CS563 Natural Language Processing: Assignment 2

Named Entity Recognition using Hidden Markov Model.

Team Details

Team Code: `1801cs15_1801cs46`

Team Name: `kacha_badam`

Team Members

Name	Roll Number
Bhumika Shivani	1801CS15
Shashwat Mahajan	1801CS46

Set up and Execution

Creating environment with the required packages

Use the `requirements.txt` file to install the required packages. The environment name here is `nlp-a2`. You may change it if you want.

```
conda create --name nlp-a2 --file requirements.txt
```

Activate the new environment

```
conda activate nlp-a2
```

Running the Code

```
python3 main.py
```

Architecture

The model comprises of calculating the transition and the emission matrices.

Transition Matrix

Bigram Case

To estimate `A[i][j]`, where `A[i][j]` denotes the probability of tag `j` succeeding tag `i`.

```
A[i][j] = freq(i, j) / freq(i)
```

Trigram Case

To estimate `A[i][j][k]`, where `A[i][j][k]` denotes the probability of tag `k` succeeding tag `j` succeeding tag `i`.

```
A[i][j] = freq(i, j, k) / freq(i, j)
```

Emission Matrix

Without Context Case

To estimate `B[i][j]`, where `B[i][j]` denotes the probability of word/token `i` emitting from tag `j`.

```
B[i][j] = freq(i, j) / freq(j)
```

With Context Case

To estimate `B[i][j][k]`, where `B[i][j][k]` denotes the probability of word/token `i` emitting from tag `j` preceded by tag `k`.

```
B[i][j][k] = freq(i, j, k) / freq(j, k)
```

These parameters being obtained, our model is now trained for usage on test cases.

The Viterbi Algorithm

For evaluation, we use the Viterbi algorithm. The algorithm uses Dynamic Programming to estimate the probability of most probable sequence of tags and then reconstructs the corresponding sequence using stored data.

Error Analysis from Output

HMM for bigram model without context

Evaluation on bigram model without context *Evaluated 3850 sentences.*
Time taken: *00:07 min (514.24it/s)*

HMM Model Accuracy = 0.9111746462492731

> Class-wise Accuracies

Class	Precision	Recall	F1
O	0.926372	0.991654	0.957902
company	0.673575	0.146727	0.240964
facility	0.402778	0.0936995	0.152031
loc	0.606327	0.313351	0.413174
movie	0.0322581	0.0121951	0.0176991
musicartist	0.340909	0.0453172	0.08
other	0.404167	0.170175	0.239506
person	0.483776	0.209719	0.292596
product	0.503876	0.0871314	0.148571
sportsteam	0.24	0.0923077	0.133333
tvshow	0	0	0

HMM for trigram model without context

Evaluation on trigram model without context *Evaluated 3850 sentences.*
Time taken: *01:29 min (42.95it/s)*

HMM Model Accuracy = 0.9113361762615494

> Class-wise Accuracies

Class	Precision	Recall	F1
O	0.927422	0.990456	0.957903
company	0.654822	0.145598	0.238227
facility	0.388889	0.124394	0.188494
loc	0.621429	0.316076	0.419025
movie	0.027027	0.0121951	0.0168067
musicartist	0.3	0.0453172	0.0787402

Class	Precision	Recall	F1
other	0.385965	0.173684	0.239564
person	0.497238	0.230179	0.314685
product	0.5	0.0938338	0.158014
sportsteam	0.225	0.0923077	0.130909
tvshow	0	0	0

HMM for bigram model with context

Evaluation on bigram model with context *Evaluated 3850 sentences.*
***Time taken**: 00:09 min (409.44it/s)*

HMM Model Accuracy = 0.911917684305744

> Class-wise Accuracies

Class	Precision	Recall	F1
O	0.924102	0.992279	0.956978
company	0.567164	0.0857788	0.14902
facility	0.503448	0.117932	0.191099
loc	0.6	0.288828	0.389945
movie	0	0	0
musicartist	0.25	0.0241692	0.0440771
other	0.40592	0.168421	0.238066
person	0.509494	0.205882	0.29326
product	0.54902	0.075067	0.132075
sportsteam	0.194444	0.0717949	0.104869
tvshow	0	0	0

HMM for trigram model with context

Evaluation on trigram model with context *Evaluated 3850 sentences.*
***Time taken**: 01:32 min (41.50it/s)*

HMM Model Accuracy = 0.9122568973315242

Class-wise Accuracies

Class	Precision	Recall	F1
O	0.924797	0.991654	0.957059
company	0.552239	0.0835214	0.145098
facility	0.467337	0.150242	0.227384
loc	0.595376	0.280654	0.381481
movie	0	0	0
musicartist	0.3125	0.0302115	0.0550964
other	0.389662	0.17193	0.238588
person	0.496894	0.204604	0.289855
product	0.612613	0.0911528	0.158693
sportsteam	0.232143	0.0666667	0.103586
tvshow	0	0	0

Comparison of all four models

The trigram models perform better than the bigram ones in their respective application areas. After adding context to the emission probabilities, they again, perform better than their context-less counterparts. The overall accuracies can be compared using the table below.

	Bigram	Trigram
Without Context	0.9111746462492731	0.9113361762615494
With Context	0.911917684305744	0.9122568973315242

Thanking You!

kacha_badam