Web Page Classification based on BERT-Geom

Guochen Yan (youth_49@163.com) **Guichao Zhu** (guichaozhu@163.com)

Abstract

Web page classification is one of the main tasks making information on a website in computer-understandable form, enabling much more sophisticated information retrieval and problem-solving. To tackle this problem, we propose BERT-Geom, which is a hybrid model inspired by natural language text classification and graph-based methods. Our model performs well on big datasets and achieves improvement on small datasets with the graph structure, which indicates the feasibility of conducting web page classification in an NLP way and the contribution of relation information to this task. We also find that for disassortative graphs, neighborhoods in latent spaces benefit the accuracy much more than neighborhoods in graphs, which implies graph models with neighborhoods in latent spaces will perform better for tasks in disassortative graphs in general.

1 Background

1.1 Problem Statement

Web page classification is one of the main tasks making information on a website in computer-understandable form, enabling much more sophisticated information retrieval and problem-solving. Our research aims to develop a deep learning model for the tasks and answer the following questions.

1. Can web page classification be conducted as an sequence text classification task in NLP method?

2. How does relation information help the classification using graph based methods?

1.2 Literature Survey

In the early age of the research on web page classification, several traditional ML models were used to do classification, including Naive Bayes, first-order text classification (a graph based algorithm) and assemble learning [1].

Web pages usually contain large amount of natural language texts, which indicates the potential of NLP models in this field. BERT [2] is a pre-training deep bidirectional transformer purposed by Google for multiple downstream NLP tasks. With the given pre-trained parameters as initialization, BERT enables researchers to achieve SOTAs on their problems by only a few epochs of fine-tuning.

Geometric Graph Convolutional Network (Geom-GCN)[3] adopts geometric aggregation scheme to alleviate existing problems in previous Message-passing Neural Network (MPNN). It can achieve an improvement on classification in disassortative graphs by creating a suitable latent space.

2 Proposed Method

2.1 BERT

As a pre-trained deep learning model with remarkable performance on sequenct classification, BERT definitely has the potential on classifying web pages.

Preprint. Under review.

Compared with the unidirectional architectures before BERT, information from **both right and left context** can be fused into the representation during the training, which brings BERT better performance.

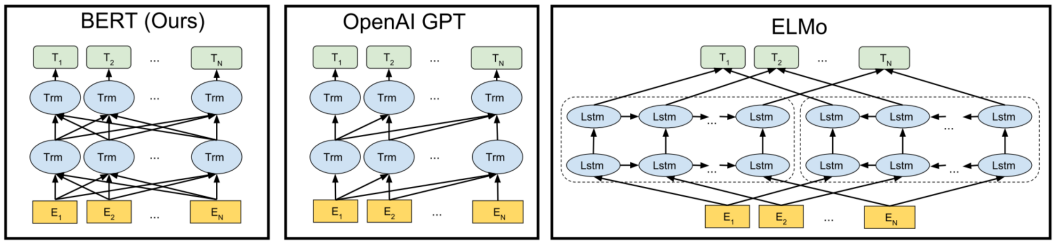


Figure 1: Innovations of BERT's bidirectional architecture. BERT uses a bidirectional Transformer. OpenAI GPT uses a left-to-right Transformer. ELMo uses the concatenation of independently trained left-to-right and right-to- left LSTMs to generate features for downstream tasks. Among the three, only BERT representations are jointly conditioned on both left and right context in all layers. In addition to the architecture differences, BERT and OpenAI GPT are fine-tuning approaches, while ELMo is a feature-based approach.

2.2 GeomGCN

The geometric aggregation scheme adopted by GeomGCN maps a graph into a continuous latent space via node embedding and utilizes neighborhoods both in graphs and in latent spaces for aggregation. This scheme extracts more structural information of the graph and can aggregate feature representations from distant nodes via mapping them to neighborhoods defined in the latent space, which can alleviate the following problems in MPNN:

- The aggregators lose the structural information of nodes in neighborhoods
- The aggregators lack the ability to capture long-range dependencies in disassortative graphs

In disassortative graphs where two nodes with the same label are not usually connected, GeomGCN can achieve better performance than previous graph models such as GCNs.

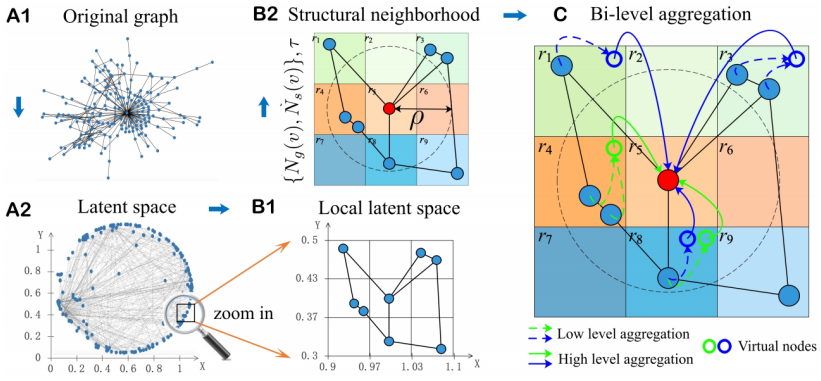


Figure 2: An illustration for geometry aggregation scheme. **A1-A2** The original graph is mapped to a latent continuous space. **B1-B2** The structural neighborhood. All adjacent nodes lie in a small region around a center node. The neighborhood in the graph contains all adjacent nodes in graph; the neighborhood in the latent space contains the nodes within the dashed circle whose radius is ρ. The relational operator τ is illustrated by a colorful 3×3 grid where each unit is corresponding to a geometric relationship to the red target node. **C** Bi-level aggregation on the structural neighborhood. Blue and green arrows denote the aggregation on the neighborhood in the graph and the latent space, respectively

More specifically, this scheme consists of three modules: node embedding, structural neighborhoods and bi-level aggregation. In node embedding, nodes in graph are mapped into a continuous latent space with their features and structural information.

Structural neighborhood is the combination of the neighborhoods in the graph and the neighborhoods within a pre-given radius in the latent space. In this way, a node's neighborhoods can contain nodes which are far in the graph but have some similarity with it. A function in latent space is given to indicate the relationship of two nodes. In bi-level aggregation, the aggregator extracts information in structural neighborhood and updates the hidden features of nodes.

2.3 Bert-Geom

Since BERT is an NLP encoder which can only handle sequential non-structural data and the texts on web pages are presented in HTML format containing relation information like URL links, the model may be improved by integrating relation information into training.

We define 2 types of relation on graphs based on 1) URL links, 2) similarity of node embeddings in latent spaces. Under our definitions, the graphs we constructed from the datasets vary. Each node in our graphs has neighborhoods both in original graphs built by URL links and in latent spaces built by embedding. Two kinds of neighborhoods can offer two different kinds information in aggregation, which may benefit different types of tasks.

We concatenate BERT and GeomGCN into a whole model to utilize text and relation information collectively. Upstream BERT extracts features of each web page (node) and downstream GeomGCN utilizes nodes features along with relation information to classify nodes.

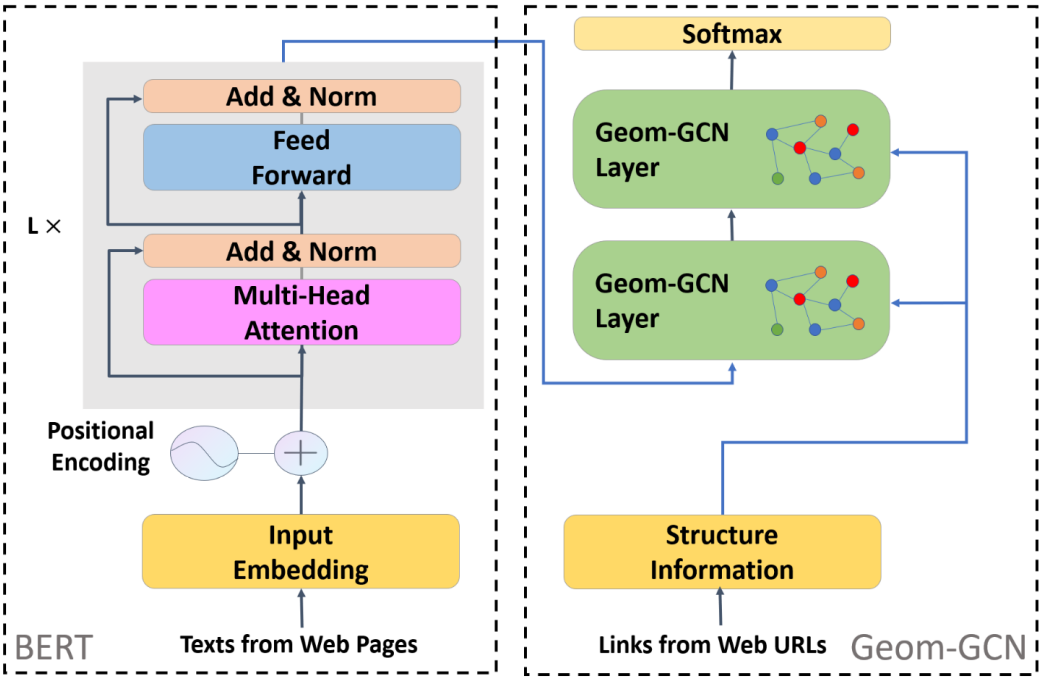


Figure 3: Architecture of BERT-Geom

3 Plan & Experiment

3.1 Dataset

WebKB data set contains web pages collected from computer science departments of various universities in January 1997 by the World Wide Knowledge Base (Web->Kb) project of the CMU text learning group. The 8,282 pages in WebKB were manually classified into the following categories:

student (1641), faculty (1124), staff (137), department (182), course (930), project (504), other (3764).

For each class the dataset contains pages from universities mainly including Cornell (867), Texas (827), Wisconsin (1263), etc. The graphs constructed by inter-linkages are disassortative graphs. We use an index denoted by β to measure the disassortativity in a graph:

$$\alpha = \frac{1}{|V|} \sum_{v \in \mathcal{V}} \frac{\text{Number of v's neighbors who have different labels from v}}{\text{Number of v's neighbors}}$$

and Table 1 shows that graphs constructed by WebKB datasets are disassortative.

Table 1: α values

	Cornell	*Texas*	*Wisconsin*
α	0.89	0.94	0.84

3.2 Experiment Setup

Accuracy is used as the metric for evaluating different architectures in our experiments. Pytorch is used as our experiments testbed.

The details of dataset pre-processing and two groups of experiment conducted are as below.

3.3 Data Pre-processing

The HTML full texts in the WebKB are pre-processed by removing noninformative contents including template front matters for configuration, HTML elements and all punctuations.

To construct connected graphs under the requirements of GCNs, we drop isolated nodes and connected components which do not connect to most of the nodes. As a result, we have 183, 183, 251 nodes in connected components of *Cornell*, *Texas* and *Wisconsin* respectively. And the rest of the data in the datasets, which cannot be added into the graphs, are gathered as the fourth sub-dataset called *Complement*.

A five-category classification on *student*, *faculty*, *course*, *project* and *staff* is conducted in this experiment. The *department* class is the most imbalanced categories with 182 samples overall and only 1 sample for each graph. The class *other* is a collection of pages that were not deemed the "main page" representing an instance of the previous six classes, which may introduce great amounts of noise into training. Therefore, most of the previous research choose to preserve only the rest five classes for classification, which is also our decision.

We split each sub-dataset into training set, validation set and testing set in the ratio of $6:2:2$.

3.3.1 BERT on Text Only

BERT with a linear layer at the end is used as the classifier. To prevent such tiny datasets from overfitting, we choose BERT-Tiny ($L = 2, H = 128$) as our upstream model. We fine-tune the whole model pre-trained parameters for BERT and random initialization for linear layer.

Table 2: Hyperparameters of Exp.1

Batch size	Number of epochs	Learning rate	Optimizer
16	400	5E-6	Adam

3.3.2 BERT-Geom

Motivated by literature and our pre-experiments, we adopt GeomGCN with neighborhoods only in latent spaces (Geom-L, see Appendix A). We concatenate such GeomGCNs and BERT into a whole

model and name it as BERT-Geom. We fine-tune BERT-Geom on three connected graphs subdatasets and we hybrid BERT to finish classification on the rest unconnected nodes.

We use BERT-Tiny memtioned in the BERT on Text Only section as our upstream model and two layers of GeomGCNs as our downstream model. Hyperparameter settings are shown in Table 3.

Table 3: Hyperparameters of Exp.2

Dropout	BERT lr	GCN (Linear) lr	Optim	Weight decay	Hidden size
0.5	5E-5	5E-2	Adam	1E-3	32

4 Results

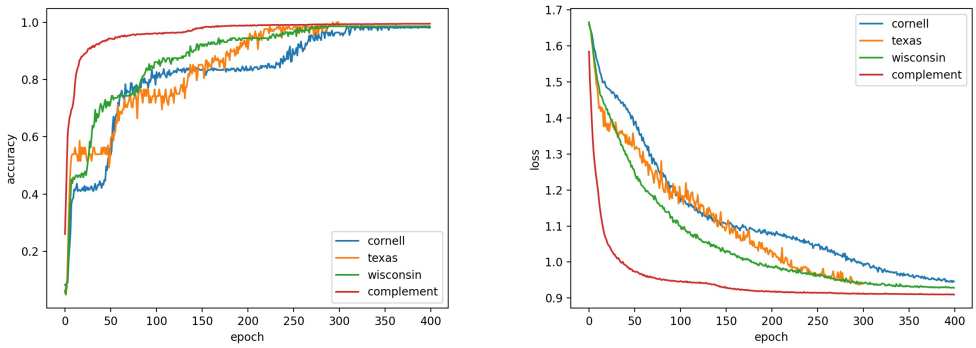
4.1 BERT on Text Only

The model achieved 90.15% accuracy on *Complement* with few tricks, indicating that it is feasible to conduct web page classification using sequence (text) classification methods in NLP when having enough amount of data. However, the accuracy drops rapidly when reducing the scale of data on the other three sub-datasets.

Table 4: Classification Accuracy of BERT on text only (Percent)

Model	*Cornell*	*Texas*	*Wisconsin*	*Complement*
BERT on Text Only	63.89	72.92	78.44	90.15

As the training history in Figure 4 shown, BERT learns quickly with big enough given training data. And the confusion matrix in Figure 5 shows apparent misclassification preference to staff, which is a category with imbalance on scale of samples.



(a) Accuracy curve

(b) Loss curve

Figure 4: Training history of BERT on text only. Red line shows that on big training set the pre-trained BERT as feature extractor enabled the model learning from the data and reached performance near the best in first several epochs. But other lines indicates that the lack of training data leads to instability and slows down the learning process.

4.2 BERT-Geom

We fine-tune BERT-Geom on three connected graphs subdatasets (*Cornell*, *Texas* and *Wisconsin*) and hybrid BERT to finish classification on the rest unconnected nodes (*Complement*).

To evaluate the overall performance, we take a weighted average of accuracy based on the number of nodes in connected graphs and the number of isolated nodes. Table 5 shows that BERT-Geom achieves an improvement on each connected graphs dataset and overall performance.

Table 5: Classification Accuracy (Percent)

Model	*Cornell*	*Texas*	*Wisconsin*	*Complement*	Average
BERT	63.89	72.92	78.44	90.15	87.74
BERT-Geom	**74.29**	**80.56**	**82.22**	90.15	**88.68**

5 Conclusions

The BERT on text only experiment shows the feasibility of conducting web page classification in sequence text classification way with enough given training data. With pre-trained BERT, few epochs of fine-tuning can achieve high accuracy.

BERT-Geom fuses text and relation information, benefiting the classification accuracy in WebKB datasets. Also, our experiment shows that neighborhoods in latent spaces make a contribution to the classification, which implies that graph models with relation defined as similarity in latent spaces are powerful and can be concatenated with existing models to improve the performance in disassortative graphs.

References

[1] M. Craven, Dan DiPasquo, Dayne Freitag, A. McCallum, Tom Michael Mitchell, K. Nigam, and Seán Slattery. Learning to construct knowledge bases from the world wide web. *Artif. Intell.*, 118:69–113, 2000.

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[3] Hongbin Pei, Bingzhen Wei, K. Chang, Yu Lei, and Bo Yang. Geom-gcn: Geometric graph convolutional networks. *ArXiv*, abs/2002.05287, 2020.

Appendices

A Pre-experiments of BERT-Geom

We concatenate 1) two layers of Geom-GCN and BERT into a whole model and name it as BERT-Geom, 2) two fully connected layers and BERT into a whole model and name it as BERT-Linear. And we conduct the following experiments.

- Fine-tune Bert-GCN with neighborhoods in the graph only.
- Fine-tune BERT-Geom with neighborhoods in both graph and latent space (BERT-Geom-GL).
- Fine-tune BERT-Geom with neighborhoods in latent space only (BERT-Geom-L).
- Fine-tune BERT-Linear with only full text (BERT-Linear).

We use Poincare as our embedding method and specify the dimension of embedding space as two. We generate neighborhoods in latent space by embedding text features (bag-of-word) of each node into the latent space.

The hyperparameters settings are shown in Table 6.

Table 6: Hyperparameters of Models

Model	Dropout	BERT lr	GCN (Linear) lr	Optim	Weight decay	Hidden size
BERT-GCN	0	5E-5	5E-2	Adam	5E-5	32
BERT-Geom-GL	0.5	5E-5	5E-2	Adam	1E-3	32
BERT-Geom-L	0.5	5E-5	5E-2	Adam	1E-3	32
BERT-Linear	0	5E-5	5E-3	Adam	1E-3	32×9

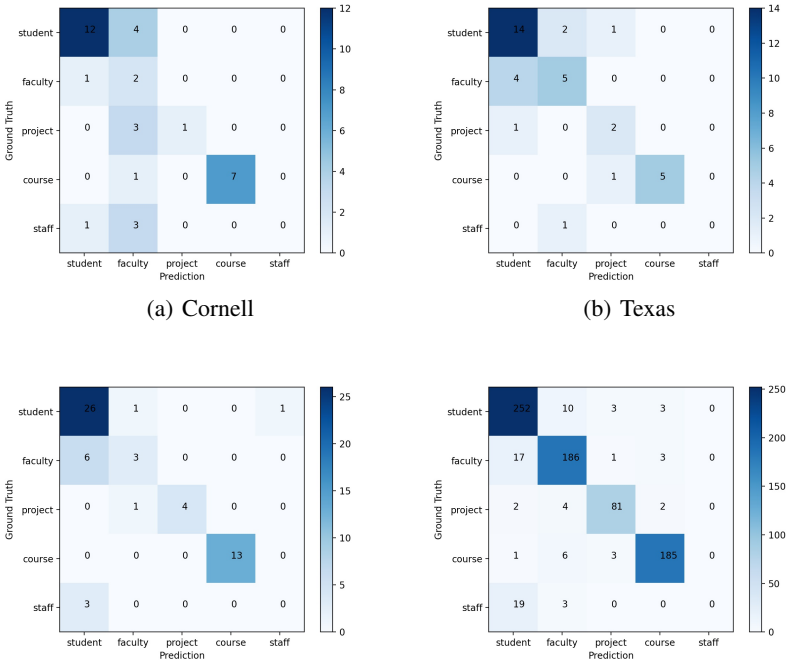
The accuracy of four models on *Cornell*, *Texas* and *Wisconsin* is shown in Table 7.

Table 7: Classification Accuracy (Percent)

model	*Cornell*	*Texas*	*Wisconsin*
BERT-GCN	40.00	44.44	33.33
BERT-Geom-GL	71.43	63.89	77.78
BERT-Geom-L	**74.29**	**80.56**	**82.22**
BERT-Linear	71.43	**77.78**	80.00

Table 7 shows that neighborhoods in the latent space has more contribution to accuracy, and neighborhoods in the graph limits the performance, which imply that use Geom-L will be a better choice in the following experiments.

B Confusion Matrices of BERT on Text Only



(a) Cornell (b) Texas

(c) Wisconsin (d) Complement

Figure 5: Confusion matrices on testing sets of BERT on text only