Distribution Agreement

In presenting this thesis or dissertation as a partial fulfillment of the requirements for an advanced degree from Emory University, I hereby grant to Emory University and its agents the non-exclusive license to archive, make accessible, and display my thesis or dissertation in whole or in part in all forms of media, now or hereafter known, including display on the world wide web. I understand that I may select some access restrictions as part of the online submission of this thesis or dissertation. I retain all ownership rights to the copyright of the thesis or dissertation. I also retain the right to use in future works (such as articles or books) all or part of this thesis or dissertation.

Signature:

Xiangjue Dong	Date

Analysis of Graph-Based Semi-Structured Categorical Model for

Competence-Level Classification

By

Xiangjue Dong
Master of Science

Computer Science

Jinho D. Choi
Advisor

Abeed Sarker
Committee Member

Carl Yang
Committee Member

Accepted:

Lisa A. Tedesco, Ph.D.
Dean of the James T. Laney School of Graduate Studies

Date

Analysis of Graph-Based Semi-Structured Categorical Model for Competence-Level Classification

By

Xiangjue Dong

M.S., University of Illinois, Urbana-Champaign, 2019

Advisor : Jinho D. Choi, Ph.D.

An abstract of

A thesis submitted to the Faculty of the

James T. Laney School of Graduate Studies of Emory University

in partial fulfillment of the requirements for the degree of

Master of Science

in Computer Science

2021

Abstract

Transformer-based models have been widely used for many natural language processing tasks and shown excellent capability in capturing contextual information, especially for document classification. Many existing transformer-based methods, however, even treat semi-structured text data as a block of text. These methods tend to ignore the hierarchical information and semantic correlations hidden in semi-structured text data, which can be captured by graph-based network models. This paper proposes a novel graph representation of semi-structured resume data that considers the categorical and hierarchical relationship in resumes. Our experiments show that our graph-based models outperform transformer methods for resume classification tasks and show better interpretability and generalization.

Analysis of Graph-Based Semi-Structured Categorical Model for Competence-Level Classification

By

Xiangjue Dong

M.S., University of Illinois, Urbana-Champaign, 2019

Advisor : Jinho D. Choi, Ph.D.

A thesis submitted to the Faculty of the

James T. Laney School of Graduate Studies of Emory University

in partial fulfillment of the requirements for the degree of

Master of Science

in Computer Science

2021

Acknowledgements

First and foremost, I am extremely grateful to my advisor Dr. Jinho D. Choi, committee members, Dr. Abeed Sarker and Dr. Carl Yang, for all their helpful advice, unwavering support, and patience during my M.S. study and Ph.D. applications. Their extensive knowledge and diligence motivate me in all the time of my academic research. In addition, I would like to extend my sincere thanks to all of the Department faculty and staff for their help and support. I gratefully acknowledge the financial support from Georgia Research Alliance Grant, Laney Graduate Scholarship - Tuition, and my parents. I would also like to thank my friends for their encouragement and continuous support throughout my life and education. Special thanks to Siqi for the company during the COVID-19 pandemic. The completion of my master's study would not have been possible without them. Finally, I would like to thank myself for not giving up. I hope everything will go well in the future.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Text classification is one of the fundamental problems in natural language processing area which has been widely studied. With the popularity of attention [10], transformer-based models achieve excellent performance on many tasks [1]. One of the important applications is to help Human Resource (HR) minimize recruiting time while maximizing proper matches. Traditional approaches rely on string/regex matching. Recently, [5] proposes context-aware transformer models which can encode the entire resumes.

However, compared to unstructured plain text data, the semi-structured data contains rich relational information, which can not be captured by transformer-based models effectively [14]. Recently, Graph Neural Networks have been extensively used in tasks which have rich relational structures because of the capability of preserving global structure information of a graph [13]. Zhang et al. [14] proposed a graph representation of semi-

structured Web data for the question answering task. Lu et al. [7] concatenated the input sentence embeddings with the vocabulary graph embedding to combine the strengths of GCN and BERT models.

In this paper, we aim to solve the resume classification task over semi-structured data and make the following contributions. First, we systematically categorize different components in semi-structured resume data, including context, category, and section, as well as their relations, including section-category relation and category-context relation. Then, we propose a graph-based categorical model over semi-structured data, which consists of three components: category relational graph construction, context encoder, and graph classifier (Chapter 4). We apply different context encoder and turn document classification problem into a graph classification problem. Compared with the state-of-the-art transformer-based methods, our experiments show that our graph-based model has better interpretability and generalization. It not only captures semantic and syntactic information but also preserves the structural relations in semi-structured data.

Chapter 2

Background

The competence-level classification task is a multiclass classification task where we need to decide whether the applicants are suitable for the level of positions they apply for based on the given resumes. In the previous work, [5] use tools to convert all the resumes into the unstructured text format, TXT, first. Then, the custom regular expressions are used to segment resumes into six sections, Profile, Education, Work Experience, Activities, Skills, and Others. After that, they proposes four systems to solve this task. The first one is section trimming, which appends all trimmed sections in order with the special token to represent the entire resume. Although part of every section is encoded, it can not guarantee that the trimmed range includes all necessary features. Then, the rule-based section pruning method is proposed. To make the model see the entire resume, each resume is segmented uniformly into multiple chunks and each chunk is encoded separately (chunk segmenting).

Furthermore, they encode the information that which sections the chunks belong to (section encoding).

In recent years, applying Graph Neural Networks to text classification task has attracted wide attention. Yao et al. [13] turned text classification problem into a node classification problem by constructing a single large graph from an entire corpus based on word co-occurrence and document word relations and then applying it into a GCN model which is initialized with one-hot representation for word and document. Huang et al. [3] built text-level graphs with global parameters sharing instead of a single graph for the whole corpus to reduce the memory consumption. Similarly, Ding et al. [2] proposed to model text documents with document-level hypergraphs and fed them into the hypergraph attention networks to learn discriminative text representations, which can also obtain more expressive power with less computational consumption. To extract deeper text features, Bidirectional Long Short-Term Memory (BiLSTM) was introduced and concatenated with the POS information to eliminate the lexical polysemy problem [9]. In addition, Liu et al. [6] proposed a text graph tensor to harmonize and integrate heterogeneous information from different kinds of graphs and Yang et al. [12] proposed to learn to generate concept maps within an end-to-end Neural

Network model. By constructing a categorical graph for each semi-structured resume, our model also turns document classification problem into a graph classification problem.

Chapter 3

Dataset

3.1 Data Processing

Between April 2018 and August 2020, the department of Human Resources (HR) at Emory university received 40,946 applications for 374 Clinical Research Coordinator (CRC) positions. The applications contain resumes in different formats, such as `DOC`, `PDF`, `TXT`, and `RTF`. The CRC positions aim to initiate and manage clinical research studies and contain four levels, CRC I, CRC II, CRC III, and CRC IV, while CRC IV is the most professional. Table 3.1 gives the detailed descriptions about four CRC levels [5].

Table 3.2 shows the statistics of the applications and resumes categorized into these levels. Out of the 40,946 applications, 90% of them are applied for the entry level positions, `CRC I-II`, which makes sense because `CRC III-IV` positions require higher requirements (`A`). In addition, because there are

Type	Description
CRC I	Manage administrative activities associated with the conduct of clinical trials. Maintain data pertaining to research projects, complete source documents/case report forms, and perform data entry. Assist with participant scheduling.
CRC II	Manage research project databases and development study related documents, and complete source documents and case report forms. Interface with research participants and study sponsors, determine eligibility, and consent study participants according to protocol.
CRC III	Independently manage key aspects of a large clinical trial or all aspects of one or more small trials or research projects. Train and provide guidance to less experienced staffs, interface with research participants, and resolve issues related to study protocols. Interact with study sponsors, monitor/report SAEs, and resolve study queries. Provide leadership in determining, recommending, and implementing improvements to policies and procedures.
CRC IV	Function as a team lead to recruit, orient, and supervise research staff. Independently manage the most complex research administration activities associated with the conduct of clinical trials. Determine effective strategies for promoting/recruiting research participants and retaining participants in long term clinical trials.

Table 3.1: Descriptions of the four-levels of CRC positions [5].

various same level positions for different division (e.g., digestive disease, infectious disease, cardiology), the same applicant may apply for multiple job postings with the same CRC level. After removing the duplicated resumes in each level, 67% of the applications are discarded and 13,317 resumes remain (C). Moreover, the same applicant may also apply for positions across multiple CRC levels. Thus, the duplicated resumes across different levels are removed and only the resumes applied for the highest level are remained. For example, if an applicant applied for both CRC II and CRC III positions, only the resume for CRC III was retrained. Then, additional 10% of the original applications are discarded and 9,156 resumes are preserved (D).

	CRC I	CRC II	CRC III	CRC IV	**Total**
A	23,658	13,176	3,246	866	40,946
B	56	157	45	16	374
C	6,855	4,315	1,454	693	13,317
D	4,154	3,222	1,087	693	9,156
E	1,477	1,172	542	234	3,425
F	690	540	180	90	1,500
G	2,167	1,712	722	324	4,925

Table 3.2: The statistics of the applications categorized into four-levels of CRC positions. A: the counts of applications. B: CRC positions. C: unique resumes for each level. D: unique resumes across all levels. E: resumes previously annotated in D and used in [5]. F: resumes newly annotated in D. G: total resumes of E and F.

[5] carefully select 3,425 resumes from D while retain the same ratios of the CRC levels (E). E is also used for our experiments to compare our results with

the state-of-the-art results in [5]. In addition, we select 1,500 more resumes using the same strategy (F) and get 4,925 resumes in total (G) to see whether enlarging the size of dataset would benefit our model.

The resumes in D are parsed by *RChilli* [1] into a structured JSON format, so advanced deep learning models can be developed for semi-structured text classification.

3.2 Annotation

An annotation team of experts with prior experience in hiring applicants for CRC positions design the annotation guidelines in 5 rounds and annotate the 3,425 resumes in Table 3.2.E with one of the 4 CRC levels, CRC I-IV, or *Not Qualified* (NQ). Thus, this annotation uses 5 labels in total. In addition, the Fleiss Kappa score measured for the inter-annotator agreement (ITA) reaches 60.8% among 3 annotators after 5 rounds, which indicate the high quality annotation [5]. Moreover, the new added resumes in Table 3.2.F are annotated using the same strategies. Thus, a total of 4,925 resumes are annotated for our study.

[1] RChilli Resume Parser API: https://www.rchilli.com/solutions/resumeparser-api

Chapter 4

Approach

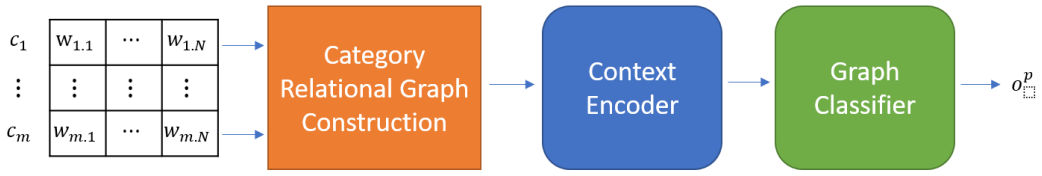

Figure 4.1: Graph-based semi-structured categorical model structure.

In this section, we proposed a graph-based categorical model of semi-structured data to address the multi-class resume classification task. By given a resume, this task is to decide which level of CRC position that the corresponding applicant is suitable for. Figure 4.1 shows the overview of our model structure. It consists three components: category relational graph construction component, context encoder component, and graph classifier component.

4.1 Graph Construction

The resume data (Chapter 3) is parsed into various categories, e.g., *Institute Name*, *Degree*, *Specialization*, *Duration*, and segmented into six sections, *Education*, *Work Experience*, *Publication*, *Hobbies*, *Objectives*, and *Achievements*. For each section, it has multiple categories which have the corresponding context. Table 4.1 shows the context under eight main categories and two main sections in the semi-structured resume data.

	Context	Category	Section
1	Portland State University	Institute Name	EDU
2	Bachelor of Science	Degree	EDU
3	Psychology	Specialization	EDU
4	4	Duration	EDU
5	Children's Healthcare	Employer	WE
6	Research Coordinator	Title	WE
7	2	Duration	WE
8	Coordinates the conduct...	Job Description	WE

Table 4.1: Example of part of semi-structured resume data. EDU: Education, WE: Work Experience.

To construct the categorical resume graph, we consider the Section-Category Relation and Category-Context Relation. Section is often a summary of a block of context that people use in their resumes. Category often outlines the classes that the context belongs to. Thus, a Category-Context Relation is usually a class-instance relation.

Given a semi-structured resume data example in Table 4.1, we construct the graph based on the components and their hierarchical relations described above. To better cluster these categories, we add the intermediate node between the section node and category node. For example, in the main section `EDU`, if applicant attended two universities, there would be two intermediate node named "School" to represent the education experience separately. For the context which doesn't have the corresponding category, it will link to the section node directly.

Figure 4.2 illustrates the graph constitution in Table 4.1: section node, intermediate node, category node, and context node.

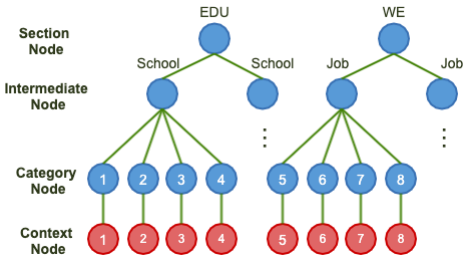

Figure 4.2: Graph constructed from semi-structured resume data in Table 4.1. The numbers shown on category nodes and context nodes are corresponding to the numbers listed in Table 4.1.`EDU`: Education, `WE`: Work Experience.

Following the two structural relations shown in the data, three types of edges are created in the graph: Section-Category Relation: edges between section nodes and intermediate nodes; edges between intermediate nodes and

category nodes; Category-Context Relation: edges between category nodes and context nodes. These structural relations carry the inherent semantic relations between the components and represent the context in the semi-structured data better.

4.2 Context Encoder

For the context s_m under category m, where s_m is a sequence of words, i.e., $s_m = (w_{m.1}, \ldots, w_{m.N})$, the context encoder converts each token of the context into a embedding matrix. Then, the phrase embedding or sentence embedding is generated by averaging the word embeddings.

4.3 Graph Classifier

After constructing the resume undirected graph, we feed the graph into a Graph Convolutional Network (GCN) [4], which is formulated as

$$L^{(i+1)} = \sigma(\widetilde{D}^{-\frac{1}{2}}\widetilde{A}\widetilde{D}^{-\frac{1}{2}}L^{(i)}W^{(i)}),$$

where i denotes the layer number, σ is the non-linear activation function, e.g. a ReLU $\sigma(x) = max(0, x)$ in our case, $\widetilde{A}$ is the adjacent matrix of the graph with additional self-connections, $\widetilde{D}$ is the degree matrix where $\widetilde{D}_{ii} = \sum_j \widetilde{A}_{ij}$,

and $W^{(i)}$ is the learnable weight matrix of the i-th layer of GCN. The input node features $L^{(i)}$ are represented by the context embeddings of nodes and the graph-level embedding is derived by a mean pooling operation on the node-level embedding output. Then, an MLP is attached to produce the predicted label $\hat{y}$, that is, $\hat{y} = MLP(Pooling(L^{(i+1)}))$. In addition, the cross-entropy loss $\sum_{d_i \in \mathcal{D}} -p(\hat{y}_i) \log p(y_i)$ is computed over all labeled resumes.

Chapter 5

Experiments

5.1 Dataset

Experiments were conducted on the resume dataset from the competence-level classification task in [5]. This dataset comprises 3,425 resumes annotated with 5 levels of real Clinical Research Coordinator (CRC) positions. By keeping similar label distributions across all sets and following the experiment setting in [5], 3,425 selected resumes are split into the training (`TRN`), development (`DEV`), and test (`TST`) sets with the ratios of 75:10:15. Then each resume is parsed into semi-structured format through the resume parsing software Rchilli[1] and segmented into the six sections, *Education*, *Work Experience*, *Publication*, *Hobbies*, *Objectives*, and *Achievements* with various categories (example shown in Table 4.1). Most resumes consistently contain the *Work Experience* and *Education* sections, whereas the others are often missing.

[1] https://www.rchilli.com/

In addition, some context under certain categories or sections, e.g., *Job Description*, *Publications*, are long sentences while the others are usually short phrases. The statistics of the data are shown in Table 5.1.

	NG	**AN**	**AE**	**AD**
TRN	2565	54	53	1.96
DEV	344	76	78	2.05
TST	516	25	23	1.84

Table 5.1: Statistics of training (TRN), development (DEV), and test (TST) sets. **NG**: number of graphs; **AN**: average number of nodes; **AE**: average number of edges; **AD**: average degree.

5.2 Experimental Setups

After an extensive hyper-parameter search, the statistics of hyper-parameters used in the proposed GCN models are shown in Table 5.2. Additionally, we set the maximum sequence length to 128 for the transformer encoder. Different seed values are used for the three runs and the average accuracy on development set and test set are calculated.

BS	**LR**	**E**	**HC**	**DR**
64	1e-3	1000	300	0.6

Table 5.2: Statistics of hyper-parameters. **BS**: batch size; **LR**: learning rate; **E**: number of training epochs; **HC**: size of hidden channel of GCN; **DR**: dropout rate.

5.3 Models

In this section, we illustrated the context-aware transformer-based model which is used as our baseline model and our proposed graph-based semi-structured categorical model with various graph initialization.

The state-of-the-art context-aware transformer model using chunk segmenting and section encoding from [5] for long document classification is used as our baseline model. Different from the data format in that paper, our resume data is pre-processed into semi-structured. Thus, we modified and implemented two types of baseline models, the fine-grained model (`FG`) and the coarse-grained model (`CG`). Following their experiment settings, the BERT-base model [1] is used as the transformer encoder.

Fine-Grained Model (`FG-BERT`): The input is the phrase context $\{r_{i,j}, \ldots, r_{i,jN}\}$ under each category, which is prepended by the special token $c_{i,j}$ and fed into the transformer encoder (`TE`) that generates the embedding list $\{e^c_{i,j}, \ldots, e^r_{i,jN}\}$. Then, list of category IDs $\{k_1, \ldots, k_m\}$ is fed into the category encoder (`CE`), an embedding layer, to generate the category embedding list $\{e^k_1, \ldots, e^k_m\}$. Finally, $e^{c+k}_{\sum}$, where $e^c_{\sum} = \sum_{\forall i \forall j} e^c_{i,j}$, is fed into the Linear Decoder (`LD`) that generates the output vector to make the prediction.

Coarse-Grained Model (`CG-BERT`): Different from `FG`, the context under

each category in the same section are concatenated together as the input. And the list of six section IDs is fed into the section encoder.

The proposed Graph-based Semi-Structured Categorical Model is illustrated in Section 4. We use two different ways to obtain the initial representation of graph nodes in our model (Section 4.2): GloVe embedding and transformer encoder.

GloVe Embedding: The input node embeddings $L^{(0)}$ are phrase embeddings, which are the average of the concatenated pretrained GloVe word embeddings [8] with dimension of 300. Then the represented graphs are fed into GCN to predict the label (`GloVe-GCN`). As a comparison, We also fed the graphs into a different graph classifier, the GAT [11] model (`GloVe-GAT`).

Transformer Encoder: We use the BERT-base model [1] as the transformer encoder (`BERT-GCN`). The input context $\{w_{m,1}, \ldots, w_{m,N}\}$ under category m is prepended by the special token c_m and fed into the transformer encoder to generate a embedding matrix and the phrase embedding or sentence embedding is generated by the last hidden states through the pooling layer. Furthermore, we fine-tuned the BERT model for language modeling on the resume data using a masked language modeling loss (`lm-BERT-GCN`).

5.4 Results

Table 5.3 shows the average accuracy and standard deviation on the development and test sets. Our graph-based categorical model `lm-BERT-Graph` gives the highest accuracy on the test set, showing 7% over the baseline model `FG-BERT`. Although `GloVe-GAT` performs best on `DEV`, it may be overfitting on `TST`. We also explored different pooling stategies for `lm-BERT-GCN` by taking the `CLS` embedding of the phrase or sentence as the node embedding. It gives lower average accuracy, which is about 0.6802 on `DEV` and 0.7229 on `TST`. The `CLS` may not learn well due to the length difference between the phrases and sentences. The improvement on `TST` shows that the structural relations learned from the semi-structure data by graph-based categorical model benefits the classification task.

	DEV	TST
FG-BERT	66.70 (±0.18)	69.40 (±0.30)
CG-BERT	70.57 (±0.50)	72.63 (±1.62)
GloVe-GCN	68.99 (±0.34)	73.64 (±0.34)
GloVe-GAT	**70.74** (±0.44)	70.87 (±0.11)
BERT-GCN	69.19 (±0.50)	74.10 (±0.30)
lm-BERT-GCN	69.00 (±0.17)	**74.29** (±0.40)

Table 5.3: Average accuracy ± standard deviation (%) on the development (`DEV`) and test (`TST`) sets.

After adding addition 1,500 resumes to the `TRN`, the comparison results of

our model on the same DEV and TST are shown in Table 5.4. The added new data doesn't improve the results significantly.

	DEV	TST	DEV	TST
GloVe-GCN	68.99 (±0.34)	**73.64** (±0.34)	**70.64** (±0.29)	72.48 (±0.34)
BERT-GCN	69.19 (±0.50)	74.10 (±0.30)	**69.77** (±0.29)	**74.35** (±0.12)
lm-BERT-GCN	**69.00** (±0.17)	**74.29** (±0.40)	67.72 (±0.28)	72.53 (±0.76)

Table 5.4: Average accuracy ± standard deviation (%) on the development (DEV) and test (TST) sets.

5.5 Error Analysis

After analyzing 100 resumes where the predicted labels are not correct, we found that 46 of them are due to overestimation (e.g., a resume rated as NQ is labeled as CRCI) and 54 of them are because of underestimation (e.g., a resume rated as CRCI is labeled as NQ). The detailed statistics are shown in Table 5.5, where 40.74% of CRC II resumes are underestimated as CRC I and 52.17% of NQ resumes are overestimated as CRC I. In addition, compared the results with the annotation guidelines, we can see the adjacent positions are difficult to be distinguished. For example, the majority of requirements for the adjacent CRC positions, CRC I and CRC II are quite similar, but they have different requirements for the number of years on research experience.

U: True - Predicted	No.	O: True - Predicted	No.
CRC I - NQ	13	NQ - CRC I	24
CRC II - CRC I	22	CRC I - CRC II	3
CRC III - CRC II	1	CRC II - CRC III	11
CRC IV - CRC III	4	CRC I - CRC III	8
CRC III - CRC I	14	-	-
Total	54	Total	46

Table 5.5: Error analysis on TST. U: Underestimated resumes. O: Overestimated resumes.

Chapter 6

Conclusion

This paper proposes a novel graph representation of semi-structured data considering the categorical and hierarchical relationship and treated document classification tasks as graph classification tasks. Our experiments show that our graph-based semi-structured categorical model outperforms the state-of-the-art transformer-based model for competence-level classification tasks and the structural relation which is learned from semi-structured data improves the model interpretability and generalization.

Bibliography

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186, 2019. URL `https://www.aclweb.org/anthology/N19-1423`.

[2] Kaize Ding, Jianling Wang, Jundong Li, Dingcheng Li, and Huan Liu. Be More with Less: Hypergraph Attention Networks for Inductive Text Classification. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4927–4936, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.399. URL `https://www.aclweb.org/anthology/2020.emnlp-main.399`.

[3] Lianzhe Huang, Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. Text Level Graph Neural Network for Text Classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3444–3450, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1345. URL `https://www.aclweb.org/anthology/D19-1345`.

[4] Thomas Kipf and M. Welling. Semi-Supervised Classification with Graph Convolutional Networks. *ICLR*, 2017.

[5] Changmao Li, Elaine M. Fisher, Rebecca Thomas, S. Pittard, V. Hertzberg, and Jinho D. Choi. Competence-Level Prediction and Resume-Job_Description Matching Using Context-Aware Transformer Models. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 8456–8466, 2020.

[6] Xien Liu, Xinxin You, Xiao Zhang, Ji Wu, and Ping Lv. Tensor Graph Convolutional Networks for Text Classification. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):

8409–8416, Apr. 2020. doi: 10.1609/aaai.v34i05.6359. URL https://ojs.aaai.org/index.php/AAAI/article/view/6359.

[7] Zhibin Lu, Pan Du, and Jian-Yun Nie. VGCN-BERT: Augmenting BERT with Graph Embedding for Text Classification. In Joemon M. Jose, Emine Yilmaz, João Magalhães, Pablo Castells, Nicola Ferro, Mário J. Silva, and Flávio Martins, editors, *Advances in Information Retrieval*, pages 369–382, Cham, 2020. Springer International Publishing. ISBN 978-3-030-45439-5.

[8] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. GloVe: Global Vectors for Word Representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014. URL http://www.aclweb.org/anthology/D14-1162.

[9] H. Tang, Y. Mi, F. Xue, and Y. Cao. An Integration Model Based on Graph Convolutional Network for Text Classification. *IEEE Access*, 8: 148865–148876, 2020. doi: 10.1109/ACCESS.2020.3015770.

[10] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, L. Kaiser, and Illia Polosukhin. Attention is All you Need. *NIPS*, 2017.

[11] Petar Velickovic, Guillem Cucurull, A. Casanova, A. Romero, P. Liò, and Yoshua Bengio. Graph Attention Networks. *ICLR*, abs/1710.10903, 2018.

[12] Carl Yang, Jieyu Zhang, Haonan Wang, Bangzheng Li, and Jiawei Han. *Neural Concept Map Generation for Effective Document Classification with Interpretable Structured Summarization*, page 1629–1632. Association for Computing Machinery, New York, NY, USA, 2020. ISBN 9781450380164. URL `https://doi.org/10.1145/3397271.3401312`.

[13] Liang Yao, Chengsheng Mao, and Yuan Luo. Graph Convolutional Networks for Text Classification. In *AAAI*, 2019.

[14] Xingyao Zhang, Linjun Shou, Jian Pei, Ming Gong, Lijie Wen, and Daxin Jiang. A Graph Representation of Semi-structured Data for Web Question Answering. In *COLING*, 2020.