JUDICIAL NER(Hindi)

Mentor - Pruthwik Mishra

Souvik Banerjee
Priyank Modi

Technologies and Resources used

- Python
- Bash
- LTRC shallow parser
- ISC tagger, parser
- Data on web for building libraries

Project Requirements

1. Analyzing the data to find the distribution of named entities in the data (frequency).
2. Creating a set of rules to identify the named entities
3. Checking and plotting the frequency of features designed.
4. Designing a date, time recognizer using a set of rules.

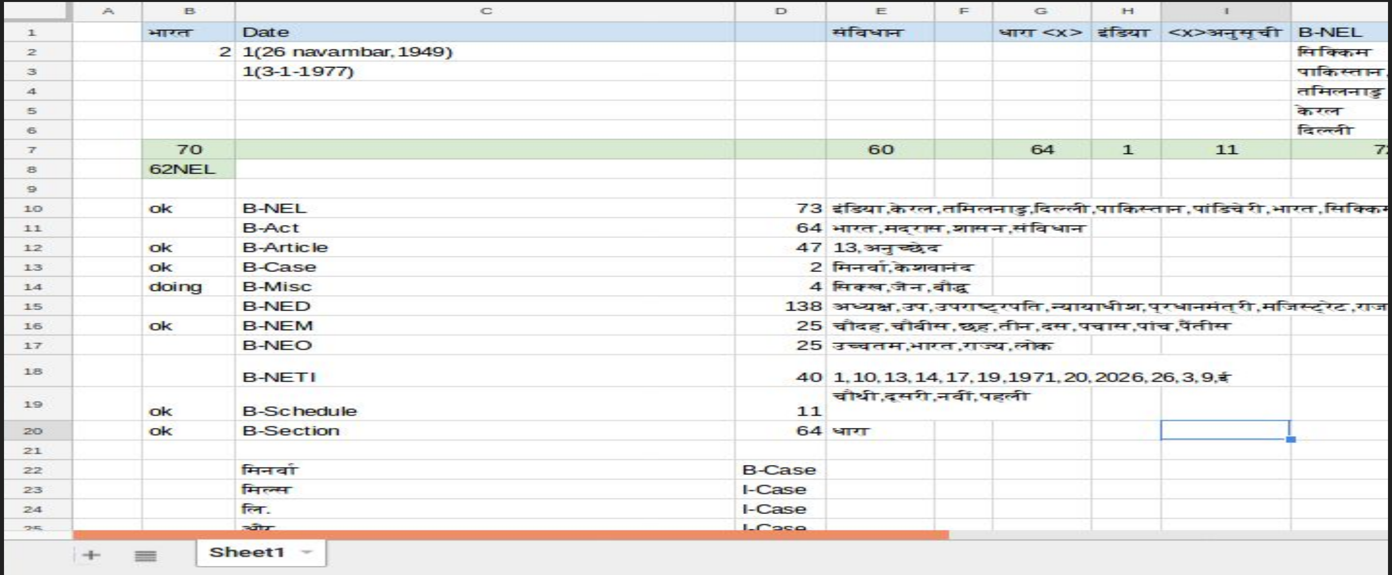
Insight into project difficulties

- Hindi data has no capitalization
- Judicial data has specific format, not captured by taggers
- Careful use of parsers because of Free word order in Hindi
- Handling multi-word NEs in Judicial data

WORKFLOW

Analysis of Training data

We first analysed Judicial documents and using Bash scripts, extracted out the frequencies of each class of Named Entity.



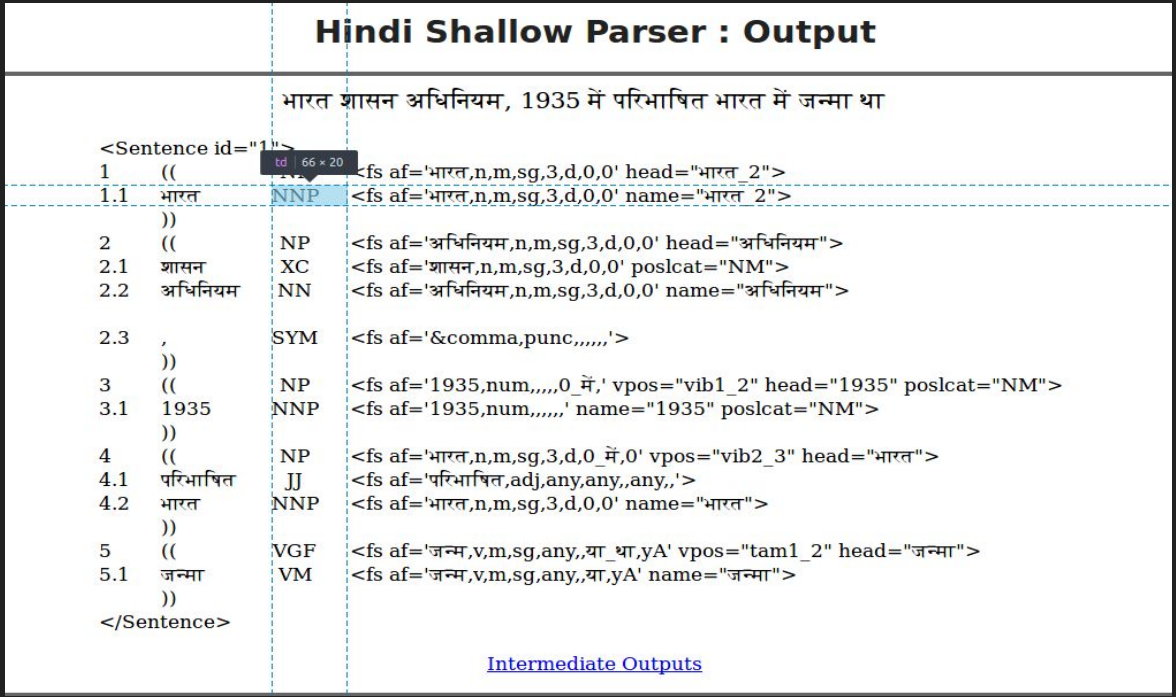
	A	B	C	D	E	F	G	H	I	
1		भारत	Date		संविधान		धारा <x>	इंडिया	<x>अनुसूची	B-NEL
2		2	1(26 navambar,1949)							सिक्किम
3			1(3-1-1977)							पाकिस्तान,
4										तमिलनाडु
5										केरल
6										दिल्ली
7		70			60		64	1	11	7
8		62NEL								
9										
10		ok	B-NEL	73	इंडिया,केरल,तमिलनाडु,दिल्ली,पाकिस्तान,पांडिचेरी,भारत,सिक्किम					
11			B-Act	64	भारत,मद्रास,शासन,संविधान					
12		ok	B-Article	47	13,अनुच्छेद					
13		ok	B-Case	2	मिनर्वा,केशवानंद					
14		doing	B-Misc	4	सिक्ख,जैन,बौद्ध					
15			B-NED	138	अध्यक्ष,उप,उपराष्ट्रपति,न्यायाधीश,प्रधानमंत्री,मजिस्ट्रेट,राज					
16		ok	B-NEM	25	चौदह,चौबीस,छह,तीन,दस,पचास,पांच,पैंतीस					
17			B-NEO	25	उच्चतम,भारत,राज्य,लोक					
18			B-NETI	40	1,10,13,14,17,19,1971,20,2026,26,3,9,ई					
19		ok	B-Schedule	11	चौथी,दूसरी,नवीं,पहली					
20		ok	B-Section	64	धारा					
21										
22			मिनर्वा	B-Case						
23			मिल्स	I-Case						
24			लि.	I-Case						
25			और	I-Case						

Sheet1

Tokenizing data

- Needed to convert hindi literals into computer readable format
- Used python’s utf-encoding
- Also used isc-parser

Tagging data



Hindi Shallow Parser : Output

भारत शासन अधिनियम, 1935 में परिभाषित भारत में जन्मा था

```
<Sentence id="1">
1       ((          NP    <fs af='भारत,n,m,sg,3,d,0,0' head="भारत_2">
1.1     भारत        NNP   <fs af='भारत,n,m,sg,3,d,0,0' name="भारत_2">
        ))
2       ((          NP    <fs af='अधिनियम,n,m,sg,3,d,0,0' head="अधिनियम">
2.1     शासन        XC    <fs af='शासन,n,m,sg,3,d,0,0' poslcat="NM">
2.2     अधिनियम     NN    <fs af='अधिनियम,n,m,sg,3,d,0,0' name="अधिनियम">

2.3     ,           SYM   <fs af='&comma,punc,,,,,,'>
        ))
3       ((          NP    <fs af='1935,num,,,,,0_में,' vpos="vib1_2" head="1935" poslcat="NM">
3.1     1935        NNP   <fs af='1935,num,,,,,,' name="1935" poslcat="NM">
        ))
4       ((          NP    <fs af='भारत,n,m,sg,3,d,0_में,0' vpos="vib2_3" head="भारत">
4.1     परिभाषित     JJ    <fs af='परिभाषित,adj,any,any,,any,,'>
4.2     भारत        NNP   <fs af='भारत,n,m,sg,3,d,0,0' name="भारत">
        ))
5       ((          VGF   <fs af='जन्म,v,m,sg,any,,या_था,yA' vpos="tam1_2" head="जन्मा">
5.1     जन्मा        VM    <fs af='जन्म,v,m,sg,any,,या,yA' name="जन्मा">
        ))
</Sentence>
```

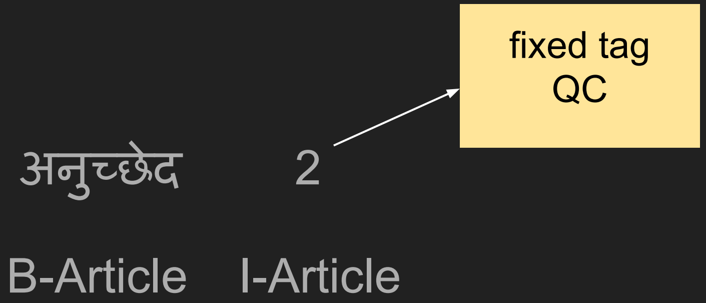
Intermediate Outputs

Making dictionaries

- Studied judicial documents
- Scraped the web to form dictionaries
- Used the given dictionaries

Forming rules

The most important part was to form the rules. We used dictionaries to compare the NE, defined a scope for the length of an NE based on the tags and dependencies of it’s neighbours and found relations between the tags and the dependencies.



fixed tag
QC

अनुच्छेद 2

B-Article I-Article

Frequency charts

The final part of the project was to assign labels to each word and calculate the frequency of the different types of NEs against each other

ROADBLOCKS

Rule clashing

...यथा अधिनियमित) भारत शासन अधिनियम , 1935 में परिभाषित भारत में जन्मा था…..

B-Case

हम, भारत के लोग, भारत को एक संपूर्ण...

B-NEL

Issues with parsing data(dependencies)

"...19 जुलाई, 1948 के पश्चात् प्रव्रजन किया है, 12-6-2014............"

```
1   19   19   NNPC   NNPC   _   3   pof__cn   _   _

2   जुलाई,   जुलाई,   NNPC   NNPC   _   3   pof__cn   _   _

3   1948   1948   NNP   NNP   _   7   k7t   _   _

4   के   के   PSP   PSP   _   3   lwg__psp   _   _

5   पश्चात्   पश्चात्   NST   NST   _   3   lwg__psp   _   _

6   प्रव्रजन   प्रव्रजन   NN   NN   _   7   pof   _   _

7   किया   किया   VM   VM   _   0   main   _   _

8   है,   है,   VAUX   VAUX   _   7   lwg__vaux   _   _

9   12-6-2014   12-6-2014   QC   QC   _   0   nmod__adj   _
_
```

Very long sentences with often with no action verb. The dependency data more often than not gives no meaningful information

Doesn't work well

Multiword Named Entities

“........., मिनर्वा मिल्स लि. और अन्य बनाम भारत संघ और अन्य (1980) 2 एससीसी 591 में अविधिमान्य घोषित कर दिया गया । ”

Also a B-NEL

Date/Time recognition

Consider the 2 phrases, the first of which has a date/time reference but the second not. However both will be assigned the same pos-tag.

1948 में

1900 लोग

Date/Time recognition

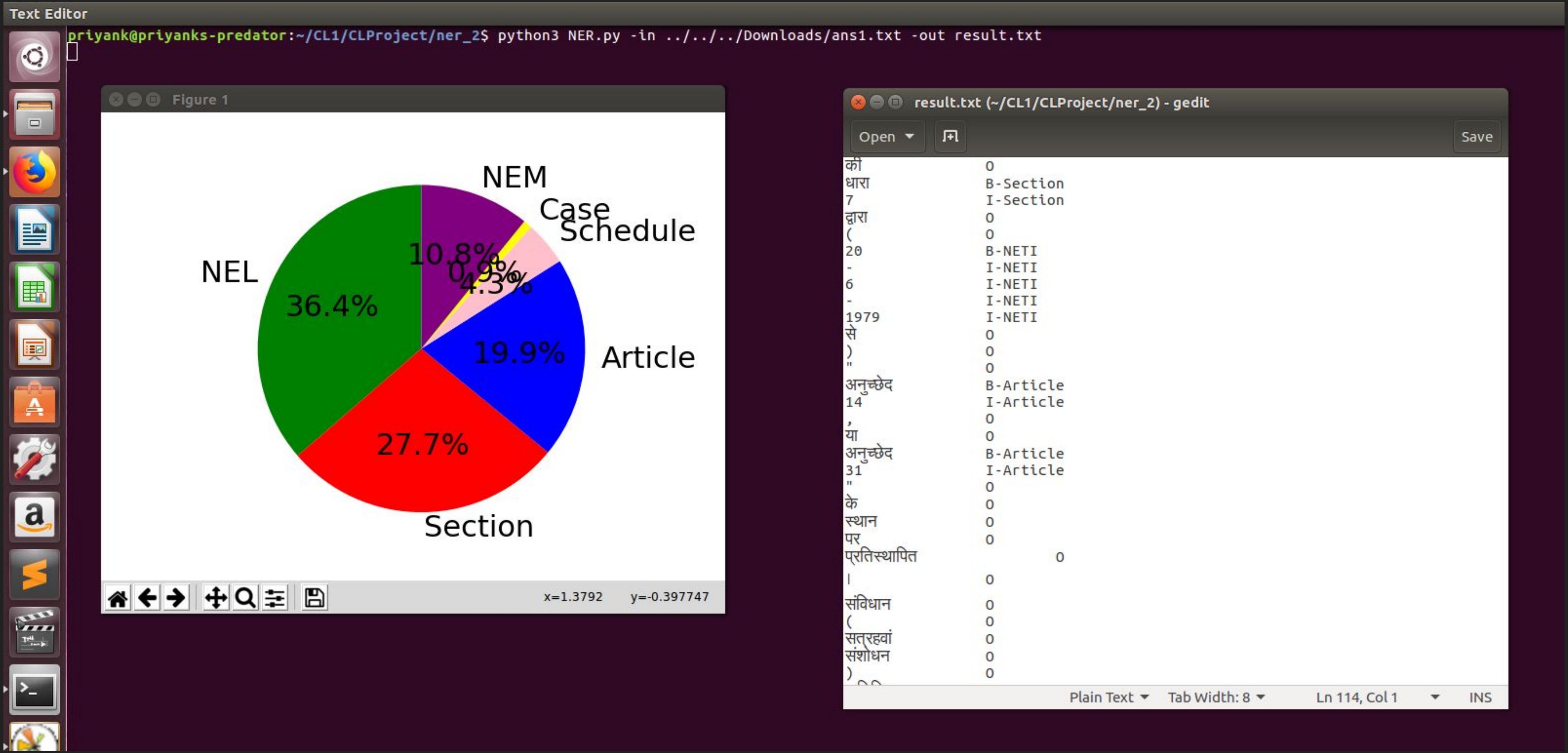
“...19 जुलाई, 1948 के पश्चात् प्रव्रजन किया है, 12-6-2014............”

```
1  19  19  NNPC  NNPC  _  3  pof__cn  _  _
2  जुलाई,  जुलाई,  NNPC  NNPC  _  3  pof__cn  _  _
3  1948  1948  NNP  NNP  _  7  k7t  _  _
4  के  के  PSP  PSP  _  3  lwg__psp  _  _
5  पश्चात्  पश्चात्  NST  NST  _  3  lwg__psp  _  _
6  प्रव्रजन  प्रव्रजन  NN  NN  _  7  pof  _  _
7  किया  किया  VM  VM  _  0  main  _  _
8  है,  है,  VAUX  VAUX  _  7  lwg__vaux  _  _
9  12-6-2014  12-6-2014  QC  QC  _  0  nmod__adj  _  _
```

Works well

Doesn’t work well

OUTPUT



```
priyank@priyanks-predator:~/CL1/CLProject/ner_2$ python3 NER.py -in ../../../Downloads/ans1.txt -out result.txt
```

```
की              O
धारा            B-Section
7               I-Section
द्वारा           O
(               O
20              B-NETI
-               I-NETI
6               I-NETI
-               I-NETI
1979            I-NETI
से              O
)               O
"               O
अनुच्छेद         B-Article
14              I-Article
,               O
या              O
अनुच्छेद         B-Article
31              I-Article
"               O
के              O
स्थान           O
पर              O
प्रतिस्थापित                O
।               O

संविधान         O
(               O
सत्रहवां         O
संशोधन          O
)               O
```

Testing

For testing our outputs, we used a second training file, removed the tags from it, converted it into a normal testing file.

We then ran a difference test and checked the percentage match.

We achieved 99% match for the file we tested.

Limitations

- Assumes tagging output is correct
- Relies on parsing output to be correct
- Rules are only as exhaustive as the training data
- Not very efficient, uses around O(n^3) but it was made faster by efficiently using the isc tagger and parser which does the work very quickly as it uses ML.

Future Work

- Testing the software against more Judicial data
- Using a Brill’s tagger based approach to automate rule formation
- Increasing efficiency by using Machine Learning

THANK YOU