GNNs for Text Classification

Pouya Sadeghi - summer 2023

Corpus-level vs. Document-level (1/2)

Corpus-level graphs intend to construct the graph to represent the whole corpus

- Word and Doc >> TextGCN, BertGCN, TG-Transformer, TensotGCN, HeteGCN
- Doc >> knn-GCN, TextGTL
- Word >> VGCN-BERT
- \+ Topic >> HGAT, DHTG

Corpus-level vs. Document-level (2/2)

Document-level graphs focus on representing the non-Euclidean relations existing in a single text body

- Local word consec. >> TextING, DADGNN
- Global word co-occ. >> DAGNN, ReGNN
- Other >> HyperGAT, GTNT

Other categories

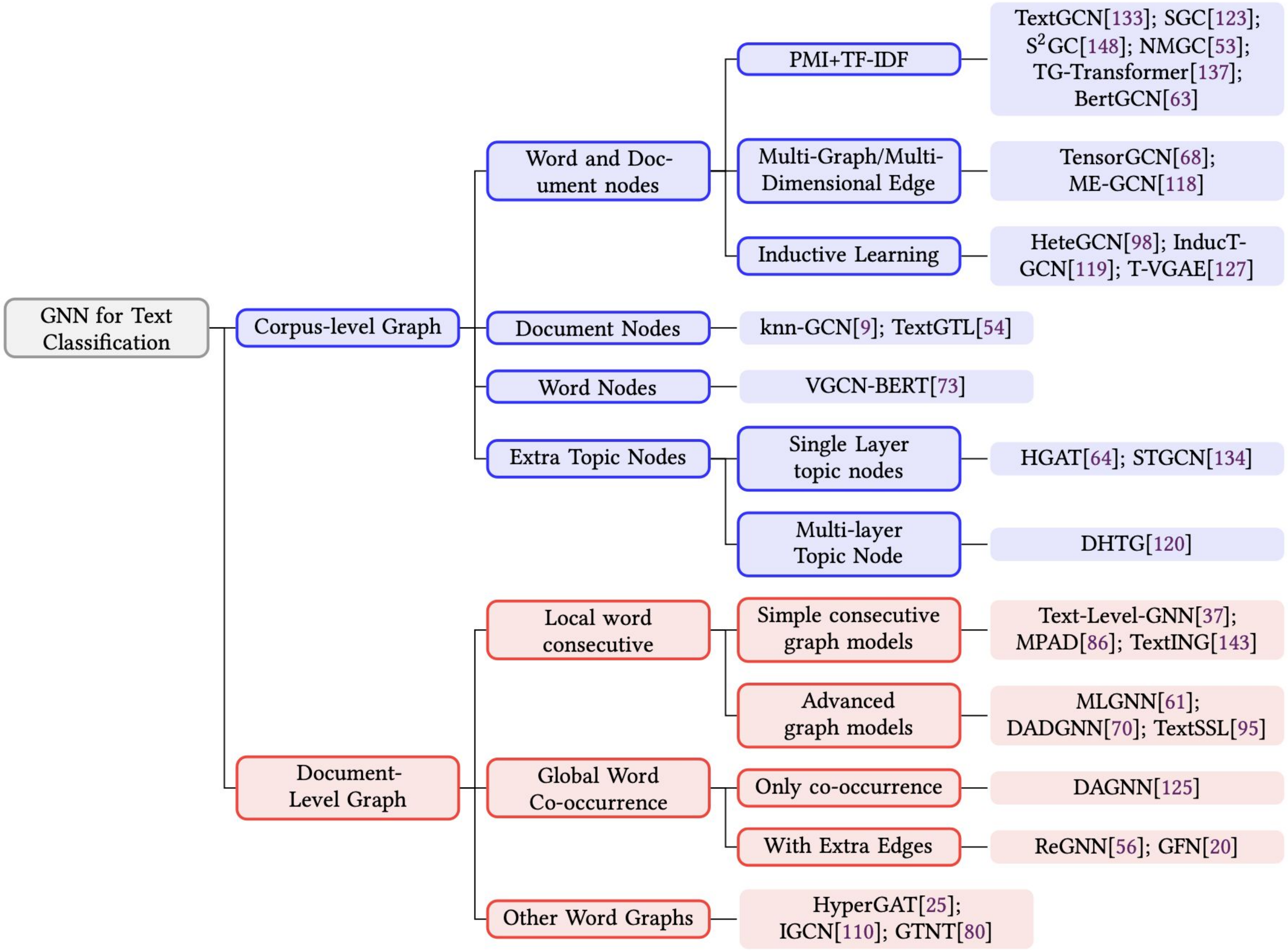
- Homogeneous/Heterogeneous (same/different node and edge type)
- Static/Dynamic

Representations

- Word-level
  - GloVe, Word2vec, FastText
    - only syntactic similarity and not the complex semantic relationships
  - ELMo, BERT, GPT
- Document-level
  - Last hidden-state of LSTM
  - [CLS] from BERT
  - TF-IDF

Training

- Supervised
- Unsupervised
- Semi-supervised
    - Inductive learning (labelled first, then unlabelled)
    - Transductive learning (labelled/unlabelled simultaneously)


GNN for Text Classification
Corpus-level Graph
Word and Document nodes
PMI+TF-IDF
TextGCN[133]; SGC[123]; S2GC[148]; NMGC[53]; TG-Transformer[137]; BertGCN[63]
Multi-Graph/Multi-Dimensional Edge
TensorGCN[68]; ME-GCN[118]
Inductive Learning
HeteGCN[98]; InducT-GCN[119]; T-VGAE[127]
Document Nodes
knn-GCN[9]; TextGTL[54]
Word Nodes
VGCN-BERT[73]
Extra Topic Nodes
Single Layer topic nodes
HGAT[64]; STGCN[134]
Multi-layer Topic Node
DHTG[120]
Document-Level Graph
Local word consecutive
Simple consecutive graph models
Text-Level-GNN[37]; MPAD[86]; TextING[143]
Advanced graph models
MLGNN[61]; DADGNN[70]; TextSSL[95]
Global Word Co-occurrence
Only co-occurrence
DAGNN[125]
With Extra Edges
ReGNN[56]; GFN[20]
Other Word Graphs
HyperGAT[25]; IGCN[110]; GTNT[80]

Method 0 >> SHINE

Hierarchical Heterogeneous Graph Representation Learning for Short Text Classification

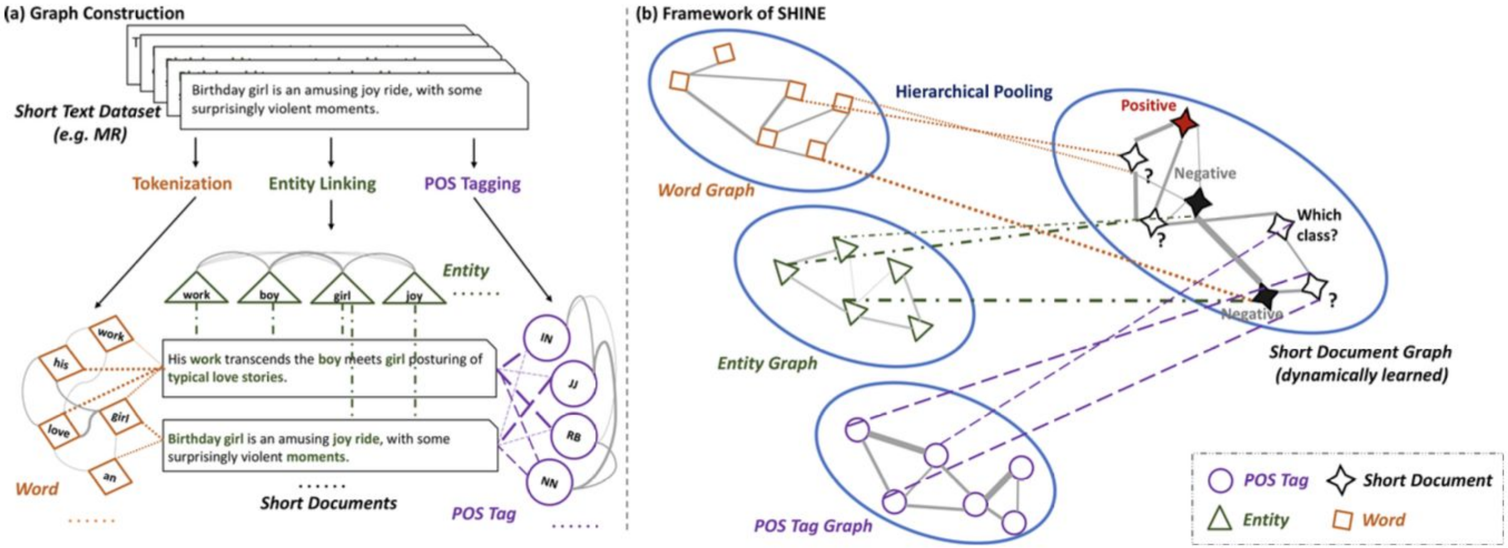
Yaqing Wang[1], **Song Wang**[1,2], **Quanming Yao**[*3], **Dejing Dou**[1]
[1]Baidu Research, Baidu Inc., China
[2] Department of ECE, University of Virginia, USA
[3] Department of EE, Tsinghua University, China
{wangyaqing01, v_wangsong07, doudejing}@baidu.com
qyaoaa@tsinghua.edu.cn

Abstract

Short text classification is a fundamental task in natural language processing. It is hard due to the lack of context information and labeled data in practice. In this paper, we propose

knowing "*Birthday girl*" is a 2001 movie. A harder case is to understand a web search snippet such as "*how much Tesla*", which usually does not contain word order nor function words (Phan et al., 2008). In addition, real STC tasks usually only

Architecture



Performance

		Ohsumed		Twitter		MR		Snippets		TagMyNews	
Group	Model	ACC	F1	ACC	F1	ACC	F1	ACC	F1	ACC	F1
	TLGNN	35.76	13.12	58.33	53.86	58.48	58.45	70.25	63.18	44.43	32.33
	TextING	38.27	21.34	59.79	59.44	58.89	58.76	71.13	70.71	52.53	40.20
	HyperGAT	36.60	19.98	58.42	53.71	58.65	58.62	70.89	63.42	45.60	31.51
	TextGCN	41.56	<u>27.43</u>	60.15	59.82	59.12	58.98	77.82	71.95	54.28	46.01
	TensorGCN	41.84	24.24	61.24	61.19	59.22	58.78	74.38	73.96	55.58	43.21
(D)	STCKA	30.19	10.12	57.45	56.97	53.22	50.11	68.96	61.27	30.44	20.01
	HGAT	<u>42.68</u>	24.82	63.21	62.48	<u>62.75</u>	<u>62.36</u>	<u>82.36</u>	74.44	<u>61.72</u>	<u>53.81</u>
	STGCN	33.91	27.22	<u>64.33</u>	<u>64.29</u>	58.18	58.11	70.01	69.93	34.74	34.01
	SHINE (ours)	**45.57**	**30.98**	**72.54**	**72.19**	**64.58**	**63.89**	**82.39**	**81.62**	**62.50**	**56.21**
	relative ↑ (%)	6.77	12.94	12.76	12.29	2.92	2.45	0.85	3.17	1.26	4.46

Method 1 >> DADGNN

Deep Attention Diffusion Graph Neural Networks for Text Classification

Yonghao Liu[†], **Renchu Guan**[†], **Fausto Giunchiglia**[‡], **Yanchun Liang**[◇], **Xiaoyue Feng**[†*]

[†]Key Laboratory of Symbolic Computation and Knowledge Engineering of the Ministry of Education,
College of Computer Science and Technology, Jilin University, China

[◇]Zhuhai Sub Laboratory, Key Laboratory of Symbolic Computation and Knowledge Engineering of the
Ministry of Education, Zhuhai College of Science and Technology, China

[‡]DISI, University of Trento, Italy

yonghao20@mails.jlu.edu.cn {guanrenchu, ycliang, fengxy}@jlu.edu.cn
fausto.giunchiglia@unitn.it

Abstract

Text classification is a fundamental task with broad applications in natural language processing. Recently, graph neural networks (GNNs) have attracted much attention due to their powerful representation ability. However, most existing methods for text classification based on

recurrent neural networks (RNNs) (Liu et al., 2015) are becoming more popular due to their strong performance in text mining. These models can capture semantic and syntactic information in local consecutive word sequences well.

Recently, graph neural networks (GNNs) have

What is it about?

GNN’s limitations:

- Restricted Receptive Fields >> only direct neighbor
  >> Effective text representation
- Shallow Layers >> best performance on 2 layer (over-smoothing) >> two-hop neighbor restriction
  >> attention diffusion technique >> capture the long-range word interactions
- Non-Precision Document-Level Representations >> avg pooling >> decrease expressiveness and effect of key nodes >> frequent word have more impact
  >> decouple the propagation and transformation processes
- Low-Pass Filters >> ignore the diffs >> high-freq data is important
  >> calculate the weight of each node to obtain precise document-level representations

Architecture

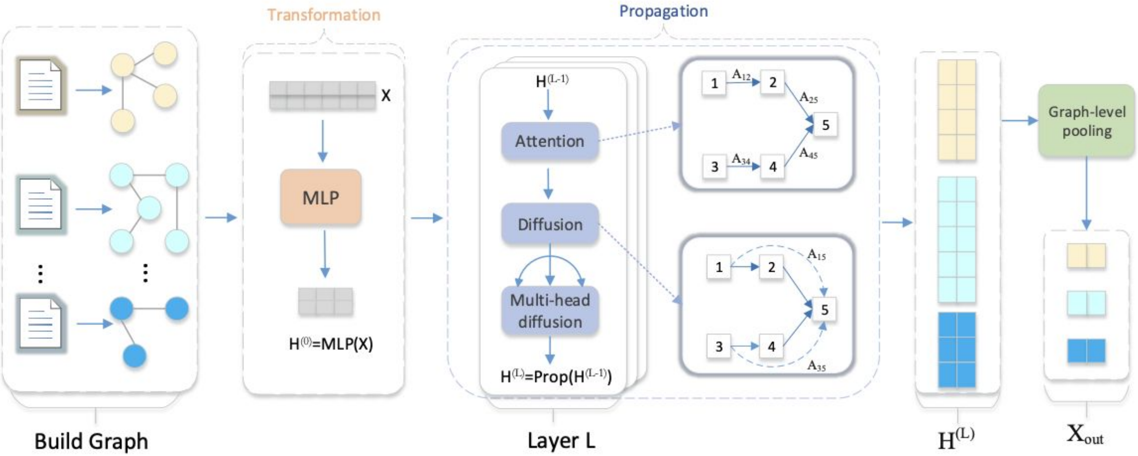


Figure 1: Overall architecture of our model (best viewed in color).

Performance

Model	IMDB	WebKB	R52	R8	AG news	DBLP	TREC	MR	SST-1	SST-2
CNN	86.15±0.60	86.87±0.23	87.59±0.48	95.71±0.52	89.13±0.31	75.28±0.61	93.62±0.55	77.75±0.72	42.30±0.41	80.07±0.51
LSTM	85.91±0.71	86.51±0.77	90.48±0.86	96.09±0.19	86.06±0.72	74.11±0.75	93.01±0.41	77.33±0.89	41.92±0.63	79.52±0.61
Bi-LSTM	86.62±0.16	86.57±0.36	90.54±0.91	96.31±0.33	86.52±0.31	72.25±1.27	93.32±0.72	77.68±0.86	42.63±0.66	80.56±0.21
fastText	80.21±0.25	82.96±0.36	92.81±0.09	96.13±0.21	91.49±0.12	71.19±0.52	91.29±0.69	75.14±0.20	36.08±0.81	81.45±0.16
PV-DBOW	75.96±0.26	72.62±0.41	78.29±0.11	85.87±0.10	81.25±0.36	63.59±0.21	80.36±0.35	61.09±0.10	38.12±0.33	72.92±0.12
LEAM	83.29±0.55	83.95±0.25	91.84±0.23	93.31±0.24	91.75±0.35	72.62±1.59	89.21±0.57	76.95±0.45	42.93±0.69	80.52±0.19
Graph-CNN	OOM	83.29±1.22	92.75±0.22	96.99±0.12	87.56±0.29	71.37±1.26	90.39±1.52	77.22±0.27	35.23±0.21	76.95±0.62
TextGCN	OOM	86.17±0.96	93.56±0.18	97.07±0.10	90.84±1.32	76.72±0.69	91.40±0.39	76.74±0.20	40.65±0.06	81.02±0.40
SGC	OOM	87.39±0.66	94.02±0.21	97.21±0.11	91.06±0.62	76.79±0.72	92.29±1.26	75.91±0.36	41.63±0.41	75.95±0.92
Text-level GCN	OOM	89.91±0.51	94.62±0.32	97.83±0.20	OOM	OOM	94.09±0.36	75.96±0.56	43.02±0.65	81.75±0.36
HyperGAT	86.32±0.71	87.46±0.55	94.98±0.27	97.97±0.23	91.24±0.56	72.56±0.96	93.55±1.79	78.32±0.27	41.96±0.35	81.26±0.72
DADGNN(ours)	**88.49±0.59**	**90.92±0.42**	**95.16±0.22**	**98.15±0.16**	**92.24±0.36**	**78.59±0.62**	**97.99±0.52**	**78.64±0.29**	**45.15±0.26**	**84.32±0.15**

Table 2: The results of test accuracy on document classification with different models. For each model, the mean ± standard deviation is reported. DADGNN significantly outperforms all the baselines based on t-tests ($p < 0.05$). Underline: runner-up. OOM:>16 GB.

Method 2 >> LG Transformer

SUTNLP at SemEval-2023 Task 4: LG-Transformer for Human Value Detection

Hamed Hematian Hemati, Sayed Hesam Alavian, Hossein Sameti, Hamid Beigy
AI Group, Computer Engineering Department, Sharif University of Technology, Tehran, Iran
{ hematian.hemati.hamed, hesam.alavian70}@gmail.com, {sameti, beigy}@sharif.edu

Abstract

When we interact with other humans, human values guide us to consider the human element. As we shall see, value analysis in NLP has been applied to personality profiling but not to

The purpose of this system paper is to present SUTNLP's (David Gauthier on leaderboard) work on the SemEval-2023 Shared Task 4 which is focused on developing a classifier to classify human values (Kiesel et al., 2023). Detecting human val-

Architecture

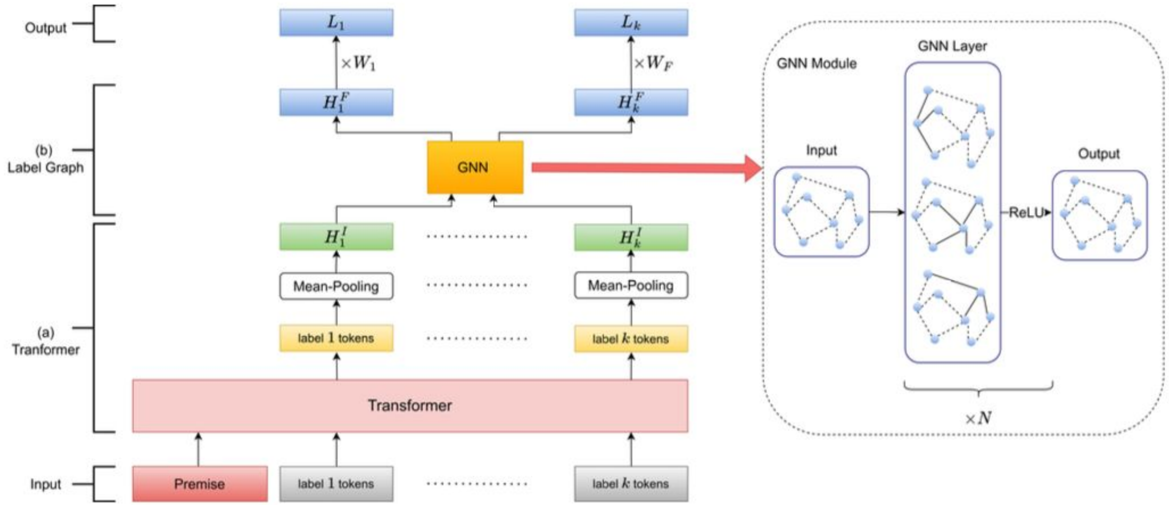


Figure 2: Architecture of the proposed model. (a) is the transformer part, (b) represents graph module, N is the hidden layers count in graph module, k is the number of labels which is 20 for the dataset. Output consists of the final logits. GNN represents Graph Neural Network.

Performance

Method	F1	Precision	Recall
1-Baseline	26.3	15.10	100.0
BERT Baseline	42.20	58.70	32.90
LG-BERT + AT	45.03	45.78	44.03
DeBERTa	48.98	51.05	47.07
DeBERTa + labels	46.30	47.45	45.21
LG-DeBERTa	49.34	48.13	50.61
LG-DeBERTa + AT	50.00	50.27	49.70

STGCN

Another Idea of bert’s embedding and for short text

BertGCN

shows that with the help of TextGCN, BERT can achieve better performance

BertGCN: Transductive Text Classification by Combining GCN and BERT

Yuxiao Lin♠, Yuxian Meng♣, Xiaofei Sun♣
Qinghong Han♣, Kun Kuang♠, Jiwei Li♠♣ and Fei Wu♠
♠College of Computer Science and Technology, Zhejiang University
♣ ShannonAI
{yuxiaolinling, kunkuang, jiwei_li, wufei}@zju.edu.cn
{yuxian_meng, xiaofei_sun, qinghong_han}@shannonai.com

Performance

Model	20NG	R8	R52	Ohsumed	MR
TextGCN	86.3	97.1	93.6	68.4	76.7
SGC	88.5	97.2	94.0	68.5	75.9
BERT	85.3	97.8	96.4	70.5	85.7
RoBERTa	83.8	97.8	96.2	70.7	89.4
BertGCN	89.3	98.1	**96.6**	**72.8**	86.0
RoBERTaGCN	**89.5**	**98.2**	96.1	**72.8**	**89.7**
BertGAT	87.4	97.8	96.5	71.2	86.5
RoBERTaGAT	86.5	98.0	96.1	71.2	89.2

VGCN-BERT

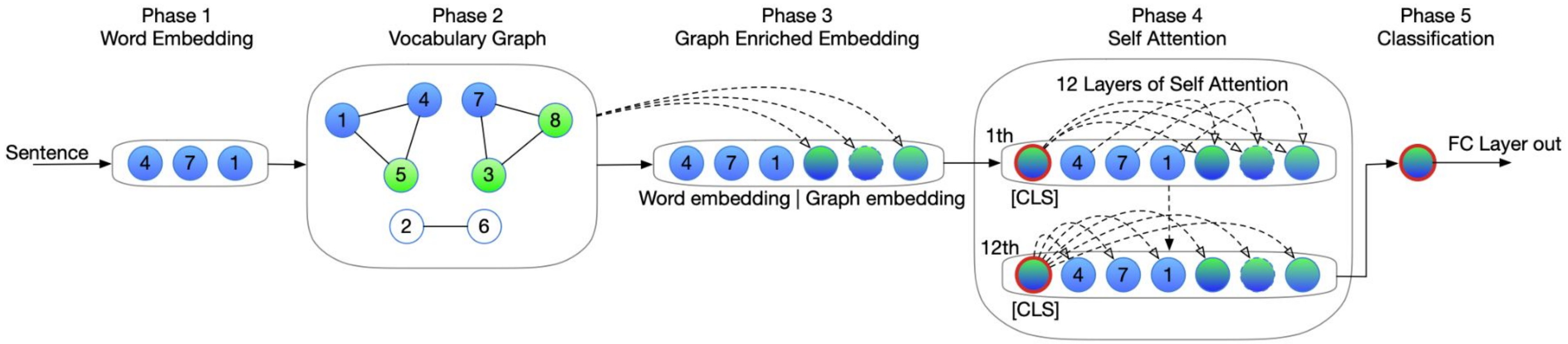
Enhances the input embedding of BERT by concatenating it with the graph embedding

VGCN-BERT: Augmenting BERT with Graph Embedding for Text Classification

Zhibin Lu, Pan Du, and Jian-Yun Nie

University of Montreal, Montreal, Canada
{zhibin.lu, pan.du}@umontreal.ca, nie@iro.umontreal.ca

Architecture



Performance

Model	**SST-2**	**MR**	**CoLA**	**ArangoHate**	**FountaHate**
MLP	80.78	75.55	61.39 (53.20)	84.71 (84.42)	79.22 (65.33)
Text-GCN	80.45	75.67	56.18 (52.30)	84.77 (84.43)	78.74 (64.54)
Bi-LSTM	81.32	76.39	62.88 (55.25)	84.92 (84.58)	79.04 (65.13)
VGCN	81.64	76.42	63.59 (54.82)	85.97 (85.69)	79.00 (64.04)
BERT	91.49	86.24	81.22 (77.02)	87.99 (87.75)	80.59 (66.61)
Vanilla-VGCN-BERT	91.38	**86.49**	80.70 (76.30)	88.01 (87.79)	81.11 (67.86)
VGCN-BERT	**91.93**	86.35	**83.68 (80.46)**	**88.43 (88.22)**	**81.26 (68.45)**

Meta-Path

TODO

Meta learning

TODO