CS 626
Assignment 2 - Chunking

Team Members:
Saurabh Parekh - 170100016
Inderjeet Nair - 170020013
Manas Jain - 170040068

MEMM: Overview

Three variants tested:

1) **MEMM with Embeddings**: Self-Implemented MEMM with support for including custom features such as word embeddings. Features were binary functions f(t, s) = (t == ‘some_tag’ and s[‘some_history_feature’] == ‘some_val’). Unable to complete training due to insufficient computational resource
2) **Baseline NLTK-MEMM with causal features**: Used the implementation “nltk.classify.MaxentClassifer. Feature functions had similar form.
3) **Final NLTK-MEMM with non-causal features**: Lexical and word features taken from future tokens. **Applied the hypothesis that the chunking tag is very much dependent on the <u>lexical Tag sequence</u> in a window.**

Inference was done using Viterbi: $P(T \mid S) = P(t_1|s_1,t_0)P(t_2|s_2,t_1) \ldots P(t_n|s_n,t_{n-1})$

MEMM: Overall Performance

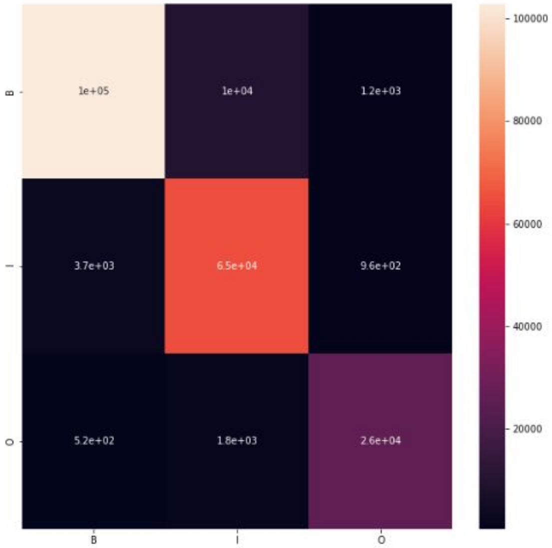
Method	Train Accuracy	Test Accuracy
MEMM with word embeddings*	91.396	91.636
MEMM baseline	96.553	93.828
MEMM Final Model	99.033	96.188

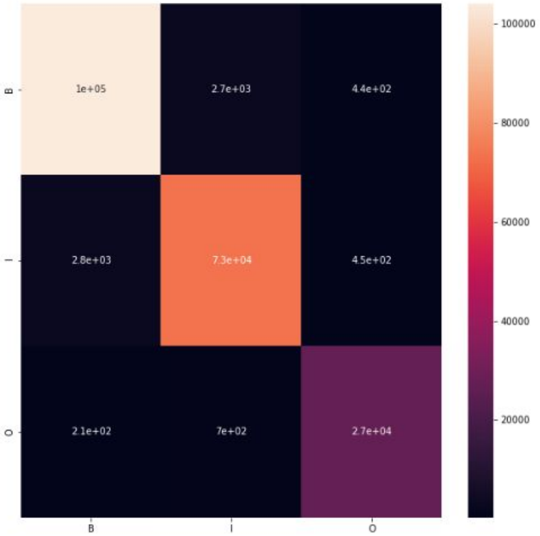
Overall Tag Accuracy
*Was not trained till convergence due to low computational power

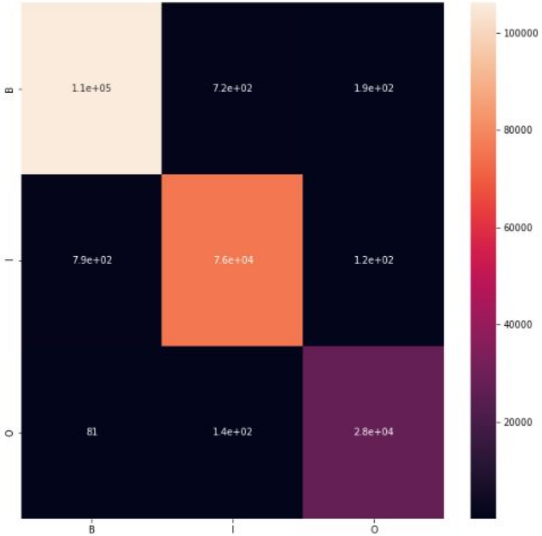
Model	Train accuracy	Test accuracy
MEMM with word embeddings	28.670	28.728
MEMM baseline	53.626	35.636
MEMM Final Model	83.337	52.833

MEMM sentence accuracy

MEMM: Confusion Matrix Train







	B	I	O
B	102772	10009	1249
I	3686	65050	965
O	520	1788	25688

MEMM with word embeddings

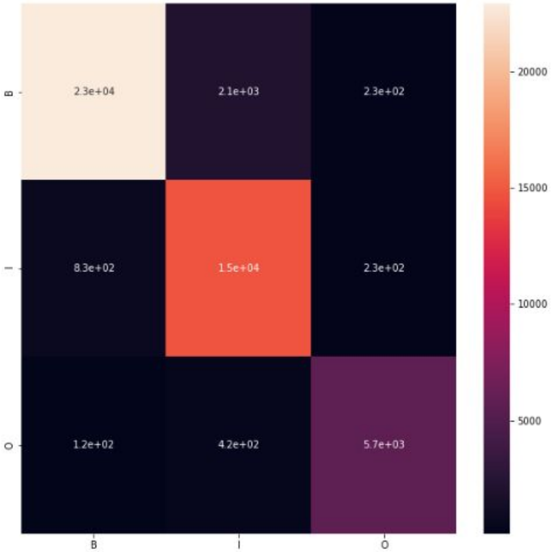
	B	I	O
B	103977	2701	445
I	2794	73444	449
O	207	702	27008

NLTK-MEMM baseline

	B	I	O
B	106110	724	191
I	787	75980	122
O	81	143	27589

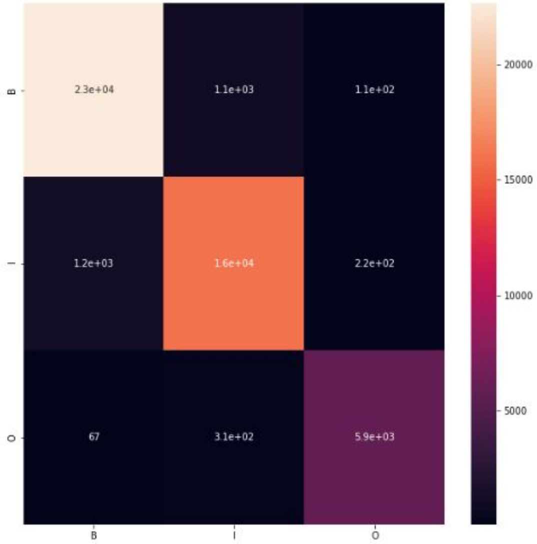
NLTK-MEMM Final Model

MEMM: Confusion Matrix Test



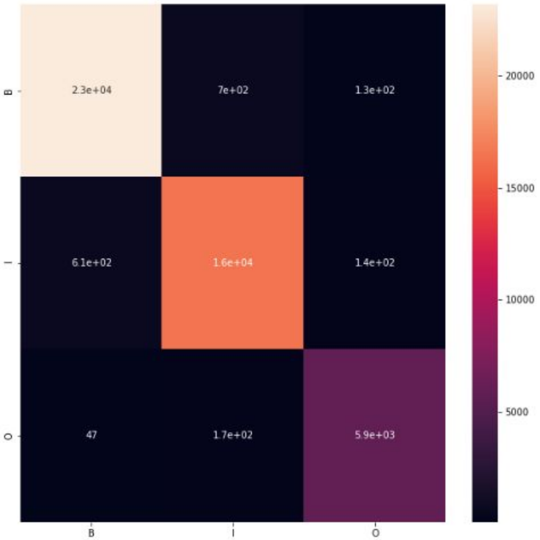
	B	I	O
B	22910	2143	226
I	826	14780	230
O	116	422	5724

MEMM with word embeddings



	B	I	O
B	22623	1059	111
I	1162	15979	218
O	67	307	5851

NLTK-MEMM baseline



	B	I	O
B	23191	697	134
I	614	16475	141
O	47	173	5905

NLTK-MEMM Final Model

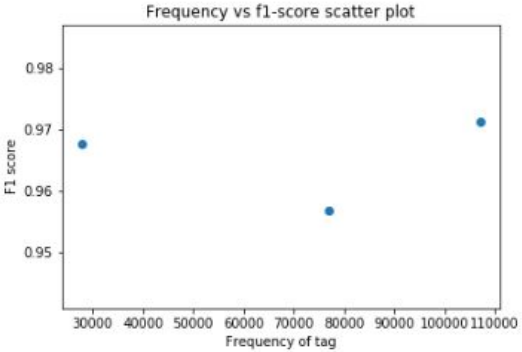
MEMM: Per Chunk Tag Accuracy (Training Domain)

Tag	Precision	Recall	F1-score
B (MEMM with word embeddings)	0.9013	0.9607	0.9300
I (MEMM with word embeddings)	0.9333	0.8465	0.8878
O (MEMM with word embeddings)	0.9176	0.9206	0.9191
B (MEMM baseline)	0.9706	0.9719	0.9713
I (MEMM baseline)	0.9577	0.9557	0.9567
O (MEMM baseline)	0.9674	0.9680	0.9567
B (MEMM Final Model)	0.9914	0.9919	0.9917
I (MEMM Final Model)	0.9882	0.9887	0.9884
O (MEMM Final Model)	0.9919	0.9888	0.9904

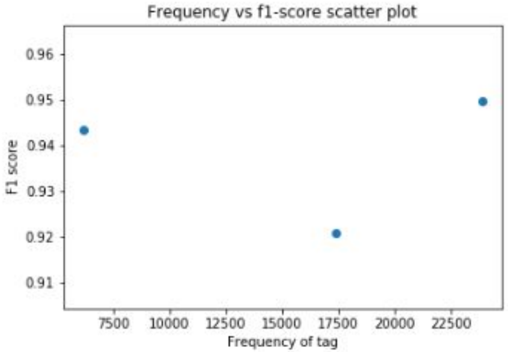
MEMM: Per Chunk Tag Accuracy (Testing Domain)

Tag	Precision	Recall	F1-score
B (MEMM with word embeddings)	0.9063	0.9605	0.9326
I (MEMM with word embeddings)	0.9333	0.8521	0.8909
O (MEMM with word embeddings)	0.9141	0.9262	0.9201
B (MEMM baseline)	0.9508	0.9485	0.9496
I (MEMM baseline)	0.9205	0.9212	0.9208
O (MEMM baseline)	0.9399	0.9468	0.9433
B (MEMM Final Model)	0.9654	0.9723	0.9688
I (MEMM Final Model)	0.9562	0.9498	0.9530
O (MEMM Final Model)	0.9640	0.9555	0.9598

MEMM: Frequency vs Per Chunk Performance

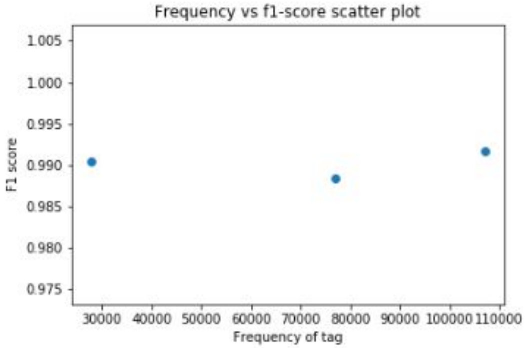


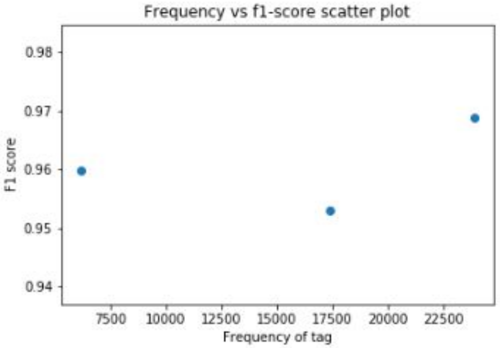
(a) Training domain



(b) Testing domain

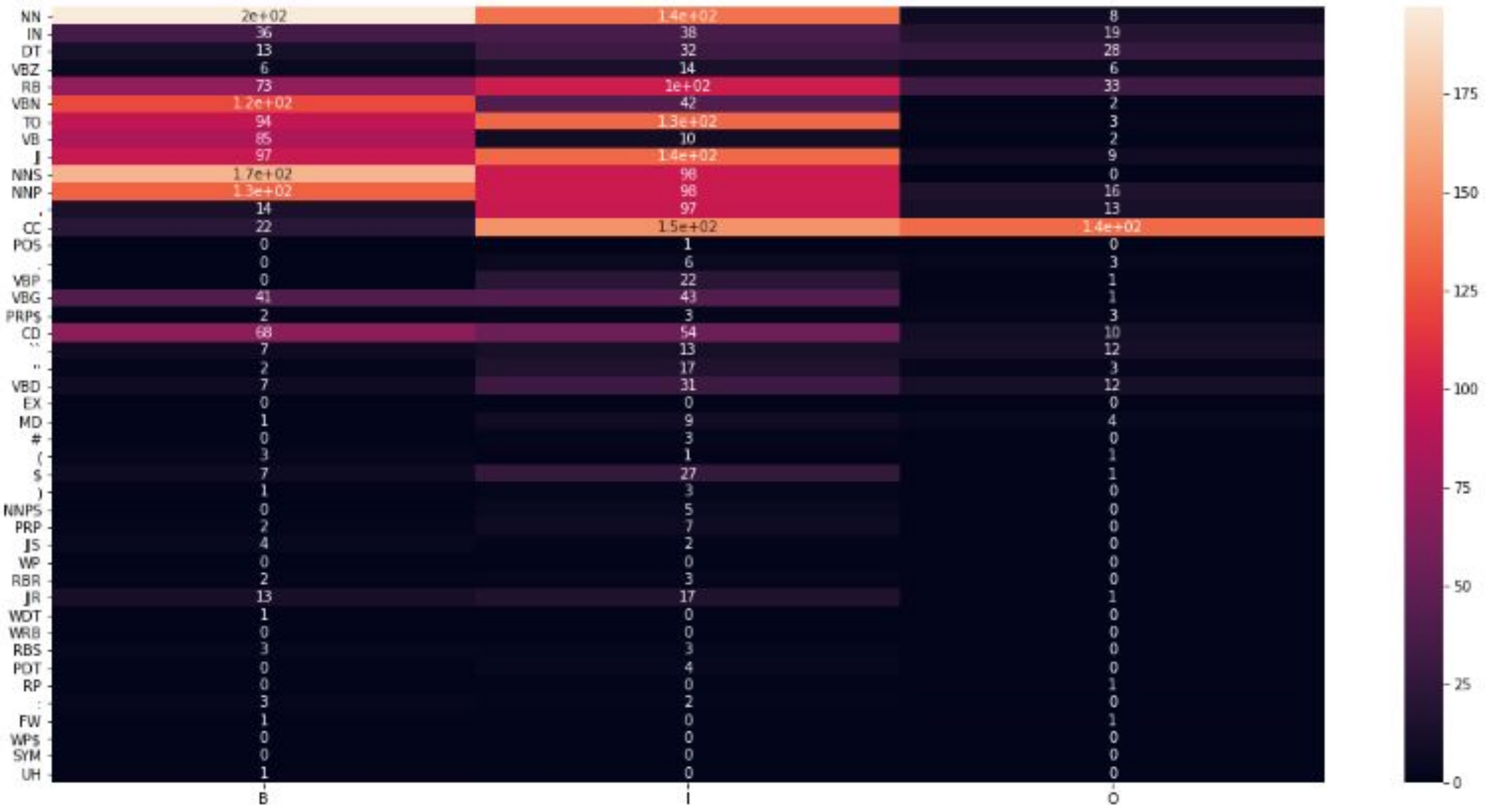
NLTK-MEMM Baseline



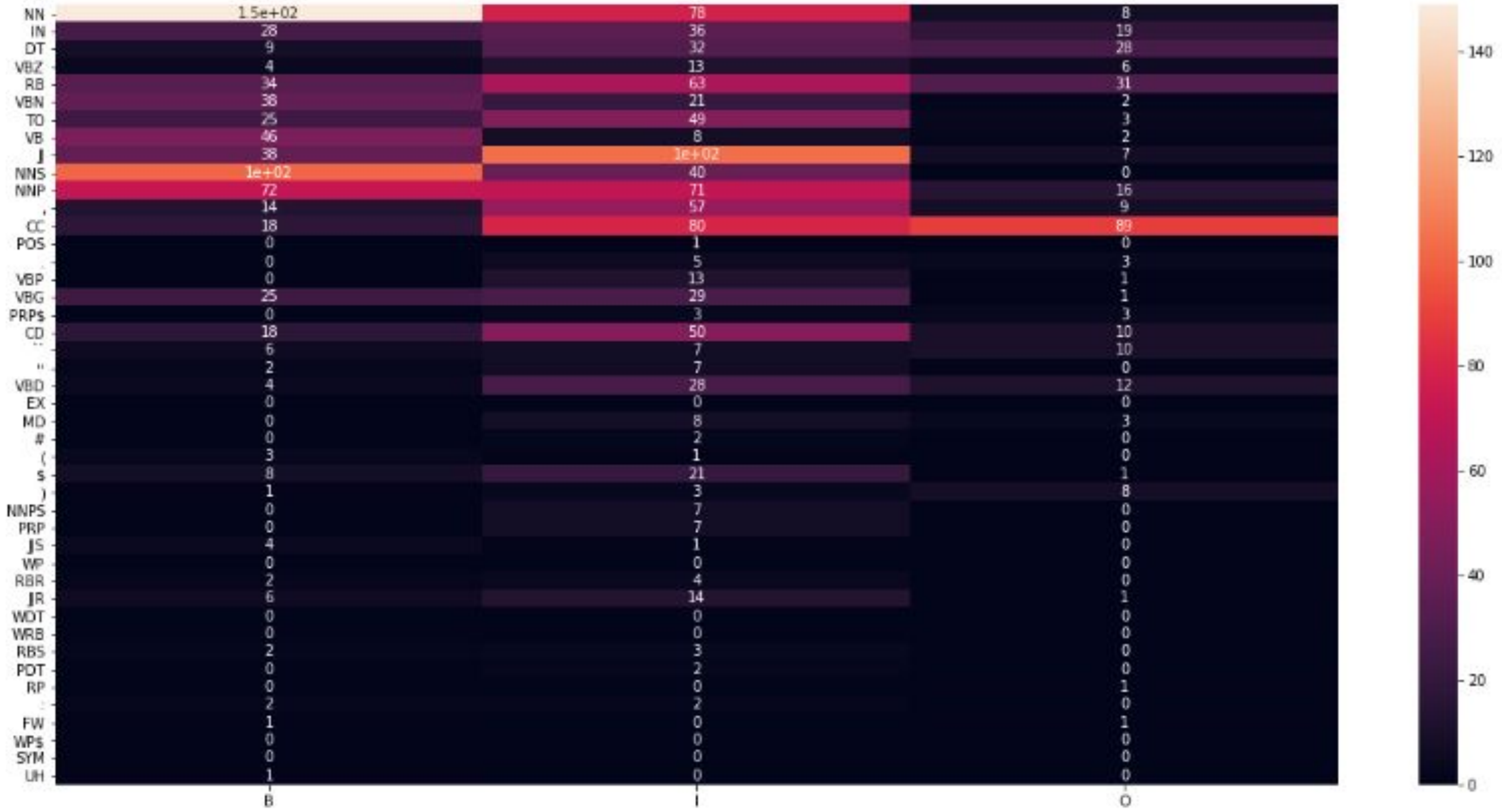


NLTK-MEMM Final Model

MEMM baseline: Mispredictions in every POS category



MEMM Final Model: Mispredictions in every POS category



MEMM: Major Misclassified Categories

Model	True Lexical Categories for Test	True Lexical Categories for Train
MEMM baseline	NN-B, NN-I, RB-I, VBN-B, TO-I, JJ-I, NNS-B, NNP-I, CC-I, CC-O	NN-B, NN-I, IN-I, DT-I, DT-O, RB-B, RB-I, RB-O, VBN-B, TO-B, TO-I, VB-B, JJ-B, JJ-I, NNS-B, NNS-I, NNP-B, NNP-I, ,-I, CC-I, CC-O, VBG-B, VBG-I, CD-B, CD-I
MEMM Final Model	NN-B, NNS-B, JJ-I	NN-B, NNS-B

MEMM: Triumph of Final Model over Baseline

Although overall revenues were stronger, Mr. Schulman said, DEC "drew down its European backlog" and had flat world-wide orders overall

"MEMM baseline" predicted 'flat' as 'I' as it has seen most of the examples of JJ acting as a qualifier for a noun along with the determiner. In usual cases, the determiner becomes the chunk initializer. This sentence did not contain any determiner in front of 'flat' and thus it must be assigned 'B' which was correctly predicted by "MEMM final model".

Jeffrey E. Levin was named vice president and chief economist of this commodity and options exchange.

Consider the chunk residing at the end of the sentence - "this commodity and options exchange". This chunk is associated with the following POS lexical category: "DT NN NNS CC NNS VBP".a "MEMM baseline" predicted the last word "exchange" as "B" because most of the cases of Verb succeeding a Noun does not account to a single chunk. "MEMM final Model" was able to correctly classify the chunk by taking into account the lexical categories of the previous and future words together.

Not so Michigan.

The sentence as a whole is an Adverb phrase. However according to the rules of CoNLP, Adverb phrase with a Noun Phrase as a post modifier must be broken down into two chunks: Adverb Phrase and Noun phrase. The actual chunk to be assigned to this must be 'B I B'. However, the baseline model assigned the sentences: 'O B B'. The final model was correctly able to classify the tags.

MEMM: Some Shortcomings of Final Model

One such company is Bankers Trust Co.

The chunk "One such company" was not marked as a single phrase by the final model. It assigned the following chunking tag: "B B I". It was not able to capture the fact that quantifiers can act as pre-modifiers in a Noun Phrase. This was correctly predicted by the baseline model.

The heavy selling by farmers helped to damp the price rally.

The phrase "helped to damp" is a single chunk. This is misclassified as "B B I" by the final model. This model failed to associate "to damp" with "helped".

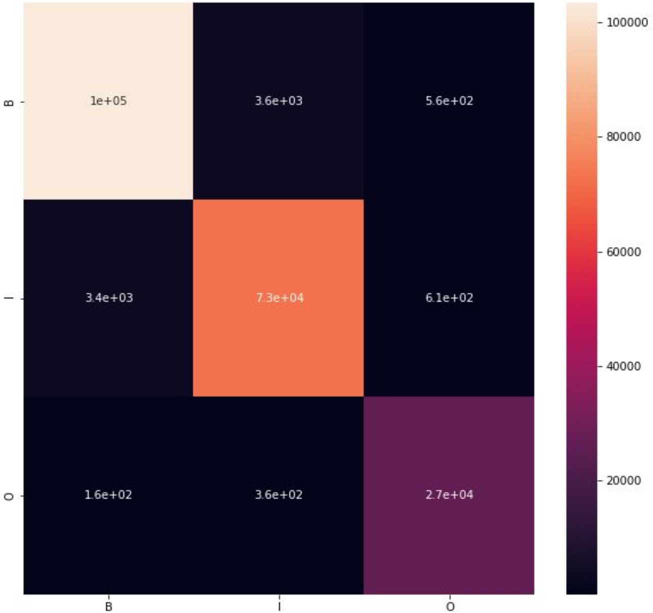
Bi-LSTM: Overall Performance

Number of epochs	Train Accuracy(%)	Test Accuracy(%)
10	85.368	85.685
100	93.245	92.78
200	94.506	93.8066
300	95	94.07
400	95.5577	94.3136
450	95.7789	94.415
500	95.915	94.565
550	96.021	94.5185

Bi-LSTM: Confusion Matrix Train

	B	I	O
B	92801	12662	1543
I	13567	62607	1089
O	527	1512	25249

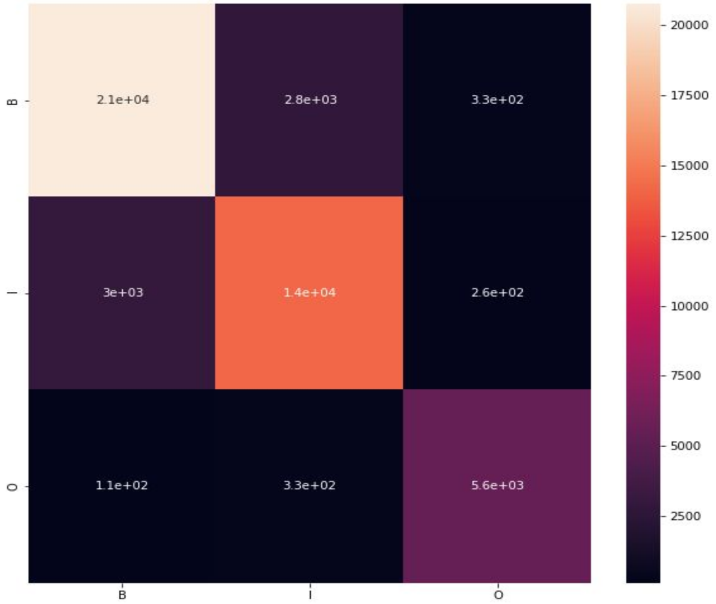
10 epochs



	B	I	O
B	103306	3637	563
I	3434	72788	606
O	155	356	26712

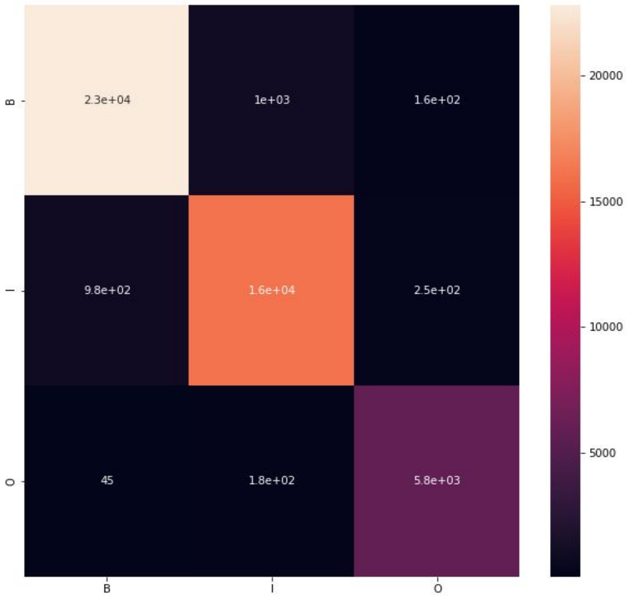
500 epochs

Bi-LSTM: Confusion Matrix Test



	B	I	O
B	20755	2826	327
I	2965	14180	257
O	112	328	5591

10 epochs



	B	I	O
B	22810	1045	160
I	977	16109	247
O	45	180	5768

500 epochs

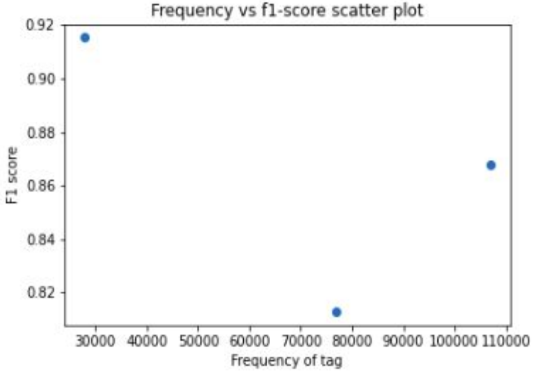
Bi-LSTM: Per Chunk Tag Accuracy (Training Domain)

Tag	Precision	Recall	F1-score
B (Bi-LSTM trained for 10 epochs)	0.8672	0.8681	0.8677
I (Bi-LSTM trained for 10 epochs)	0.8103	0.8154	0.8128
O (Bi-LSTM trained for 10 epochs)	0.9253	0.9056	0.9153
B (Bi-LSTM trained for 500 epochs)	0.9609	0.9664	0.9637
I (Bi-LSTM trained for 500 epochs)	0.9474	0.9479	0.9477
O (Bi-LSTM trained for 500 epochs)	0.9812	0.9581	0.9695

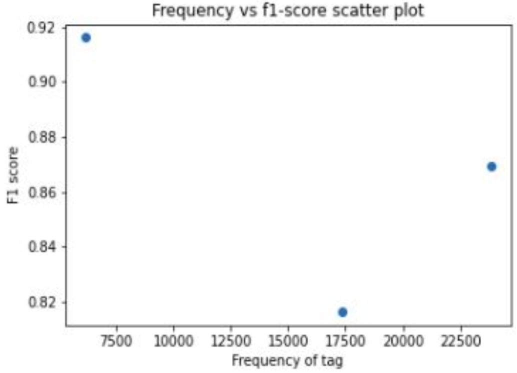
Bi-LSTM: Per Chunk Tag Accuracy (Testing Domain)

Tag	Precision	Recall	F1-score
B (Bi-LSTM trained for 10 epochs)	0.8672	0.8681	0.8677
I (Bi-LSTM trained for 10 epochs)	0.8103	0.8154	0.8128
O (Bi-LSTM trained for 10 epochs)	0.9253	0.9056	0.9153
B (Bi-LSTM trained for 500 epochs)	0.9609	0.9664	0.9637
I (Bi-LSTM trained for 500 epochs)	0.9474	0.9479	0.9477
O (Bi-LSTM trained for 500 epochs)	0.9812	0.9581	0.9695

Bi-LSTM: Frequency vs Per Chunk Performance

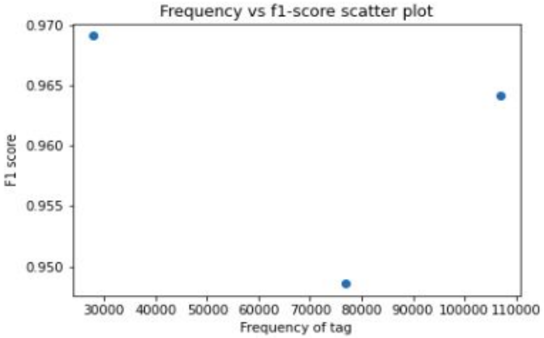


(a) Training domain

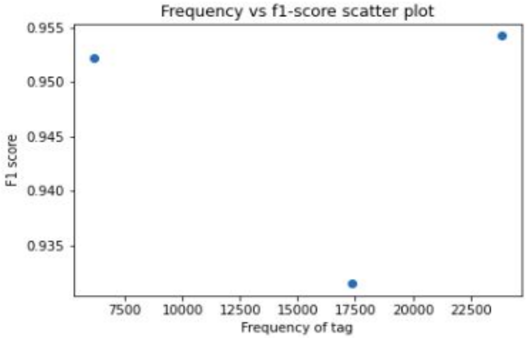


(b) Testing domain

10 epochs



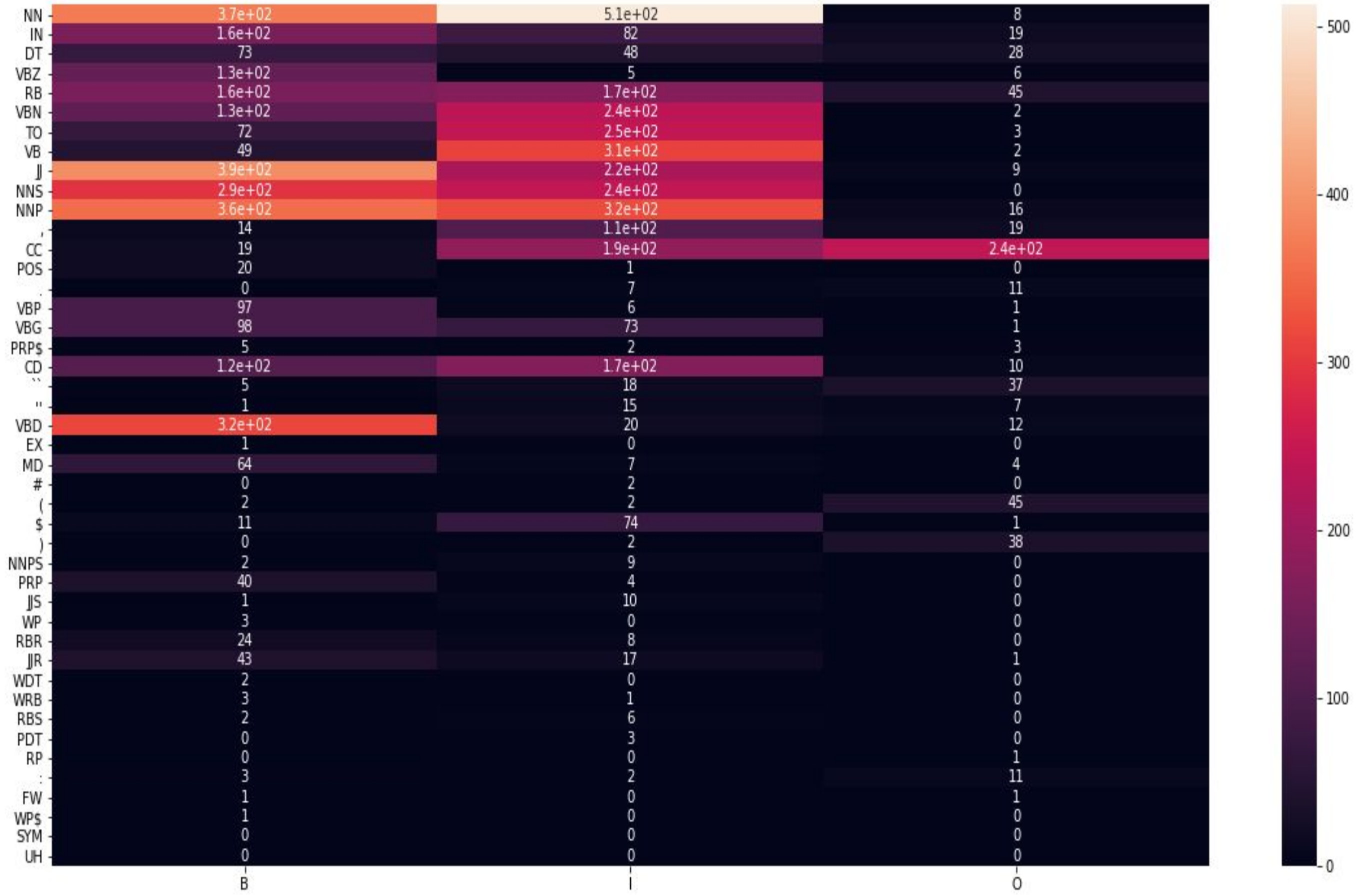
(a) Training domain



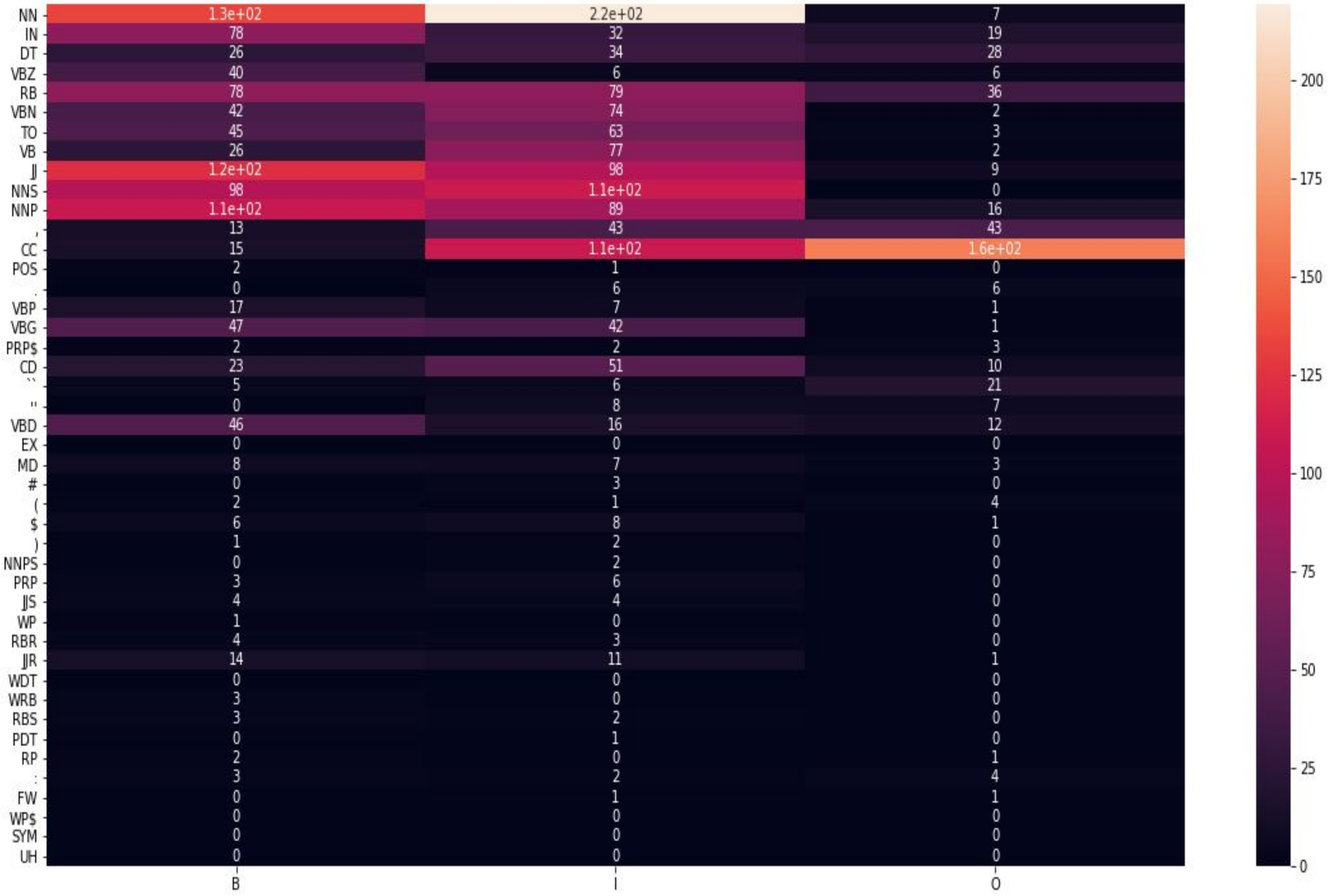
(b) Testing domain

500 epochs

Bi-LSTM 10 epochs: Mispredictions in every POS category



Bi-LSTM 500 epochs: Mispredictions in every POS category



Bi-LSTM: Major Misclassified Categories(> 100)

Model	True Lexical Categories for Test	True Lexical Categories for Train
Bi-LSTM trained for 10 epochs	NN-I, JJ-B, NN-B, NNP-B, NNP-I, VBD-B, VB-I, NNS-B, TO-I, NNS-I, CC-O, VBN-I, JJ-I, CC-I, RB-I, CD-I, RB-B, IN-B, VBZ-B, VBN-B, CD-B, ,-I	NN-B, NN-I, IN-B, IN-I, DT-B, DT-I, DT-O, VBZ-B, RB-B, RB-I, RB-O, VBN-B, VBN-I, TO-B, TO-I, VB-B, VB-I, JJ-B, JJ-I, NNS-B, NNS-I, NNP-B, NNP-I, ,-I, ,-O, CC-I, CC-O, VBP-B, VBG-B, VBG-I, CD-B, CD-I, VBD-B, MD-B, PRP-B, JJR-B
Bi-LSTM trained for 500 epochs	NN-B, NN-I, JJ-B, NNS-I, NNP-B, CC-I, CC-O	NN-B, NN-I, IN-B, IN-I, DT-I, DT-O, VBZ-B, RB-B, RB-I, RB-O, VB-B, VB-I, TO-B, TO-I, VB-B, VB-I, JJ-B, JJ-I, NNS-B, NNS-I, NNP-B, NNP-I, ,-O, CC-I, CC-O, VBG-B, VBG-I, CD-I, VBD-B

Conditional Random Field Model

- Used sklearn-crf_suite for implementation
- Training accuracy - 94.81%
- Iterations = 100
- L-BFGS training algorithm (it is default) with Elastic Net (L1 + L2) regularization (hyperparameters set was 0.1 and 0.1)

Features used (example)

- {'bias': 1.0,
- 'word.lower()': 'confidence',
- 'word[-3:]': 'nce',
- 'word[-2:]': 'ce',
- 'word.isupper()': False,
- 'word.istitle()': True,
- 'word.isdigit()': False,
- 'postag': 'NN',
- 'postag[:2]': 'NN',
- 'BOS': True,
- '+1:word.lower()': 'in',
- '+1:word.istitle()': False,
- '+1:word.isupper()': False,
- '+1:postag': 'IN',
- '+1:postag[:2]': 'IN'}

Results

	precision	recall	f1-score	support
B	0.960	0.954	0.957	23852
I	0.932	0.938	0.935	17345
O	0.948	0.952	0.950	6180
accuracy		0.948		
macro avg	0.947	0.948	0.947	47377
weighted avg	0.948	0.948	0.948	47377

Error analysis

- Top likely transitions:
- O -> O 2.090910
- B -> I 1.964276
- I -> I 1.085829
- O -> B 0.601540
- B -> B 0.369312
- Top unlikely transitions:
- B -> O -0.263557
- I -> B -1.151294
- I -> O -1.825871
- O -> I -12.267853

Error analysis (What actually the model has learned)

- **Top positive:**
- 7.742169 B BOS
- 7.366682 I -1:word.lower():'ve
- 6.476407 I -1:word.lower():trying
- 6.388363 O BOS
- 6.378277 O word.lower():n't
- 6.317444 I -1:word.lower():interbank
- 6.070636 O -1:word.lower():says
- 6.065570 I -1:word.lower():capital-gains
- 5.928229 I -1:word.lower():tens
- 5.524759 I -1:word.lower():vice
- 5.490279 I -1:word.lower():intends
- 5.439327 B -1:word.lower():able
- 5.400217 I -1:word.lower():an
- 4.888253 I word.lower():million
- 4.696904 I -1:word.lower():tend
- 4.686398 I -1:word.lower():because
- 4.658433 I -1:word.lower():refuse
- 4.639044 I -1:word.lower():very

Error analysis (What the model has learned?)

- **Top negative**
- -3.137113 O postag:PRP
- -3.159732 B word.lower():least
- -3.183352 O +1:word.lower():out
- -3.207126 I word.lower():are
- -3.247528 B -1:word.lower():try
- -3.260032 B +1:word.lower():admittedly
- -3.356895 B -1:word.lower():a
- -3.414751 O postag[:2]:VB
- -3.559705 B -1:word.lower():continue
- -3.575742 B -1:word.lower():proving
- -3.714003 O -1:word.lower():patent
- -3.733714 B word.lower():order
- -3.879370 O postag[:2]:NN
- -3.882207 B -1:word.lower():because
- -4.584444 B -1:word.lower():the
- -4.607698 I -1:word.lower():down
- -4.622986 I -1:postag:PRP