledge, no prior works on POS tagging has been done for Somali language. Hence, we compared the earlier POS-tagging works which are categorized under the same language branch as the Somali language. Among these categorized language, [11, 12] proposed Hidden Markov Model (HMM) and Maximum-Entropy Markov Model (MEMM) for part-of-speech tagging for Afaan Oromo language. The highest POS tagging accuracies have been achieved by both HMM and Maximum-Entropy Markov model (MEMM). The maximum-entropy Markov model (MEMM) tagger achieved an average accuracy of 93.01% on a tenfold cross-validation while under the same conditions, HMM achieved an average of 90.77%. Also, the Brill tagger [22] achieved an average overall accuracy of 95.60%, which is statistically higher than 90.77% for HMM. We have compared the performance of Oromo and Somali with limited language resources (i.e. annotated corpora of large size) in Table 10.

For Somali, we did not have any large size annotated corpus available. This lack of large size annotated corpus and sparseness of training data can be the key reason to get low performances. In addition, writing system for the Somali language has still not been stabilized. The same word can be written according to people in different ways. This lack of standardization can cause a decreasing in performances for the proposed approach.

As can be seen from the table, there is no previous work for Somali. Therefore, we develop the first POS tagger for Somali language using three different approaches (i.e., HMM, CRF, and Neural Networks). Since as the first POS tagger, the result of POS tagging accuracy is well. All POS

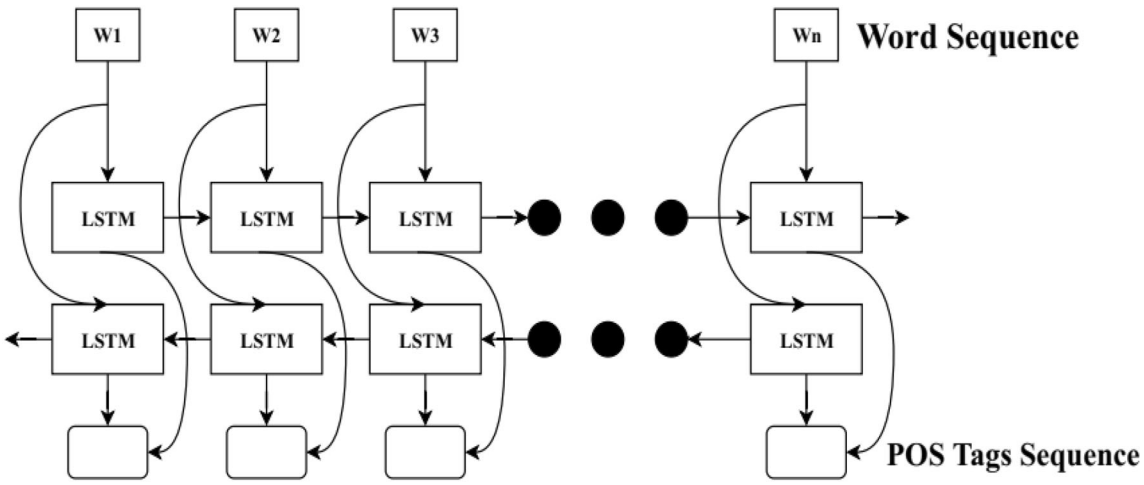

Fig. 7 Sequence to sequence learning using LSTM for POS Tagging

Table 9 Precision, Recall and F1-measure for Neural Models

Model	Features	Precision	Recall	F-measure
Simple RNN	Non pre-trained word embeddings	86.8	84.6	85.68
LSTM		88.08	87.69	87.88
Simple RNN	Pre-trained word embeddings	89.72	90.03	89.87
LSTM		90.02	91.18	90.60

Table 10 Comparison of the proposed work with the existing works

Language	Tag	Training (in word)	Test (in word)	Method	Result (%)
Oromo	17	1315	146	HMM	90.77 [11]
Oromo	26	6750	997	Brill's	95.60 [22]
Oromo	33	5480	614	MEMM	93.01 [12]
(Somali)Proposed work	24	14,369	5000	HMM, CRF, and neural networks	87.51

tagger achieved an average accuracy of 87.51% on a ten-fold cross-validation while under the same conditions.

7 Conclusion and future work

This paper presents a model for an effective part of speech tagging for under-resourced language, in the case of Somali. First, we have collected texts from different domains. Subsequently, pre-preprocessing for cleaning and normalizing the text was carried out. This cleaned corpus has been partitioning into training and test sets. Then, we trained our Part of Speech Tagger on 14,369 tagged tokens, in this cause the frequency of the taggers in the entire corpus, training set and testing set was considered. Finally, the whole training data set was partition into ten equivalent sizes. The accuracy of the tagger was tested starting by the first 10% of the data and repeating the process by adding 10% of training data set to the previous data until the entire training corpus is used. Finally, the average performance of the tagger is 87.51%.

In the future work, further tasks should be considered since as it is the first attempt for Somali. POS tagging accuracy is expected to increase by correcting typographical errors in the untagged corpus and by increasing the accuracy of the morphological analyzer. Some rule-based components can also be used for detecting and correcting existing errors in the model. Furthermore, considering comparative study on different approaches such as, SVM, Rule- based, deep learning based tagger with more training and testing data are expected.

References

1. Yemane K, Kazuhide Y, Ashuboda M (2016) Tigrinya part-of-speech tagging with morphological patterns and the new Nagaoka Tigrinya corpus. Int J Comput Appl 146(14):0975–8887
2. Jurafsky D, Martin JH (2009) Speech and language processing, 2nd edn. Prentice-Hall Inc, Upper Saddle River, p 2009
3. Gebrekidan B (2009) Part-of-speech tagging for Amharic. Bulletin de linguistique appliquée et générale, Presses Universitaires de Franche-Comté, pp 114–120
4. Jama JM (2013) Somali Corpus: state of the art, and tools for linguistic analysis. https://www.academia.edu/26504727/Somali_Corpus_state_of_the_art_and_tools_for_linguistic_analysis. Accessed 12 Aug 2018
5. Darwish K, Mubarak H, Abdelali A (2017) Arabic POS tagging: don't abandon feature engineering just yet. WANLP 2017 (co-located with EACL 2017), pp 130–137
6. Tosco M (2010) Somali Writings. Afrikanistik online, vol 2010. http://www.afrikanistik-online.de/archiv/2010/2723/. Accessed Aug 2017
7. Abdulkadir A (2010) Somali writing system. https://www.omniglot.com/writing/somali.htm. Accessed July Aug 2017
8. David LG (2016) Somali as a tone language. Normandie Université, UR, DySoLa, France Speech Prosody 2016 31, Boston, USA
9. Lampitelli N (2013) Evaluative morphology in Somali. Université Paris Diderot-Paris. http://www.linguist.univ-paris-diderot.fr/nlampitelli/somali_lampitelli_V6.pdf. Accessed July Aug 2017
10. Garside R, Smith N (1997) A hybrid grammatical tagger: Claws4, Corpus annotation: linguistic information from computer text corpora, pp 102–121
11. Getachew M, Million M (2015) Parts of speech tagging for Afaan Oromo. Int J Adv Comput Sci Appl. https://doi.org/10.14569/SpecialIssue.2011.010301
12. Nedjo AT, Huang D, Liu X (2014) Automatic part-of-speech tagging for Oromo language using maximum entropy Markov model (MEMM). J Inf Comput Sci 11(10):3319–3334. https://doi.org/10.12733/jics20103906
13. Francis WN, Kucera H (1997) Brown corpus manual. Providence, Rhode Island Department of Linguistics Brown University 1964. Revised 1971. Revised and Amplified 1979

14. Abdillahi N (2014) Building and evaluating Somali Language Corpora. In: Proceedings of the 2014 workshop on the use of computational methods in the study of endangered languages, Baltimore, Maryland, USA, pp 73–76
15. Kaur D, Jain U (2017) Automatic rule detection and POS tagging of Punjabi text. Int J Eng Comput Sci 6(3):20780–20784. https://doi.org/10.18535/ijecs/v6i3.68
16. Brill E (1992) A simple rule-based part of speech tagger. In: Proceedings of the DARPA speech and natural language workshop, Morgan Kauffman. San Mateo, California, pp 112–116
17. Kerk JWC (2003) A grammar of the Somali language. Trubner Publisher, London
18. Todi KK, Mishra P, Sharma DM (2018) Building a Kannada POS tagger using machine learning and neural network models. arXiv:1808.03175
19. Chollet F et al (2015) Keras. https://github.com/fchollet/keras. Accessed July Aug 2018
20. Kingma D, Ba J (2014) Adam: a method for stochastic optimization. arXiv preprint arXiv:1412.6980
21. Gimenez J, Marquez L (2004) SVMTOOL: a general POS tagger generator based on support vector machines. In: Proceedings of the 4th international conference on language resources and evaluation, Citeseer, pp 43–46
22. Abraham Gizaw Ayana (2015) Towards improving Brill's tagger lexical and transformation rule for Afaan Oromo language. Department of Geographic Information Science, Hawassa Universty, Hawassa