A PROJECT REPORT

ON

“POS Taggers for Indian Language”

SUBMITTED BY

Ms .Dhawade Sarika

Ms .Ghegade Mayuri

Ms.Unde Pratiksha

UNDER THE GUIDENCE OF

Prof. Suryavanshi.A.P

parikrama
Revolution in Education

HSBPVT’s FACULTY OF ENGINEERING, KASHTI

DEPARTMENT OF COMPUTER ENGINEERING

YEAR OF SUBMISSION

2023-2024

parikrama
Revolution in Education

CERTIFICATE

This is certified that **Ms. Sarika Gorakh Dhawade Roll No. 8** of VIII Semester of Bachelor Of Computer Engineering of Institute HSBPVT's Faculty of Engineering, Kashti (Code:5303) has completed the mini project satisfactorily in course Natural Language Programming for the academic year 2023-2024 as prescribed in the curriculum.

Place**:-Kashti.** Seat No.:- **B190774211**

Date **:-** PRN No**:-722155645M**

Subject Teacher **H.O.D.** **Principal**

INDEX

Topic Page No

Abstract

This mini project aims to develop Part-of-Speech (POS) taggers for Indian languages, addressing the need for natural language processing tools in linguistically diverse regions. POS tagging is crucial for various NLP tasks such as sentiment analysis, machine translation, and information retrieval. However, the lack of robust POS taggers for Indian languages poses a significant challenge.

The project will involve collecting annotated corpora for multiple Indian languages, preprocessing the data, and training POS taggers using machine learning or deep learning techniques. Various approaches such as Hidden Markov Models (HMM), Conditional Random Fields (CRF), and Recurrent Neural Networks (RNN) will be explored to develop accurate and efficient taggers.Evaluation of the taggers will be conducted using standard metrics such as accuracy, precision, recall, and F1-score on test datasets.

The performance of the taggers will also be compared with existing tools and benchmarks to assess their effectiveness.Additionally, the project will investigate techniques for handling morphological complexity and out-of-vocabulary words common in Indian languages. This includes exploring techniques like morphological analysis, word embeddings, and transfer learning to improve the robustness of the POS taggers.

ACKNOWLEDGEMENT

I feel emotionally moved when it comes to acknowledgement after the task is accomplished. Our mind is full of image of those who direct or indirectly helped me in this endeavour. We thanks to one and all.

I would like to express my deepest sense of gratitude to my guide Proof. Suryavanshi A. P. who offered her continuous advice and encouragement throughout the course of this project work. We thank her for the systematic guidance and providing all the assistance needed to complete the work. She inspired us greatly to work in this area. Her willingness to motivate us contributed tremendously to our project. Her guidance and discussions with me are invaluable in realization of this report I also thankful to all Faculties Computer Engineering Department for their continuous encouragement and guidance throughout the entire project work.

Ms. Sarika Gorakh Dhawade

(722155645M)

Introduction

Part of Speech (POS) Tagging is the first step in the development of any NLP Application. It is a task which assigns POS labels to words supplied in the text. Each word's tag is identified within a context using the previous word/tag combination. POS tagging is used in various applications like parsing where word and their tags are transformed into chunks which can be combined to generate the complete parse of a text.

Problem Statement

Implement POS tagging for simple sentences written in Hindi or any Indian Language.

OBJECTIVE

Implementation of POS tagging on simple sentences which are written in Hindi or any other Indian Language.

System Requirements

Hardware Requirements:
Ubuntu/Windows OS, 512 MB HDD, 4 GB RAM
Software Requirements:
Python 3. Jupyter Notebook

THEORY CONCEPTS

POS Tag

Part-of-speech (POS) tagging is a popular Natural Language Processing process which refers to categorizing words in a text (corpus) in correspondence with a particular part of speech, depending on the definition of the word and its context

A POS tag (or part-of-speech tag) is a special label assigned to each token. (word) in a text corpus to indicate the part of speech and often also other grammatical categories such as tense, number (plural/singular), case etc. POS tags are used in corpus searches and in text analysis tools and algorithms.

Use of POS Tags

POS tags make it possible for automatic text processing tools to take into account which part of speech each word is. This facilitates the use of linguistic criteria in addition to statistics. For languages where the same word can have different parts of speech, e.g. work in English, POS tags are used to distinguish between the occurrences of the word when used as a noun or verb.

POS tags are also used to search for examples of grammatical or lexical patterns without specifying a concrete word, e.g. to find examples of any plural noun not preceded by an article. Or both of the above can be combined, e.g. find the word help used as a noun followed by any verb in the past tense.

POS Tagset

A set of all POS tags used in a corpus is called atagset. Tagsets for different languages are typically different. They can be completely different for unrelated languages and very similar for similar languages, but this is not always the rule. Tagsets can also go to a different level of detail. Basic tagsets may only include tags for the most common parts of speech (N for noun.

V for verb. A for adjective etc.).

It Is, however, more common to go into more detail and distinguish between nouns in singular and plural, verbal conjugations, tenses, aspect, voice and much more. Individual researchers might even develop their own very specialized tag sets to accommodate their research needs.

POS Tagging

Part-of-speech (POS) tagging is a popular Natural Language Processing process which refers to categorizing words in a text (corpus) in correspondence with a particular part of speech, depending on the definition of the word and its context.

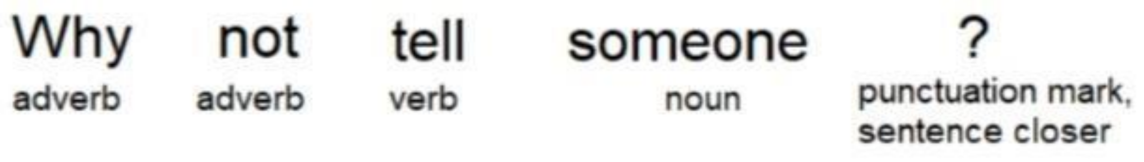
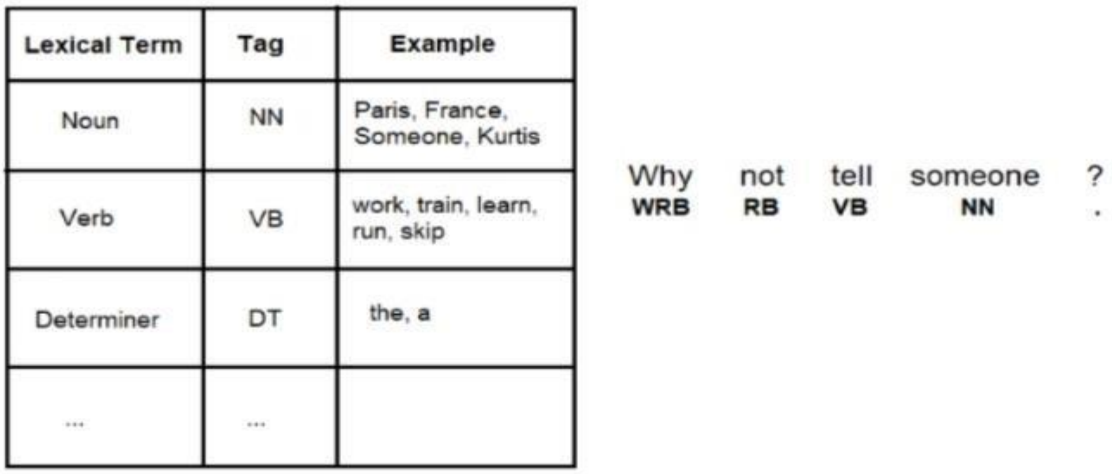


Figure 1. Example of POS tagging

In Figure 1, we can see each word has its own lexical term written underneath, however, having to constantly write out these full terms when we perform text analysis can very quickly become cumbersome especially as the size of the corpus grows. Thence, we use a short representation referred to as "tags" to represent the categories.

As earlier mentioned, the process of assigning a specific tag to a word in our corpus is referred to as part-of-speech tagging (POS tagging for short) since the POS tags are used to describe the lexical terms that we have within our text.



Lexical Term	Tag	Example
Noun	NN	Paris, France, Someone, Kurtis
Verb	VB	work, train, learn, run, skip
Determiner	DT	the, a
...	...	

Why	not	tell	someone	?
WRB	RB	VB	NN	.

Figure 2. Grid displaying types of lexical terms,tags

Part-of-speech tags describe the characteristic structure of lexical terms within a sentence or text, therefore, we can use them for making assumptions about semantics. Other applications of POS tagging include:

- **Named Entity Recognition**
- **Co-reference Resolution**
- **Speech Recognition**

When we perform POS tagging, it’s often the case that our tagger will encounter words that were not within the vocabulary that was used. Consequently, augmenting your dataset to include unknown word tokens. Will aid the tagger in selecting appropriate tags for those words.

HMM (Hidden Markov Model)

Taking the example text we used in Figure 1, “Why not tell someone?”, imaging the sentence is truncated to “Why not tell…” and we want to determine whether the following word in the sentence is a noun, verb, adverb, or some other part-of-speech.

Now, if you are familiar with English, you’d instantly identify the verb and assume that it is more likely the word is followed by a noun rather than another verb. Therefore, the idea as

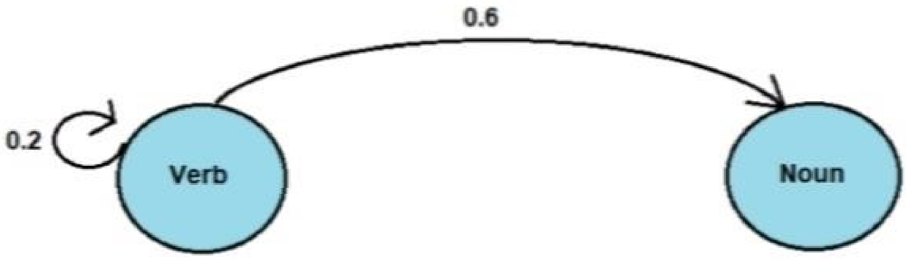


Figure 3. Representing Likelihoods visually

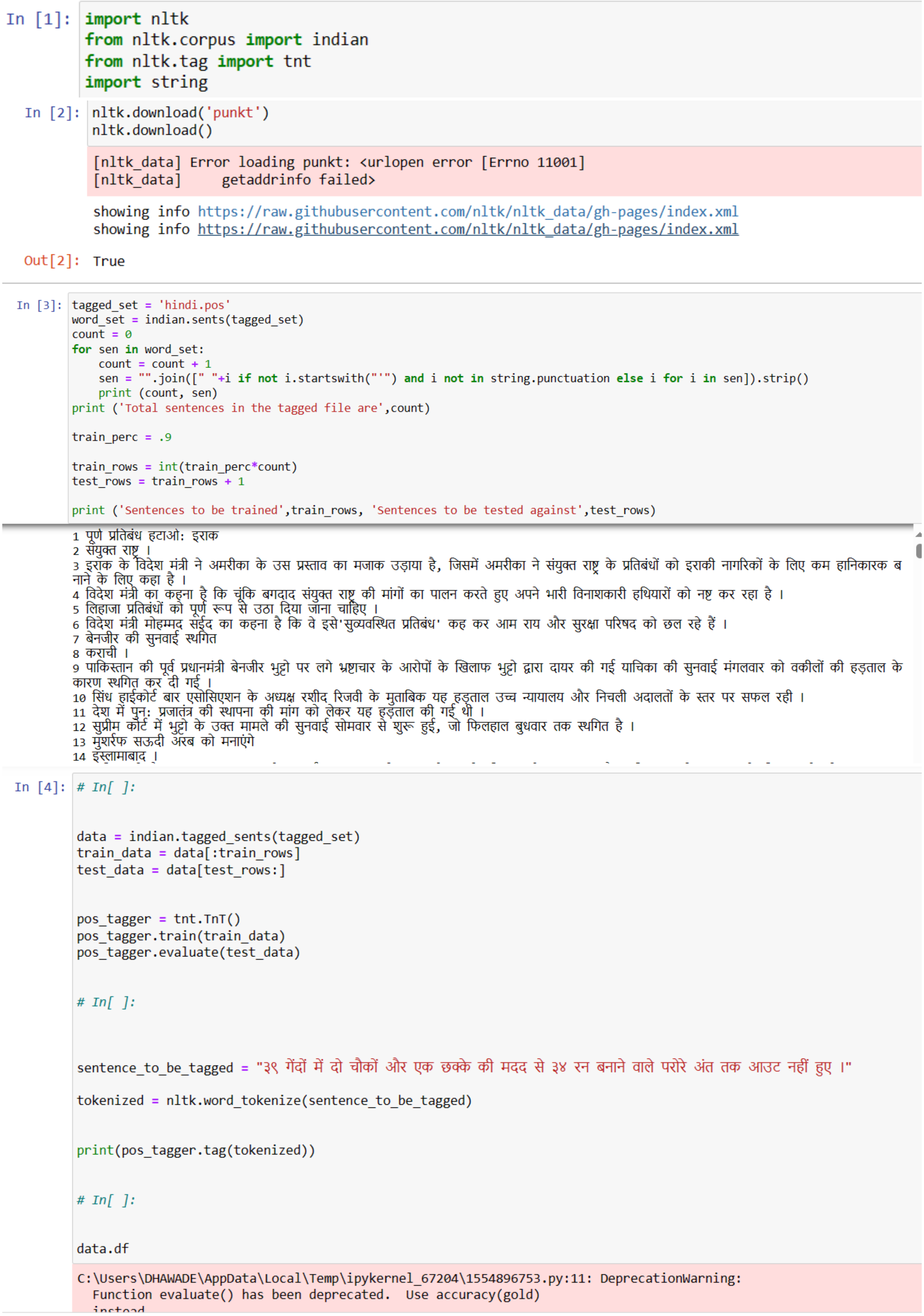
shown in this example is that the POS tag that is assigned to the next word is dependent on the POS tag of the previous word.

By associating numbers with each arrow direction, of which imply the likelihood of the next word given the current word, we can say there is a higher likelihood the next word in our sentence would be a noun since it has a higher likelihood than the next word being a verb if we are currently on a verb. The image in Figure 3, is a great example of how a Markov Model works on a very small scale.

Given this example, we may now describe Markov models as "a stochastic model used to model randomly changing systems. It is assumed that future states depend only on the current state, not on the events that occurred before it (that is, it assumes the Markov property).".
Therefore, to get the probability of the next event, it needs only the states of the current event.

We could use this Markov model to perform POS. Considering we view a sentence as a sequence of words, we can represent the sequence as a graph where we use the POS tags as the events that occur which would be illustrated by the stats of our model graph.

Implementation



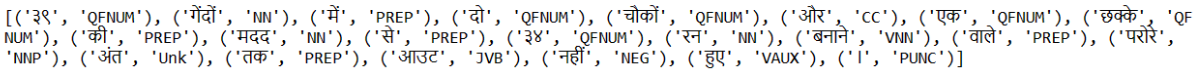
```
In [1]: import nltk
        from nltk.corpus import indian
        from nltk.tag import tnt
        import string
```

```
In [2]: nltk.download('punkt')
        nltk.download()

        [nltk_data] Error loading punkt: <urlopen error [Errno 11001]
        [nltk_data]     getaddrinfo failed>

        showing info https://raw.githubusercontent.com/nltk/nltk_data/gh-pages/index.xml
        showing info https://raw.githubusercontent.com/nltk/nltk_data/gh-pages/index.xml

Out[2]: True
```

```
In [3]: tagged_set = 'hindi.pos'
        word_set = indian.sents(tagged_set)
        count = 0
        for sen in word_set:
            count = count + 1
            sen = "".join([" "+i if not i.startswith("'") and i not in string.punctuation else i for i in sen]).strip()
            print (count, sen)
        print ('Total sentences in the tagged file are',count)

        train_perc = .9

        train_rows = int(train_perc*count)
        test_rows = train_rows + 1

        print ('Sentences to be trained',train_rows, 'Sentences to be tested against',test_rows)
```

```
1 पूर्ण प्रतिबंध हटाओ: इराक
2 संयुक्त राष्ट्र ।
3 इराक के विदेश मंत्री ने अमरीका के उस प्रस्ताव का मजाक उड़ाया है, जिसमें अमरीका ने संयुक्त राष्ट्र के प्रतिबंधों को इराकी नागरिकों के लिए कम हानिकारक ब
नाने के लिए कहा है ।
4 विदेश मंत्री का कहना है कि चूंकि बगदाद संयुक्त राष्ट्र की मांगों का पालन करते हुए अपने भारी विनाशकारी हथियारों को नष्ट कर रहा है ।
5 लिहाजा प्रतिबंधों को पूर्ण रूप से उठा दिया जाना चाहिए ।
6 विदेश मंत्री मोहम्मद सईद का कहना है कि वे इसे'सुव्यवस्थित प्रतिबंध' कह कर आम राय और सुरक्षा परिषद को छल रहे हैं ।
7 बेनजीर की सुनवाई स्थगित
8 कराची ।
9 पाकिस्तान की पूर्व प्रधानमंत्री बेनजीर भुट्टो पर लगे भ्रष्टाचार के आरोपों के खिलाफ भुट्टो द्वारा दायर की गई याचिका की सुनवाई मंगलवार को वकीलों की हड़ताल के
कारण स्थगित कर दी गई ।
10 सिंध हाईकोर्ट बार एसोसिएशन के अध्यक्ष रशीद रिजवी के मुताबिक यह हड़ताल उच्च न्यायालय और निचली अदालतों के स्तर पर सफल रही ।
11 देश में पुन: प्रजातंत्र की स्थापना की मांग को लेकर यह हड़ताल की गई थी ।
12 सुप्रीम कोर्ट में भुट्टो के उक्त मामले की सुनवाई सोमवार से शुरू हुई, जो फिलहाल बुधवार तक स्थगित है ।
13 मुशर्रफ सऊदी अरब को मनाएंगे
14 इस्लामाबाद ।
```

```
In [4]: # In[ ]:


        data = indian.tagged_sents(tagged_set)
        train_data = data[:train_rows]
        test_data = data[test_rows:]


        pos_tagger = tnt.TnT()
        pos_tagger.train(train_data)
        pos_tagger.evaluate(test_data)


        # In[ ]:



        sentence_to_be_tagged = "३९ गेंदों में दो चौकों और एक छक्के की मदद से ३४ रन बनाने वाले परोरे अंत तक आउट नहीं हुए ।"

        tokenized = nltk.word_tokenize(sentence_to_be_tagged)


        print(pos_tagger.tag(tokenized))


        # In[ ]:


        data.df

        C:\Users\DHAWADE\AppData\Local\Temp\ipykernel_67204\1554896753.py:11: DeprecationWarning:
          Function evaluate() has been deprecated.  Use accuracy(gold)
          instead
```



```
[('३९', 'QFNUM'), ('गेंदों', 'NN'), ('में', 'PREP'), ('दो', 'QFNUM'), ('चौकों', 'QFNUM'), ('और', 'CC'), ('एक', 'QFNUM'), ('छक्के', 'QF
NUM'), ('की', 'PREP'), ('मदद', 'NN'), ('से', 'PREP'), ('३४', 'QFNUM'), ('रन', 'NN'), ('बनाने', 'VNN'), ('वाले', 'PREP'), ('परोरे',
'NNP'), ('अंत', 'Unk'), ('तक', 'PREP'), ('आउट', 'JVB'), ('नहीं', 'NEG'), ('हुए', 'VAUX'), ('।', 'PUNC')]
```

CONCLUSION:

In this, we have implemented POS tagging for simple sentences written in Hindi Language or any other Indian Language.