E-017

グラフ畳み込みによる論文カテゴリー推定
Prediction of Category of Scientific Article by Graph Convolution

廣川佐千男‡
Sachio Hirokawa

1. はじめに

各データの周辺情報も、そのデータの情報として畳み込み込むことで識別性能が向上することが知られている。[7]は手書文字認識において、周辺情報を用いたニューラルネットで認識性能が向上することを示した。[6]では、このCNN (Convolution Neural Network)の手法を用いた大規模ニューラルネットで従来の画像認識の性能を一段と向上し、Deep Learning の研究のきっかけとなった。一方、[4]は節と枝からなるグラフのような離散的データを、その部分グラフなどの部分データを素性とベクトル化として捉えるための畳み込みカーネルを提案し、例えば蛋白質の 3 次元立体構造の解析にも利用されている。学術論文の引用関係のグラフ構造に着目し、畳み込みを利用することで、学術論文のカテゴリー推定性能向上を目指す研究が[6,11]などで始まった。

本稿では、引用元、引用先のキーワードも対象とする論文のキーワードとすることで、識別性能が向上することを示す。 [3,5,12] でベンチマークデータとして使われた cora,citeseer,pubmed の 3 つのデータセットを対象に性能評価を行った。cora,citeseer については現在知られている最高性能である[3]よりも高い識別性能、pubmed についてはほぼ同じ性能となった。

2. 引用グラフ

図 1 は、3 つの論文をノードとする引用関係のグラフを表している。論文 2 は論文 1 から引用されていて、論文 3 を引用している。ノードの上には各論文に現れるキーワードを示している。通常のベクトル化では、それらのキーワードで文書はベクトル化される。グラフ畳み込みでは、引用情報もベクトルの成分として利用される[3])。論文 2 の情報として、引用元情報(source)として論文 2 と、引用先情報(destination)として論文 3 があり、s:1、d:3 という仮想的なキーワードとしてベクトル化に利用される。本稿では、さらに、引用元、引用先の論文に出現するキーワードを、論文 2 のキーワードとして利用する。ただし、論文 2 そのものに出現するキーワードと区別するために、キーワードの前にタグとして記号 S,D を付ける。その結果、図 1 の論文 2 には、本来のキーワード、引用情報、引用元／先キーワード、という 3 種類のキーワードを持つことになる（表 1）。

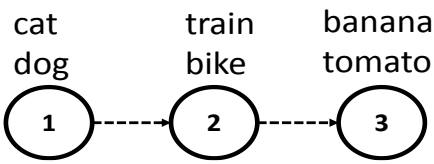

図 1 引用グラフ

表 1　畳み込みベクトル化

種類	ベクトル化キーワード
単語	train,bike
引用情報(conv.)	s:1,d:3
拡張引用情報(wide conv.)	S:cat,S:dog D:banana,D:tomato

3. 実験データと実験手順

本論文では、[3]と同様に、3 つデータセット cora,citeseer,pubmec について、提案手法の性能を評価した（表 2）。[3]では、各データセットからランダムに 1000 件をテストデータとして選んでいる。次に各カテゴリーについてテストデータ以外からランダムに文書を 20 件選び、それらをトレーニングデータとして機械学習でモデルを作り、そのモデルをテストデータに適用して accuracy を求めている。本稿では、これを 5 回繰り返し平均を求めた。

表 2　実験データ

dataset	node	cat	train		test
			labled	unlabeled	
citeseer	3327	6	6*20=120	2207	1000
cora	2708	7	7*20=140	1568	1000
pubmed	19717	3	3*20= 60	18657	1000

[3]では、カテゴリーごとの 20 件を合せたデータとテストデータを除いた残りのデータについて、カテゴリーのラベルを外した unlabeled データも学習データとして利用している。ところが、[13]では、unlabeled データを使わない方が性能が高と示されており、本研究でも unlabeled データを利用しない方が性能が高かった。そこで本稿では、unlabeled データは学習には使わない結果を示す。また、正例が少数なので学習データ水増し[2]を適用した。

4. 実験結果

表３、図２に各手法での Accuracy の比較を示す。表 3 において、Baseline は、文書のベクトル化で、各論文に出現する単語だけを使っときの性能を表す。Convolution は、単語の他に、その論文を引用している論文とその論文が引用している論文の番号も、その文書に出現する単語と同様に使ってベクトル化した。Wide Convolution は、さらに、引用あるいは被引用論文に現れる単語も、その論文の単語としてベクトル化した。ただし、もともとその論文に出現する単語と区別するため、引用文に単語が wi が出現している場合には s:wi、被引用文に出現している場合には d:wi としてもとの論文に出現している単語と区別した。表３、図２において、「Feature Selection」および「no Feature Selection」は[10]の属性選択を適用した場合と、適用しなかった場合を示す。表 3 の下半分は、[10]の属性選択の手法を適用した場合の最適属性選択の結果を表す。最後の列は、[3]の性能を示す

表 3　データセットごとの識別性能

		cora	citeseer	pubmed
noFS	Baseline	0.8711	0.8183	*0.7870
	Conv.	0.8746	0.8041	0.7852
	W.Conv.	*0.8939	0.7896	0.7836
FS	Baseline	0.8719	*0.8296	0.7856
	Conv.	0.8855	**0.8457	0.7852
	W.Conv.	**0.9083	0.8293	0.7842
Gao2018		0.833	0.730	**0.790

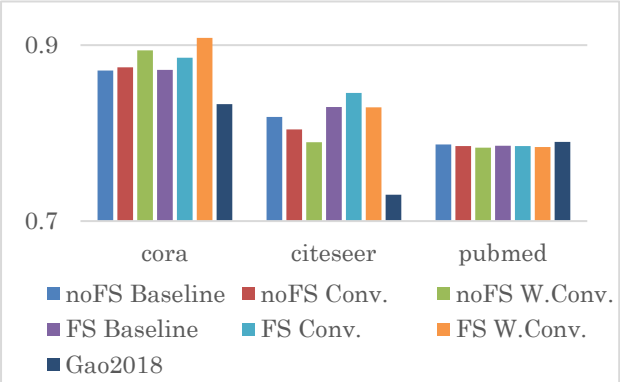

図 2　cora についのて識別性能

各データセットの列において、最高性能の手法に**、第 2 位の手法に*の印を付けている。cora と citeseer については、引用先や引用元の論文中の単語も利用する提案手法が、[3]の結果よりもよくなっている。cora については 7%、citeseer については、12%の性能向上が達成できている。pubmed については、ほとんど同じレベルである。

表 4 に Cora の各カテゴリーについての実験結果(accuracy)を示す。Genetic_Algorithms のカテゴリー以外については、単語も含むグラフ畳み込みを行った提案手法が、素朴な手法(BL)よりも 1%から 13%よくなっている。

5. まとめと今後の課題

本稿では、引用先、引用元の論文中のキーワードも利用するグラフ畳み込みのベクトル化を提案した。先行研究で使われている 3 つのデータセットで評価を行い、cora では 7%、siteseer では 12%の向上があり、pubmed では同じレベルとなった。文献情報へのグラフ畳み込みの利用は、関連研究調査や文献データベースなどへの応用が期待されている[1,9]。

表 4　cora についての識別性能

category	no Feature Selection			Feature Selection		
	BL	Conv.	W.Conv	BL	Conv.	W.Conv
Neural_Networks	0.7703	0.7903	*0.8343	0.7712	0.7900	**0.8351
Probabilistic_Methods	0.8708	0.8851	*0.8981	0.8705	0.8805	**0.9008
Genetic_Algorithms	0.9233	0.9251	0.9105	*0.9336	**0.9391	0.9302
Theory	0.8127	0.8372	**0.8898	0.8139	0.8375	*0.8751
Case_Based	0.9034	0.9147	0.8291	0.9023	*0.9144	**0.9219
Reinforcement_Learning	0.8985	0.9334	*0.9494	0.8969	0.9336	**0.9517
Rule_Learning	0.9186	0.8366	**0.9461	0.9151	0.9032	*0.9435
average	0.8711	0.8746	*0.8939	0.8719	0.8855	**0.9083

参考文献

[1] J. Chen, H. Zhuge, 2019, Automatic generation of related work through summarizing citations, Concurrency Computation, Vol.31, No.3 ,e4261
[2] T.G.Dietterich, R.H.Lathrop, T. Lozano-Perez, 1997, Solving the multiple instance problem with axis-parallel rectangles Artifical Inteligence, Vol. 89, No.1-1, pp.31-71
[3] H. Gao, Z. Wang, S. Ji, 2018, Large-scale learnable graph convolutional networks, Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp.p1416-1424
[4] D. Haussler, Convolution Kernels on Discrete Structures, 1999, Technical Report UCSC-CRL-99-10, University of Calfornia in Santa uze
[5] T. N. Kipf, M. Welling, 2017, Semi-Supervised Classification with Graph Convolutional Networks, Publishd as a conference paper at ICLR2017
[6] A. Krizhevsky, I. Sutskever, G. E. Hinton, 2012, Imagenet classification with deep convolutional neural networks, Proceeding of NIPS'12 Proceedings of the 25th International Conferenceon Neural Information Processing Systems , Vol.1, pp.1097-1105
[7] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, 1998, Gradient-based learning applied to document recognition, Proceedings of the IEEE, Vol.86, Issue 11 , pp.2278-2324
[8] Q. Lu, L. Getoor, 2003, Link-based Classification, ICML'03 Proceedings of the Twentieth International Conference on International Conference on Machine Learning pp. 496-503
[9] T. Saier, M. Farber, 2019, Bibliometric-enhanced arXiv: A data set for paper-based and citation-based tasks, CEUR Workshop Proceedings 2345, pp. 14-26T.
[10] Sakai, S. Hirokawa, 2012, Feature words that classify problem sentence in scientific article, Proceedings of 14th International Conference on Information Integration and Web-Based Applications and Services, pp. 360-367
[11] P. Sen, G. Namata, M. Bilgic, L. Getoor, B. Gallagher, T. Eliassi-Rad, 2008, Gollective Classifiation in Network Data, AI Magazine, Vol.29, No.3
[12] Z. Yang, W.W. Cohen, R. Salakhutdinov, 2016, Revisiting semi-supervised learning with graph embeddings, ICML'16 Proceedings of the 33rd International Conference on International Conference on Machine Learning - Vol. 48, pp. 40-48
[13] A. Zubiaga, V. Fresno, R. Martinez, 2009, Is unlabeled data suitable for multiclass SVM-based web page classification?, Proceedings of the NAACL HLT 2009 Workshop on Semi-Supervised Learning for Natural Language Processing, Association for Computational Linguistics, pp.28-36