SDML HW1 Presentation
2019 AI Cup Abstract Classification

Team: AndyIsMyBoss

Member: D06943020 鄧傑方

D06943017 陳奕達

Advisor: Prof. Shou-De Lin

Date: 2019/10/17

Outline

- Problem Formulation
- Graph-based Embedding
  - Citation
  - Author
  - Category
- Text-based Embedding
  - GloVe + GRU
  - Doc2Vec
  - BERT
- Graph + Text Embedding
- Other Techniques
  - Regularization
  - Specific output threshold
  - Ensemble

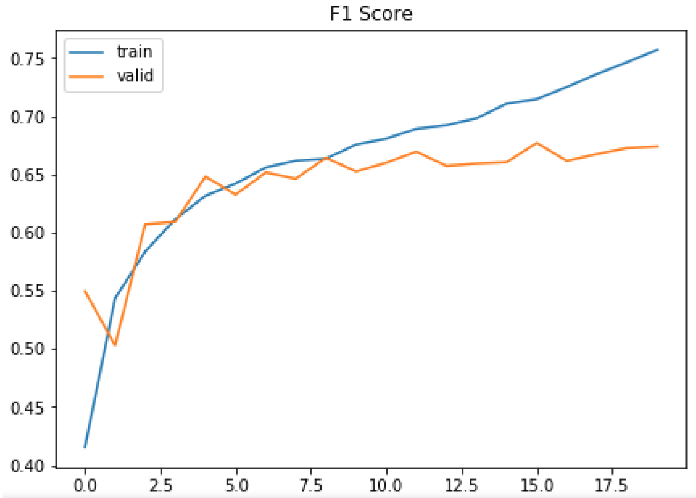
Problem Formulation

- **Title:**
  - A Brain-Inspired Trust Management Model to Assure Security in a Cloud based IoT Framework for Neuroscience Applications
- **Abstract:**
  - Rapid popularity of Internet of Things (IoT) and cloud computing permits neuroscientists to collect multilevel and multichannel brain data to better understand brain functions, diagnose diseases, and devise treatments.$$$To ensure secure and reliable data communication between end-to-end (E2E) devices supported by current IoT and cloud infrastructure, trust management is needed at the IoT and user ends.$$$This paper introduces a …
- **Authors:**
  - Mahmud/Kaiser/Rahman/Rahman/Shabut/Al-Mamun/Hussain
- **Categories:**
  - cs.CR/cs.AI/q-bio.NC
- **Date:**
  - 2018-01-11

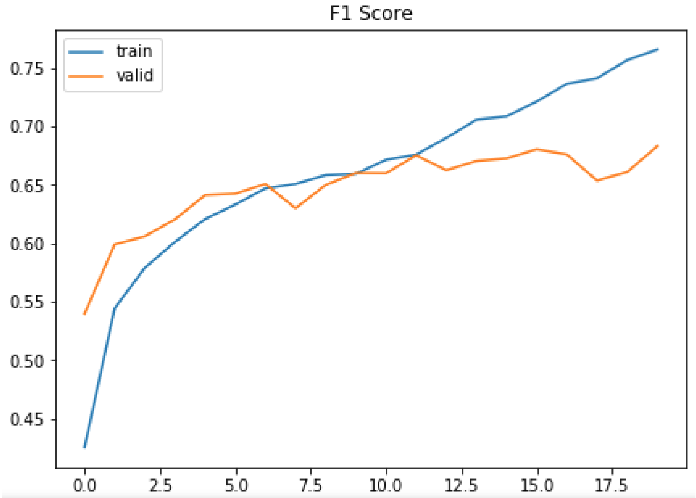
Theoretical
Engineering
Empirical
Others

Citation Graph Embedding

- Normalize all paper title from paper_title_abstract.csv and train/test dataset for matching
- ~1372K connections
- ~744K nodes
- Utilize Node2Vec to generate 64 dimensional embedding
  - Concatenate with GRU output for FC layers



0.6771

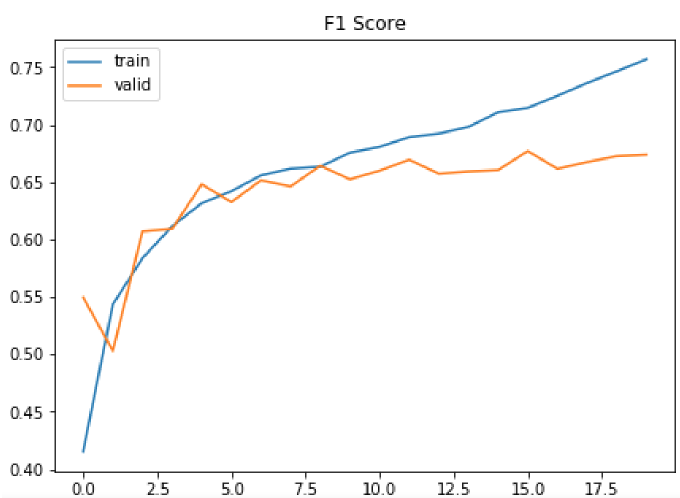


0.6830

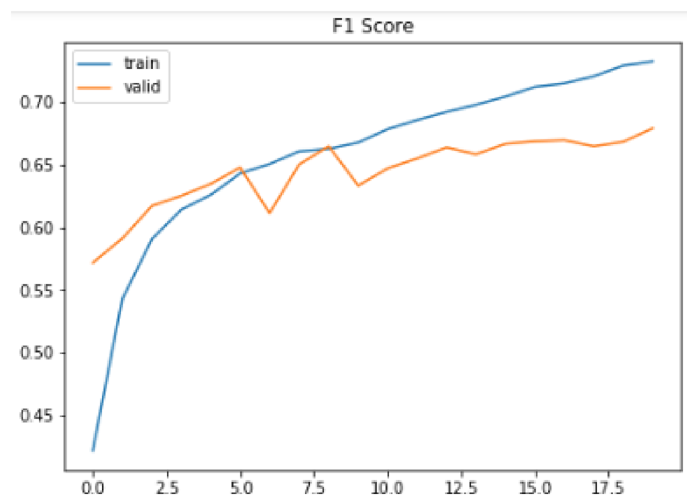
Author Graph Embedding

- Because the column only lists last name
  - Same last name but different person
  - Pick the row has exact same author list
- 1129 groups have the same author list
- 2766 nodes
- 3403 connections

0	Zhao/Yang/Yu/Yao/Lin/Li
1	Mishra
2	Kundu/Kalsi/Backhaus
3	Bläsius/Rutter
4	Scott/De Jong
	...
19995	Cuff
19996	Schreiber/Loft
19997	Agrawal/Vishwanath
19998	Abuelenin
19999	Nguyen



0.6771



0.6791

Category Graph Embedding (1/2)

- There are overall 156 different categories
- For each category, connect all papers have this category
  - O(N^2)
  - ~47M connections
  - Too many connections ➔ 302 hours…

0	cs.NI/cs.CR
1	cs.HC
2	cs.SY/math.OC
3	cs.DS/cs.DM/math.CO
4	cs.NE
	...
19995	cs.IT/math.IT
19996	physics.comp-ph/cs.PF
19997	cs.IT/math.IT
19998	cs.SY
19999	stat.CO/cs.LG/stat.ML

```
embed_dim = 64
node2vec = Node2Vec(graph, dimensions=embed_dim, walk_length=6, num_walks=10, q=4, p=2, workers=32)
```

```
Computing transition probabilities:   0%|          | 2/26998 [01:05<302:05:57, 40.29s/it]
```

- Some categories have more than 2000 papers
  - May be meaningless
- Only choose categories have less than 1500 papers
  - ~9M connections, 15 hours
  - But some papers are not connected

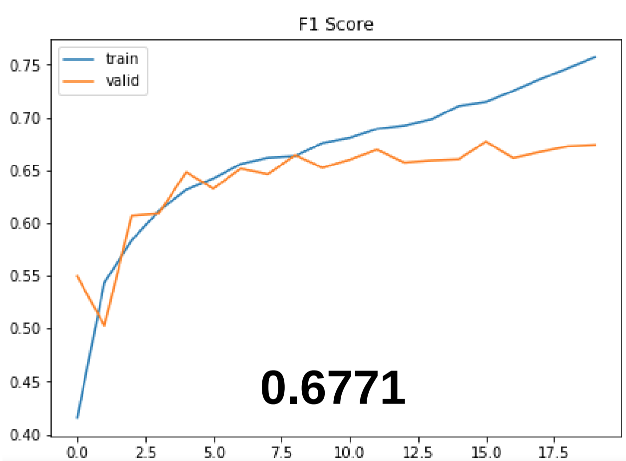
Category Graph Embedding (2/2)

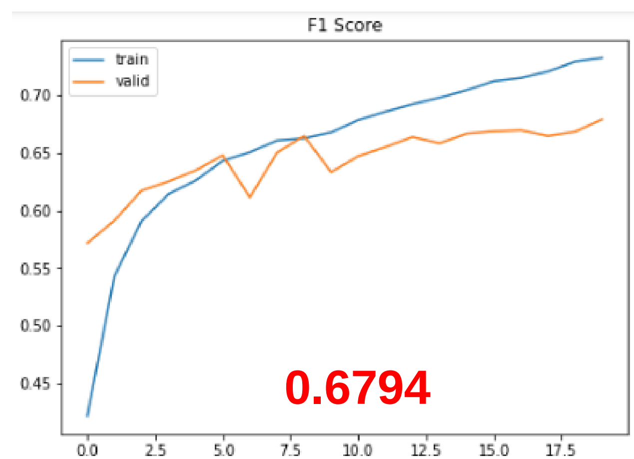
- For each paper, choose the rarest category
- Besides, random sample 200 papers from the category as connections
  - ~5M connections
  - Ensure each paper has connections

```
quant-ph/cs.ET/hep-lat [361, 210, 5]
cs.CR [1403]
cs.CV [4058]
cs.NI [1555]
cs.CR/cs.NI [1403, 1555]
cs.CE [431]
cs.IT/math.CO/math.IT [3043, 337, 3043]
cs.NI [1555]
cs.IT/math.IT/q-bio.NC [3043, 3043, 187]
cs.CR [1403]
cs.NI/cs.AI [1555, 2673]
cs.DC [1186]
cs.IT/math.IT [3043, 3043]
cs.PF/cs.DC/cs.MS/cs.SE [246, 1186, 169, 894]
```

```
embed_dim = 64
node2vec = Node2Vec(graph, dimensions=embed_dim, walk_length=6, num_walks=10, q=4, p=2, workers=32)
```

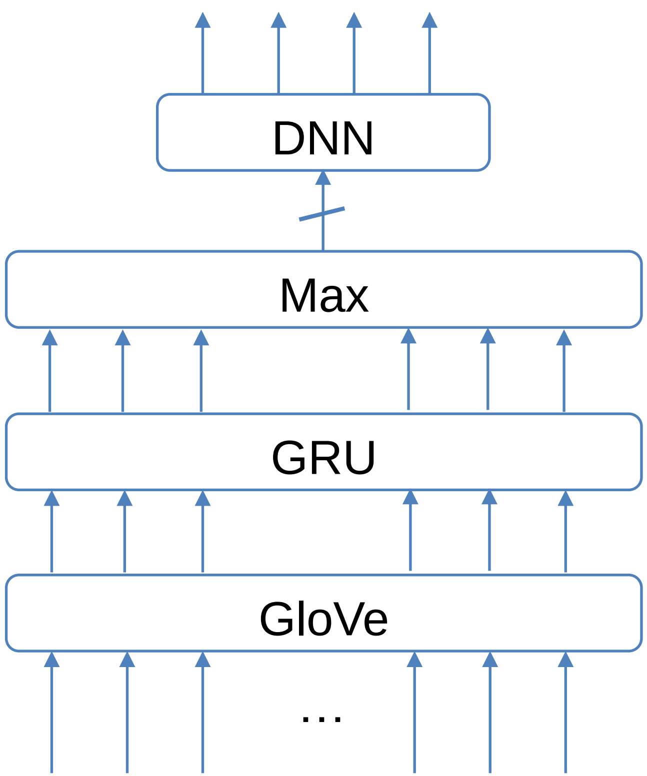
```
Computing transition probabilities:   0%|          | 2/26998 [00:00<1:46:04,  4.24it/s]
```

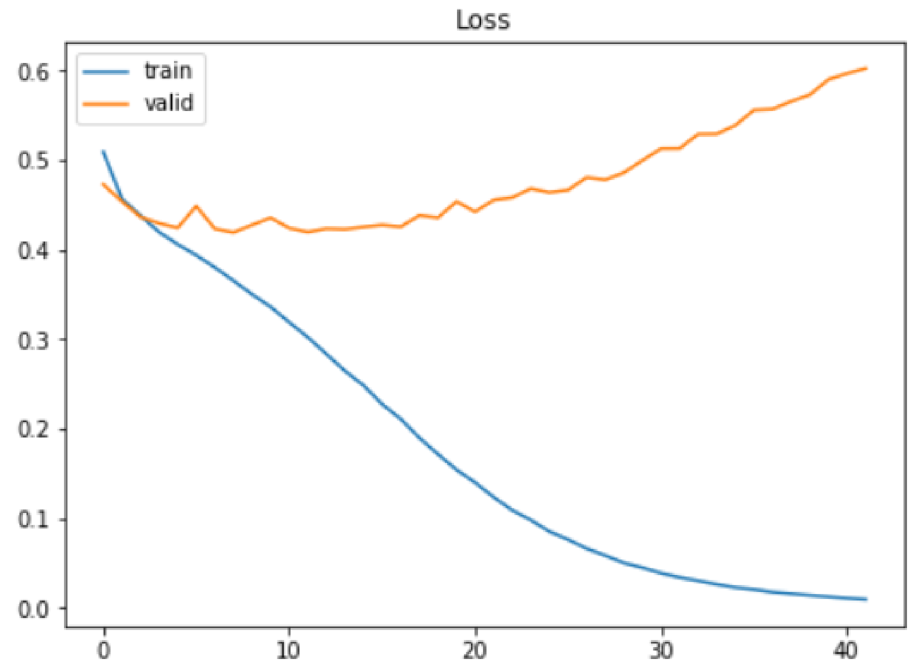




Baseline: GloVe + GRU

- GloVe: Pretrained word vector on several datasets (dim=300)
- Gated Recurrent Unit (GRU)
- Overall architecture:





Baseline performance: 0.6771

Doc2Vec

- Training:
  - All paper abstract (685163)
  - Train/test paper abstract (27000)
  - All paper title + abstract
  - Train/test paper title + abstract

	PaperId	PaperTitle	Abstract
0	695875	dynamic bayesian networks in dynamic reliabili...	In this paper, we review briefly the different...
1	2207358	an algorithm for sat without an extraction phase	An algorithm that could be implemented at a mo...
2	5536715	customizable isolation in transactional workflow	In Workflow Management Systems (WFMSs) safety ...
3	9313061	a comparative analysis of the ant based system...	The huge growth of mobile and handheld devices...
4	9560197	social networks and natural resource managemen...	Social networks and natural resource managemen...
...	...	...	...
685158	2962678111	a foundry of human activities and infrastructures	Direct representation knowledgebases can enhan...
685159	2962678112	on the strong converses for the quantum channe...	A unified approach to prove the converses for ...
685160	2962678113	the multimedia product between design and info...	The paper investigates the possible coherent a...
685161	2962678114	coordinated heterogeneous distributed percepti...	We investigate a reinforcement approach for di...
685162	2962678115	processing of test matrices with guessing corr...	It is suggested to insert into test matrix 1s ...

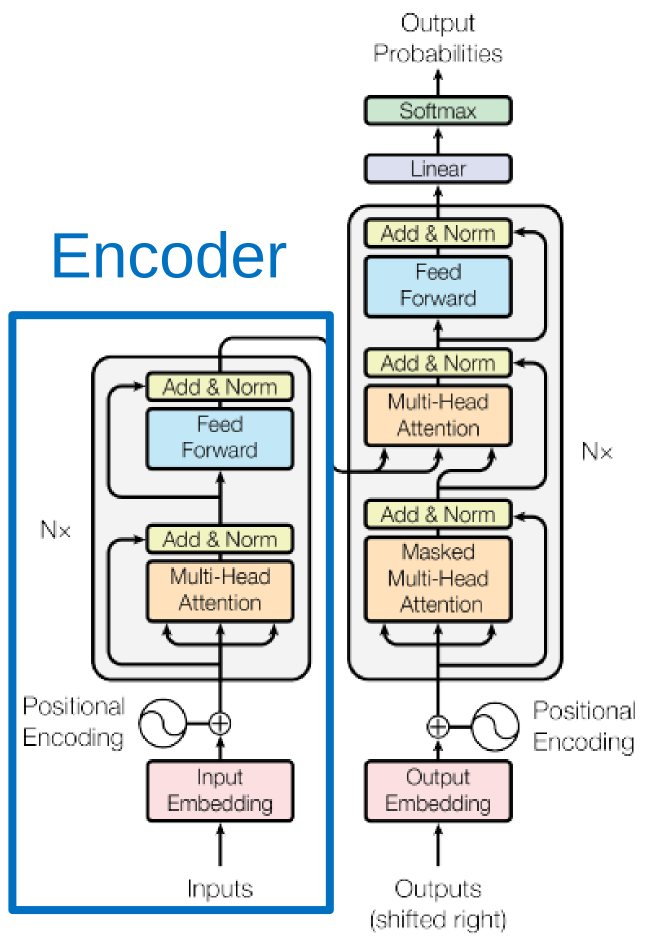
- Evaluate on Gradient Boosting Classifier

Dimension	All abstract	Train/test abstract	All abstract + title	Train/test abstract + title
32	0.6100	0.6185	0.6151	**0.6283**
64	0.6202	0.6166	0.6034	0.6259

Training on train/test dataset with title

Bidirectional Encoder Representation of Transformer (BERT) [1]

- Detailed description can refer to L4_bert.ppt
- Use BERT to generate abstract embedding
- Method
  - Load pretrained model (tokenizer, model)
  - Insert [CLS], [SEP] to abstact
  - Encode abstract by tokenizer
  - Input encoded abstract to model
  - Get encoder output or pooled output from BERT model
- Lots of pre-trained model
  - bert-base-uncased: 768-hidden, 110M param
  - bert-large-uncased: 1024-hidden, 340M param
  - bert-base-cased: 768-hidden, 110M param
  - bert-large-cased: 1024-hidden, 340M param

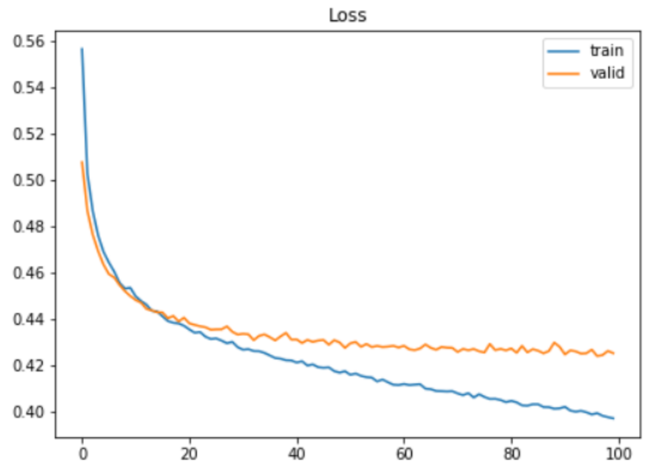


- ***Python Package:*** https://github.com/huggingface/transformers

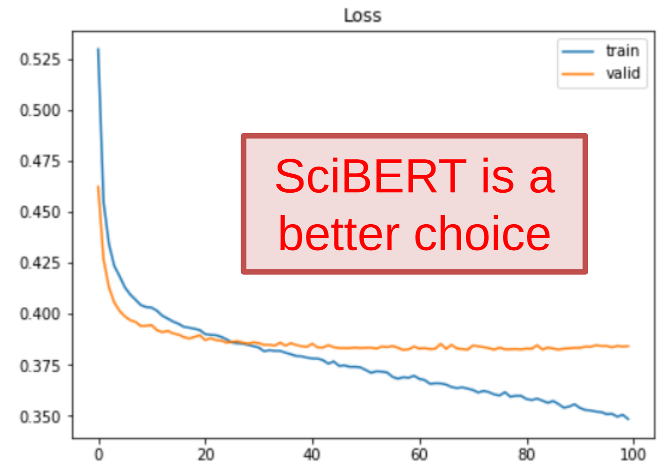
SciBERT: Enhanced BERT on Scientific Text

[2]

- Enhancement
  - Trained on a large corpus of scientific text (1.14M papers)
  - Having an in-domain vocabulary, 42% overlap of same size vocab comparing to BERT
- Suggest fine-tuning on specific task using 2-4 epochs and learning rate equals 2e-5
- Final used model: scibert-scivocab-uncased (768-hidden, 110M param)



Best F1 score [0.6955665024630542, 43]



Best F1 score [0.7214611872146119, 5]

Joint Graph and Text Embedding [3-4]

- Text-associated DeepWalk (TADW)
  - Incorporate text features of vertices and graph representation learning

Classifier	Transductive SVM				SVM				
% Labeled Nodes	1%	3%	7%	10%	10%	20%	30%	40%	50%
DeepWalk	-	-	49.0	52.1	52.4	54.7	56.0	56.5	57.3
PLSA	45.2	49.2	53.1	54.6	54.1	58.3	60.9	62.1	62.6
Text Features	36.1	49.8	57.7	62.1	58.3	66.4	69.2	71.2	72.2
Naive Combination	39.0	45.7	58.9	61.0	61.0	66.7	69.1	70.8	72.0
NetPLSA	45.4	49.8	52.9	54.9	58.7	61.6	63.3	64.0	64.7
TADW	**63.6**	**68.4**	**69.1**	**71.1**	**70.6**	**71.9**	**73.3**	**73.7**	**74.2**

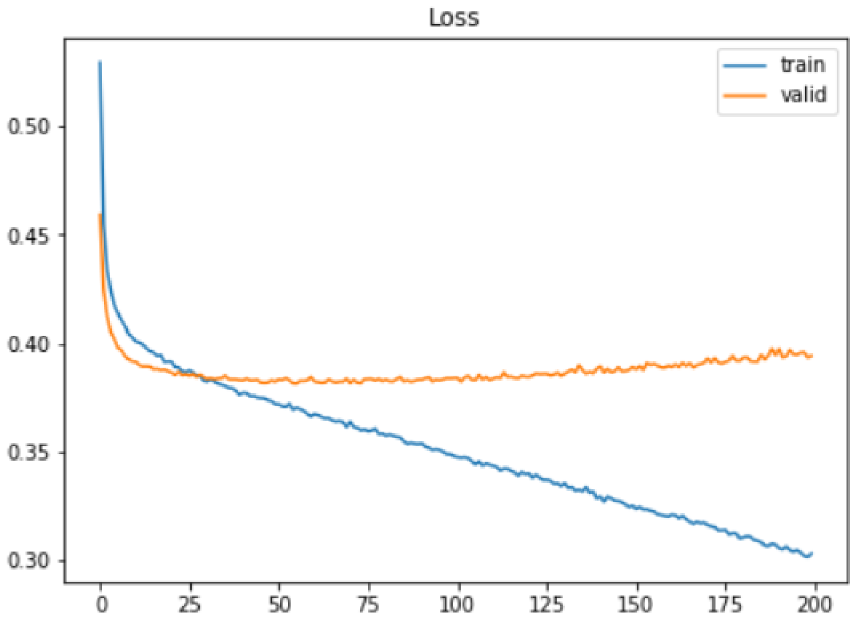
	Feature	Category Graph	TADW
All_Abstract_64	0.6202	0.5299	0.6298
Train/Test_Abstract_Title_64	0.6259	0.5299	0.6271

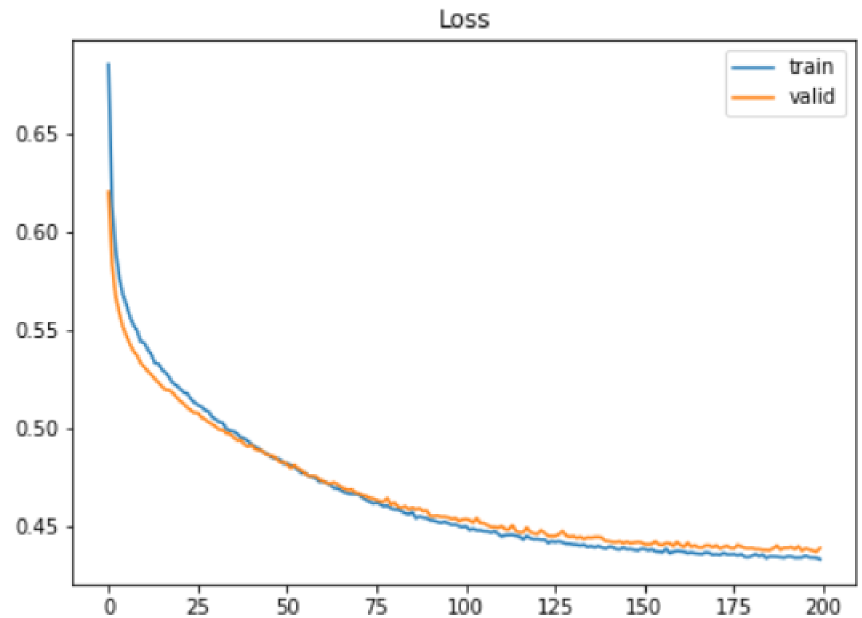
Other Techniques: Regularization

- Apply L2 regularization to overall loss

$$L_2 \text{ regularization term} = ||\boldsymbol{w}||_2^2 = w_1^2 + w_2^2 + \ldots + w_n^2$$

- Avoid overfitting



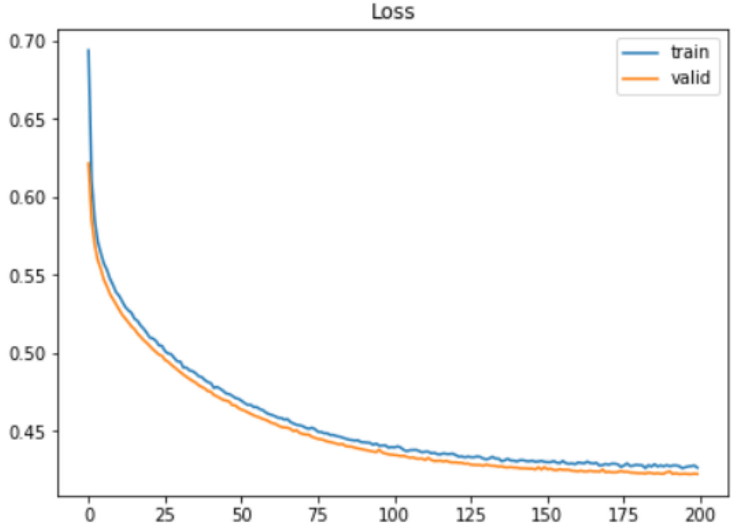


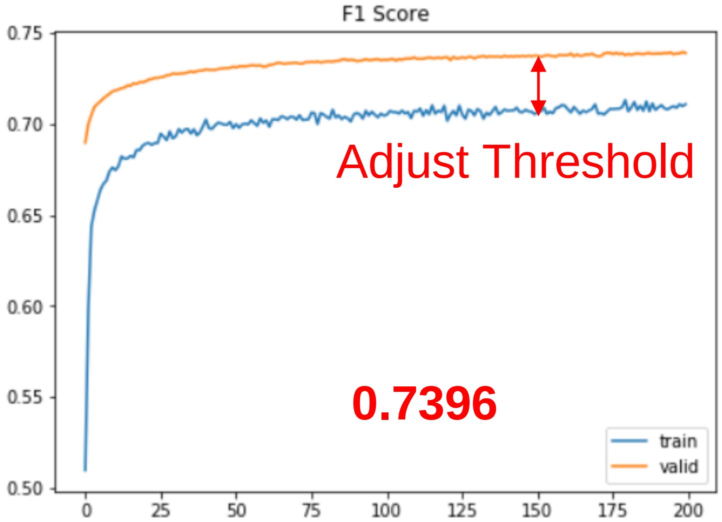
Overall Training Process with Threshold

- Select all data for training (No validation data due to no overfitting)
- Pass title+abstract to SciBERT model ➔ SciBERT embedding
- Pass title+abstract to Doc2Vec ➔ DocEmbed
- Pass DocEmbed + category graph to TADW ➔ CateEmbed
- Concat SciBERT embedding, DocEmbed, CateEmbed
- Train simple hidden layer NN using concat embedding with L2 regularization loss
- Train for large epoch to converge the final model
- Find the best threshold on each category based on training data with the best F1 score
- After deciding the first 3 categories, the other category can be decided if the first 3 categories equal 0
- Save model, prediction and threshold

Overall Experiment Result

- Experiment setup
  - Training data: all train data
  - Model size: 864 ➔ 512 ➔ 4
  - Learning rate: 2e-5
  - Lambda for L2 regularization: 0.012
  - Epoch: 200
- Experiment result





Other Techniques: Ensemble

- For each prediction, save
  - Hard prediction results
  - Soft prediction results
  - Thresholds
- Use validation F1 score to weight prediction results and thresholds

Pool	Hard Ensemble	Soft Ensemble
Validation F1 Score ~ 0.7 **without Bert** (best TBrain: 0.6869)	0.6942	0.6955
Validation F1 Score > 0.735 (best TBrain: 0.7242)	0.7211	0.7225
Validation F1 Score > 0.74 (best TBrain: 0.7242)	0.7216	0.7241

Conclusion

- Citation, author, and category graph can improve a little
- BERT is the key to improve F1 score
  - SciBERT is better than BERT in our task
- Regularization is important for this task to avoid overfitting
- Adjust threshold for each output is important

Future Work

- Take training data from another task
- Take BERT output as TADW initial feature
- Try UPP-SNE
- Finetune SciBERT model using our corpus
- Adaboost multi trained model

	Method	Amherst	Hamilton	Mich	Rochester	Citeseer	Cora	Facebook
	DeepWalk	0.6257	0.6273	0.3944	0.5593	0.3365	0.5062	<u>0.6953</u>
	LINE	**0.6908**	**0.6718**	**0.4127**	**0.6070**	0.2806	0.3905	0.6952
	node2vec	0.6662	0.6328	0.4114	0.5777	<u>0.4574</u>	<u>0.6216</u>	0.6952
Accuracy	M-NMF	0.6545	0.6374	0.3279	0.5071	0.2379	0.3640	0.6952
	TADW					0.2778	0.4731	0.6953
	HSCA					0.2794	0.4594	0.6957
	UPP-SNE					**0.5748**	**0.6832**	**0.8328**
	DeepWalk	0.4873	0.4390	**0.1897**	0.3468	0.0896	0.3308	0.0142
	LINE	**0.5030**	**0.4529**	0.1858	**0.3547**	0.0511	0.1639	0.0113
	node2vec	0.4742	0.4144	0.1824	0.3193	<u>0.2027</u>	<u>0.4333</u>	0.0162
NMI	M-NMF	0.4696	0.4330	0.1304	0.2971	0.0464	0.1201	<u>0.0176</u>
	TADW					0.0845	0.3001	0.0651
	HSCA					0.0902	0.3148	0.0151
	UPP-SNE					**0.3005**	**0.4911**	**0.2095**

[5]

Reference

[1] Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805,* 2018.

[2] Beltagy, Iz, Arman Cohan, and Kyle Lo. "Scibert: Pretrained contextualized embeddings for scientific text," *arXiv preprint arXiv:1903.10676,* 2019.

[3] C. Yang, Z. Liu, D. Zhao, M. Sun, and E. Y. Chang, "Network representation learning with rich text information," in *Proceedings of the 24th International Joint Conference on Artificial Intelligence*, 2015, pp. 2111–2117.

[4] https://github.com/thunlp/OpenNE

[5] D. Zhang, J. Yin, X. Zhu, and C. Zhang, "Network representation learning: A survey," *IEEE Transactions on Big Data*, 2018.