Minimally Supervised Semi-Supervised Text Classification

April 17, 2019

Vignesh Viswanathan (16CS10056)
Lovish Chopra (16CS10058)
Sarthak Chakraborty (16CS30044)

Problem Statement

Using Self Attention based RNN to convert documents into semantic vectors

Using Determinantal Point Process to select diverse hypercubes and representative samples

Semantic Vector Conversion

Hypercube based classification

Determinantal Point Process

Semi Supervised Learning

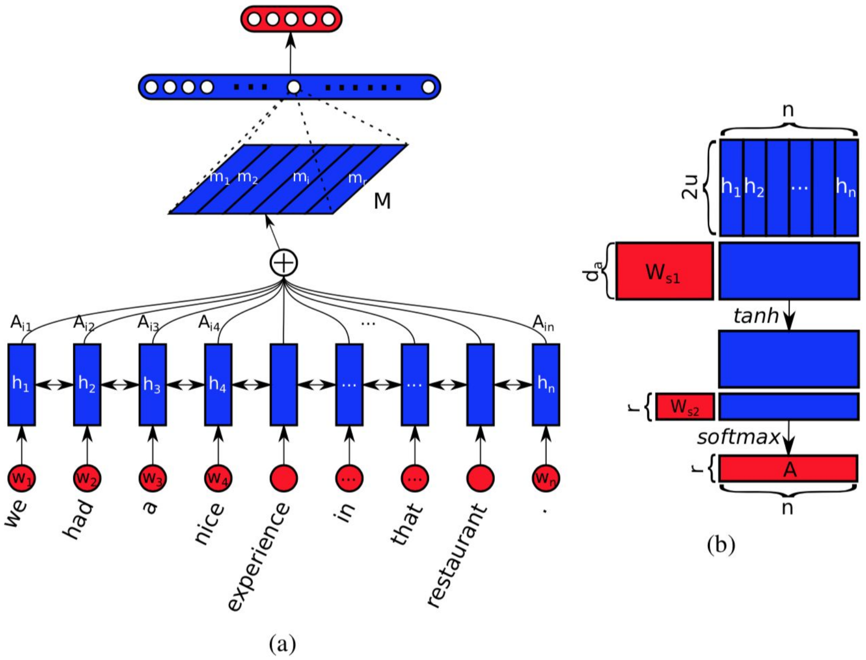
Finding Topology adaptive hypercubes to detect decision boundaries

Use the representatives samples to train the model in a semi-supervised approach

Self Attention Based RNN

Dataset: Keras Reuters Dataset

- Created a self-attention based bidirectional LSTM network and obtained summation weight vectors.
- Dot product of summation weight vectors with weighted LSTM hidden states generates semantic vectors.



Example

Example:
graham mccormick oil and gas partnership said it completed the sale of interests in two major oil and gas fields to lt energy assets international corp for 21 mln dlrs the company said it sold about one half of its 50 pct interest in the oak hill and north fields its two largest producing properties it said it used about 20 mln dlrs of the proceeds to prepay principal on its senior secured notes semi annual principal payments on the remaining 40 mln dlrs of notes have been satisfied until december 1988 as a result it said the company said the note agreements were amended to reflect an easing of some financial covenants and an increase of interest to 13 5 pct from 13 0 pct until december 1990 it said the noteholders exercise price for 1 125 000 warrants was also reduced to 50 cts from 1 50 dlrs the company said energy assets agreed to share the costs of increasing production at the oak hill field reuter 3

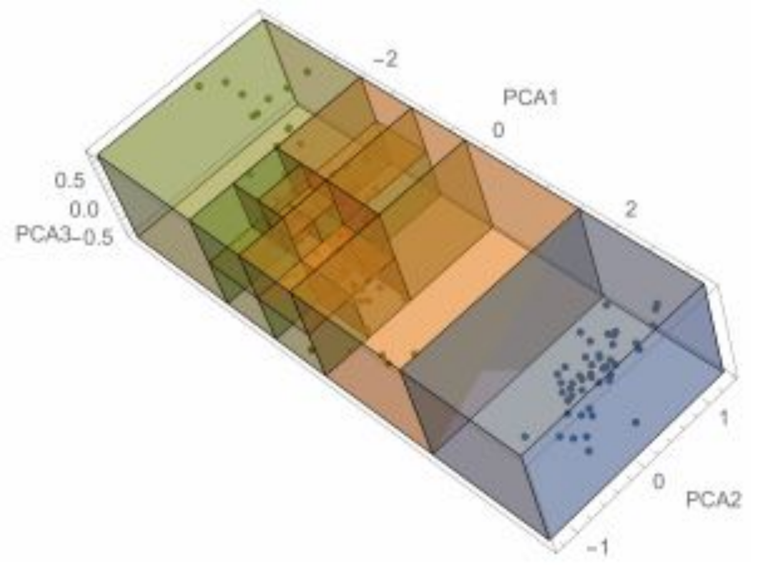
Semantic Vector Obtained:

[0.00000715, 0.0000268, 0.000229083, ..., 0.002717707, 0.000000487, 0.0000377]

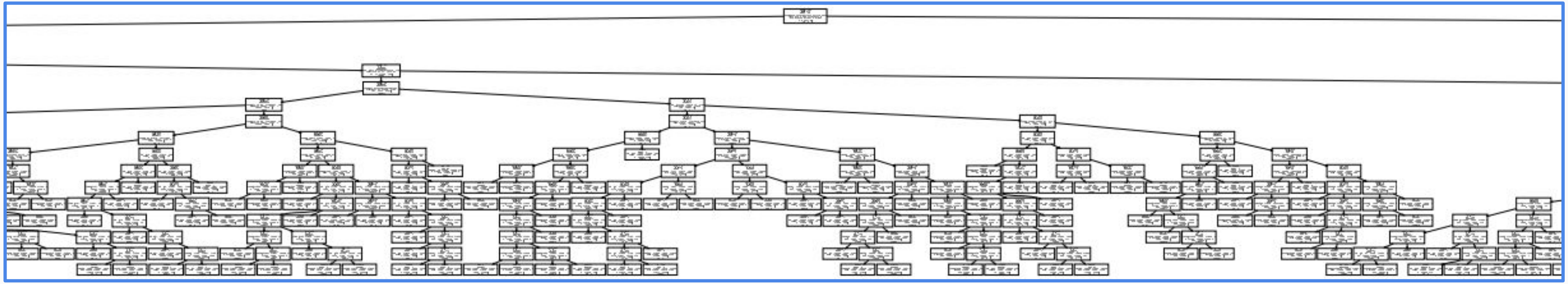
Dimension of Semantic Vector: 300

Topology Adaptive Hypercubes

- Used Decision Tree model to detect class boundaries from the semantic vectors obtained.
  - Depth of Decision Tree: 15
- Computed hypercubes (min-max pair for each feature) using the above model



Result

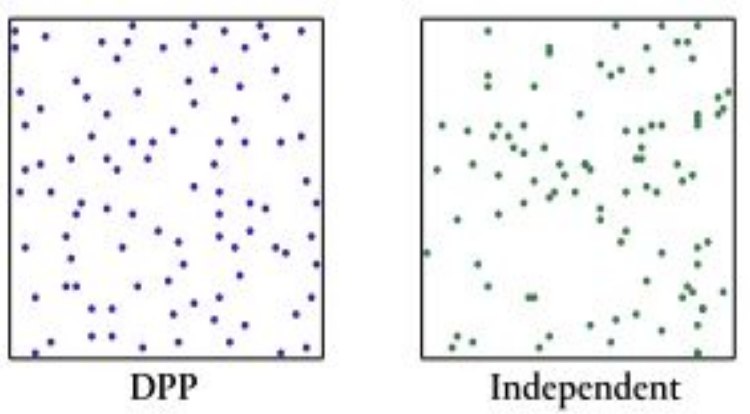


Sample Hypercube:

[[5.14387821e-09, 4.95612955e+00], [7.42266160e-09, 3.09693003e+00], [1.41198413e-08, 8.10773563e+00], ..., [6.98655605e+00, 9.89116669e+00], [6.87536339e-09, 9.78756011e-01], [1.14355033e-03, 6.75162077e-01]]

Determinantal Point Process

- Selected diverse hypercubes using DPP
    - Metric for diversity: Inverse of Euclidean Distance between hypercube centres.
- Used greedy approach to select the subset of hypercubes of required length
- Used the points of the hypercube as the representative points



Result

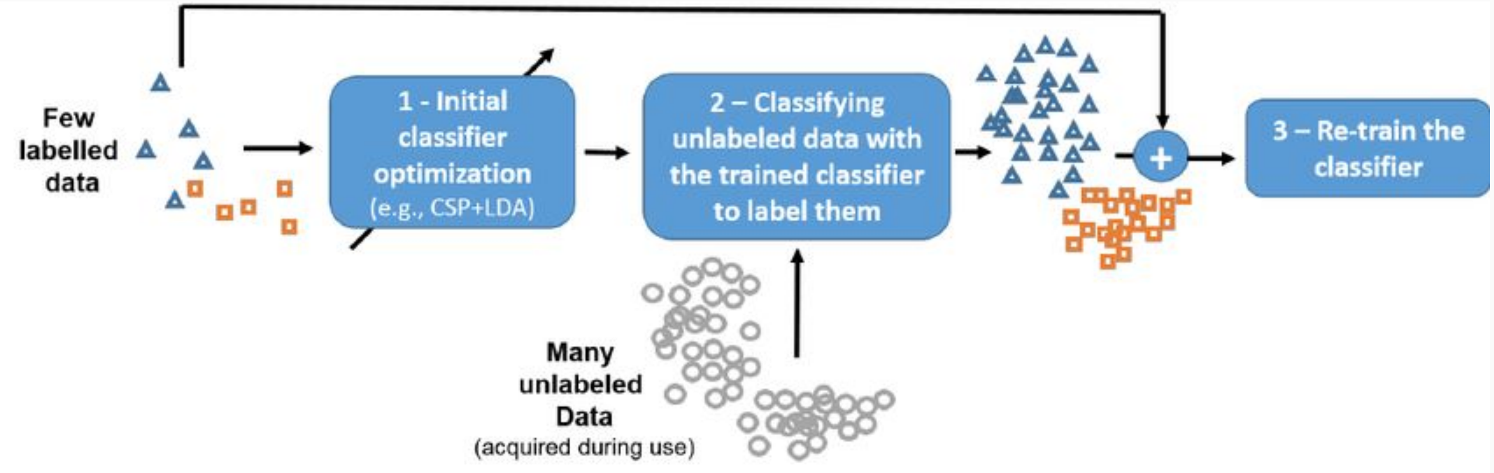
- Hypercubes selected: Found 150 most diverse hypercubes out of 1173

```
[668, 575, 574, 669, 261, 576, 260, 545, 1054, 546, 1053, 500, 1055, 497, 993, 498, 992, 1168, 547, 1169, 548, 553, 1170, 555, 1173, 258, 259,
741, 1167, 740, 1164, 549, 1156, 570, 665, 501, 667, 457, 1155, 432, 1158, 439, 1159, 435, 1171, 666, 443, 554, 1162, 1157, 551, 454, 1163,
442, 1142, 440, 482, 1130, 1166, 1131, 73, 552, 417, 71, 483, 481, 1137, 461, 884, 1138, 1136, 460, 885, 529, 1144, 959, 530, 958, 208, 398,
493, 437, 339, 490, 13, 164, 10, 163, 594, 269, 1172, 1141, 595, 331, 1049, 265, 396, 335, 447, 262, 1145, 153, 337, 161, 209, 886, 527, 1146,
1143, 556, 593, 444, 448, 557, 1019, 1140, 1073, 994, 1027, 1075, 9, 154, 581, 1161, 1139, 999, 494, 782, 167, 1134, 17, 492, 572, 777, 598,
778, 1056, 792, 998, 1060, 1057, 1058, 403, 401, 399, 1035, 573, 625, 1034, 624, 40, 1038, 31, 1042, 628, 1044, 629, 1063, 1064, 1062, 32, 599,
1165, 867, 859, 866, 649, 752, 621, 619, 611, 783, 749, 871, 648, 750, 979, 268, 970, 39, 875, 650, 1103, 1051, 969, 793, 600, 601, 977, 978,
989, 1059, 558, 34, 1000, 1147, 571, 1148, 631, 630, 1050, 647, 4, 786, 354, 3, 84, 28, 1, 2, 0, 66, 16, 24, 155, 80, 11, 12, 23, 15, 20, 19,
26, 5, 58, 70, 8, 6, 25, 18, 38, 7, 21, 29, 43, 22, 45, 44, 37, 35, 41, 27, 46, 47, 51, 50, 36, 64]
```

- Number of labelled training points used in SSL: 924
- Number of unlabelled training points: 8058

Semi-Supervised Learning

- Used the generated representative samples to create class boundaries using Decision Tree model
- Predicted the samples of the rest of the data considering it as unlabeled
- Trai



Final Results

Using the Fully Supervised Learning:

Training Accuracy: 77.02%(At depth 15)
Test Accuracy: 47.51%

Using Semi-Supervised Learning:

Training Accuracy: 94.74%
Test Accuracy: 42.61%

Scope of Improvement

- Analyse results for different lengths of semantic vectors
- Analyse the results for different depths of the decision tree
- Use a more complex classification model like Neural Networks to generate the topology adaptive hypercubes
- Implement DPP using distinct paradigms to select diverse hypercubes
- Use DPP to select representative samples from the selected diverse hypercubes

References:

[1] Xiaojin Zhu. Semi-Supervised Learning Tutorial. ICML 2017

[2] Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, Yoshua Bengio. A Structured Self-attentive Sentence Embedding. ICLR 2017

[3] W. Brent Daniel, Enoch Yeung. A Constructive Approach for One-Shot Training of Neural Networks Using Hypercube-Based Topological Coverings. arXiv:1901.02878 [cs.LG], 2019

[4] Alex Kulesza, Ben Taskar. Determinantal point processes for machine learning. arXiv:1207.6083 [stat.ML], 2013