Syntactic Graph Co-attention Network for Automatic Short Answer Grading

Anonymous Submission

Abstract

In this work, we addressed the problem of Automatic Short Answer Grading (ASAG). The task involves assigning a grade to a student's answer by comparing it against a model answer for a given question. Previous works in this domain mostly used rule-based and machine learning methods to tackle the problem, wherein the creation of handcrafted features and the use of neural networks have been the most common practice. Different variations of syntactic and semantic similarity between a student and model answer pair have been used as features in earlier works. We hypothesize that the extent of alignment between the graph representations of a student and model answer is a good indicator of their relative similarity. In this direction, we propose an end-to-end ASAG system that models the alignment as co-attention between the nodes in the dependency graphs corresponding to an answer pair. We leveraged the representational power of BERT and Graph Convolutional Network (GCN) along with a co-attention mechanism to capture the intrinsic similarities between student answers and reference answers. Our proposed method surpasses most of the existing state-of-the-art results on the SemEval-2013 SciEntsBank and BEETLE datasets.

1 Introduction

Quick evaluation and grading along with feedback from the instructor can help students to work upon their mistakes and thus to move up on the learning curve. This becomes an exhausting task as evaluator has to manual screen through each of the answer and then score it. Also, due to differences in learning strategies, cognitive capacities, and knowledge levels, students may convey the same response in multiple ways, which makes it more difficult for the evaluator. Further, manual grading of the responses can be erroneous and may inculcate instructor bias. One way to mitigate these myriad of challenges is to automatically grade the student answers. Though natural language and free-text based responses are very difficult to evaluate, recent advancements in the domain of Natural Language Processing (NLP) has made this grading process feasible.

Automatic Short Answer Grading (ASAG) is viewed as a classification or regression task in most of the existing literature. The research in this domain gained momentum with comprehensive benchmark dataset, namely, SemEval-2013 (Dzikovska et al., 2013). The existing approaches rely on traditional machine learning techniques with handcrafted features (Mohler et al., 2011). Several handcrafted features have been employed in the earlier works in the form of POS tag, n-gram features, context overlap features (Heilman and Madnani, 2013) (Ott et al., 2013). Subsequently, deep learning techniques like Long Short Term Memory Networks aka LSTMs and Convolutional Neural Networks aka CNNs became prevalent (Alikaniotis et al., 2016) (Hassan et al., 2018) (Huang et al., 2018) (Kumar et al., 2017) (Riordan et al., 2017) (Yang et al., 2018). Both the lines of research indicate the reliance of the models on the measure of similarity between the input student answer and the corresponding model or reference answer. The representational ability of the deep learning models have been shown to be more effective (Peng et al., 2018).

Pre-trained Language Models (PLMs) have been extremely successful in crossing benchmarks on multiple NLP sub-tasks by fine-tuning them with task-based or domain specific data. Lun et al. (2020); Ghavidel. et al. (2020) in their papers showed that the transformer based models perform extremely well on the benchmark dataset for SemEval short answer grading. Sung et al. (2019) showed that task-specific fine-tuning on enhanced PLMs achieve much better performance for ASAG task. Camus and Filighera (2020) in their paper demonstrated that large Transformer-based pre-trained models achieve state-of-the-art results in ASAG. Ndukwe et al. (2020) utilised Sentence-BERT, to perform automatic grading of three variations of short answer questions. Recently, the use of Graph Convolutional Networks (GCNs) in NLP tasks has gained attention and there have been promising results and crossing of benchmarks in many NLP based sub-tasks (Marcheggiani and Titov, 2017), (Sahu et al., 2019), (Zhang et al., 2018). Zhang et al. (2018) proposed a one-of-a-kind model made up of: a CNN-based instance encoder, a graph convolutional network and a knowledge-aware attention for ASAG. Very recently, relation networks (Li et al., 2021) have been used to capture three-way relation between questions, reference answers, and student answers.

While the existing works leveraged textual similarity between a student answer (SA) and a reference answer (RA), we hypothesize that their similarity can be captured in both textual (words) and structural (dependency graph) domain. In earlier works, the representations of SA and RA have been obtained using independent components (e.g., two parallel LSTMs) of the architecture. We challenge this view by considering an architecture that learns joint representation of SA and RA. In this work, we present a novel approach for automatic evaluation of student answers by employing a co-attention based Graph Neural Network architecture to jointly learn representation of the SA and the MA from their dependency graph. Following are the key contributions of our paper:

1. The automated short answer grading problem has been modelled as a graph representation learning problem.

2. A joint feature learning method has been considered using a co-attention based graph neural network architecture that captures both textual and structural similarity of a given answer pair.

2 Proposed Approach

Syntactic structures are useful for cross-domain generalisation of NLP models as has been found in literature and previous study (Wang et al., 2017). Thus, encoding structural information into the model could make the model more robust. Following the above lines and inspired by (He et al., 2020)(Lu et al., 2016) we built an improvised architecture which could effectively capture syntactics and semantics of the answer pair along with added co-attention to effectively represent the word-alignments between the pair of student answer and reference answer.

Given a pair $\langle SA, RA \rangle$, we obtain the dependency graphs of each sentence by using a neural parser. Word level contextual representation of the raw SA and RA pairs are obtained using BERT or Bi-LSTM models. The syntactic dependency trees thus obtained have words as the nodes with their corresponding embedding representation as node features. To facilitate joint representation, the SA and RA graphs are combined by adding alignment edges between all pairs of nodes, one from the SA, other from the RA. Further, for each dependency edge, a reverse edge is added and for each node a self loop is added. The resulted graph is then passed through a sequence of GCN layers, followed by co-attention matching layer that captures relative attention between word pairs from SA and RA. The co-attention pooled output representations SA and RA are then passed through output softmax layer for the final prediction. The schematics of the architecture is presented in Fig. 1.

GCN module

Graph Convolutional Networks (GCNs) efficiently use dependency paths to transform and transmit path information, and updates node embeddings by effectively combining the transmitted information. Consecutive such p GCN transformations cause the information to propagate through the neighborhood nodes of order p.

Here, the feature vector of node t is updated at the p^{th} layer by:

$$h_{\text{t}}{}^{(p+1)} = g\big(\sum_{u \epsilon N(t)} \phi^{(p)} h_{\text{u}}{}^{(p)} + b^{(p)}\big) \tag{1}$$

where $g(.)$ is ReLU activation function and $N(.)$ is neighbourhood function.

Co-attention Layer

If $\mathcal{S}$ and $\mathcal{R}$ symbols are used to denote the node representations of dependency graph for SA and RA respectively in the GCN output. Then an affinity matrix reflecting the contingent alignment of the words in SA and that of RA is calculated as follows

$$\mathcal{A} = \tanh(\mathcal{S}^{\top} W_c \mathcal{R}) \tag{2}$$

This affinity matrix is then further used to calculate the directional co-attention maps from SA to RA and reverse:

$$\mathcal{M}_S = \tanh(W_A \mathcal{S} + \mathcal{A}^{\top}(W_B \mathcal{R})) \tag{3}$$

$$\mathcal{M}_R = \tanh(W_B \mathcal{R} + \mathcal{A}^{\top}(W_A \mathcal{S})) \tag{4}$$

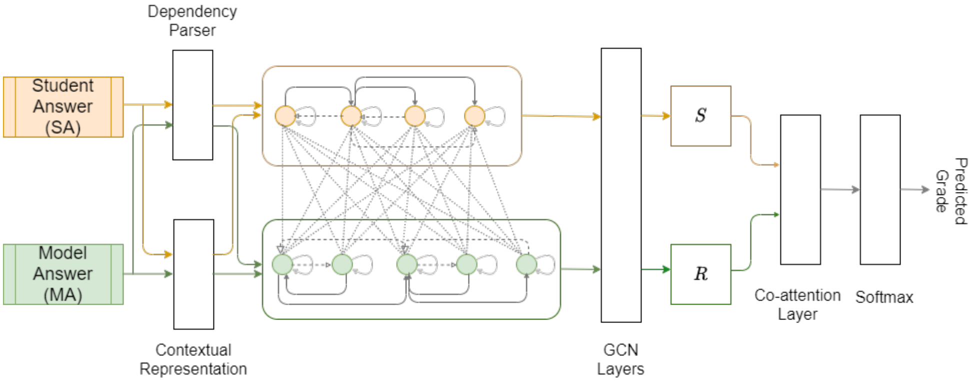

Figure 1: Co-attention coupled GCN architecture for ASAG. $\mathcal{S}$ and $\mathcal{R}$ represent the matrix representations of GCN node embedding of SA and RA respectively.

The attention weights corresponding to SA and RA are computed as follows:

$$\alpha_S = \texttt{softmax}(a_A^\top \mathcal{M}_S) \tag{5}$$

$$\alpha_R = \texttt{softmax}(a_B^\top \mathcal{M}_R) \tag{6}$$

Weight parameters are denoted by W_A, W_B, a_A, a_B and a_A and a_B store the attention weights of words in SA and RA respective;y. Finally, we calculate the vector representations of SA or RA as:

$$h_{SA} = \sum_{w_n \in SA} \alpha_S^{(n)} \mathcal{S}^{(n)}, \;\; h_{RA} = \sum_{w_m \in RA} \alpha_R^{(n)} \mathcal{R}^{(m)} \tag{7}$$

here n^{th} entry in α is denoted by $\alpha^{(n)}$, n^{th} column in $\mathcal{X}$ is denoted by $\mathcal{X}^{(n)}$.

Output Softmax Layer

The final classification output is obtained by concatenating vector representations of the input answer pair with their element-wise difference and multiplication as $[h_{SA},\ h_{RA},\ h_{SA} \bullet h_{RA},\ h_{SA} - h_{RA}]$, which is then forwarded to a linear layer with softmax activation. The final model is trained using a cross-entropy loss.

3 Implementation Details

The PyTorch implementations of BERT-base and GCN were leveraged in our experiments. The BERT models were initialized with the same pretrained weights and their baselines were optimised using the Adam optimizer (Loshchilov and Hutter (2018)). The BERT embedding size and GCN encoder output dimension were considered to be 768. The number of GCN layers was set to 5 with dropout of 0.2. The size of the directional co-attention maps calculated in the co-attention layer has been set to 512. The learning rate was set to 5×10^{-3} for the experiments on both SciEntsBank and BEETLE. We train our models on SciEntsBank and BEETLE datasets. Following standard, evaluation instances in SciEntsBank are segmented into three categories, namely , unseen aswer (UA), unseen question (UQ) and unseen domain (UD) whereas those in BEETLE are segmented into two : UA and UQ. The test set has been so designed to test the efficacy and generalization ability of the trained model.

4 Results and Discussions

The SemEval-2013 BEETLE and SciEntsBank datasets (Dzikovska et al., 2013) have been used in this study. We report the results of 2-way, 3-way and 5-way classification tasks related to the SemEval-13 dataset (Dzikovska et al., 2013). Accuracy and Macro F1 (M-F1) are used as the evaluation metrics. As baselines, representatives of different appraoches towards ASAGA have been considered: 1) Lexical Overlap (LO) (Dzikovska et al., 2013), 2) ETS$_2$ (Heilman and Madnani, 2013), 3) CoMeT (Ott et al., 2013), 4) TF+SF (handcrafted feature + sentence embedding) (Saha et al., 2018), 5) LR+BERT (logistic regression with pre-trained BERT) (Sung et al., 2019), 6) SFRN+(Relation network with BERT encoder) (Li et al., 2021). The models proposed by us are named as GASAG*. The '+' symbol in the model name indicates availability of the alignment linkage between student answer and model answer dependency graphs. The performance values are presented in Table 1. The following observations can be made:

1) Across all the classification levels (2-way, 3-way and 5-way), our proposed method has outperformed the baselines with some exceptions. Out of

ASAG	**SciEntsBank**						**BEETLE**			
Models	**Acc**			**M-F1**			**Acc**		**M-F1**	
	UA	**UQ**	**UD**	**UA**	**UQ**	**UD**	**UA**	**UQ**	**UA**	**UQ**
2-WAY CLASSIFICATION PERFORMANCE										
Lexical Overlap(LO)	66	66	67	61	63	65	79	75	78	72
ETS_2	72	71	69	70	68	68	81	74	80	72
CoMeT	77	60	67	76	57	67	83	70	83	69
TF+SF	**79**	70	71	**78**	68	70	-	-	-	-
LR+BERT	70	59	57	70	57	53	82	67	82	65
SFRN+	78	64	67	70	64	67	89	70	**89**	70
GASAG-LSTM	61	64	55	60	64	47	67	63	67	61
GASAG-BERT	74	71	67	73	70	66	**83**	73	83	70
GASAG-LSTM+	64	63	60	64	63	57	75	67	74	62
GASAG-BERT+	78	**73**	**71**	77	**72**	**71**	82	**75**	82	**74**
3-WAY CLASSIFICATION PERFORMANCE										
Lexical Overlap (LO)	55	54	51	40	39	41	60	51	55	47
ETS_2	72	62	**62**	64	42	42	63	55	59	52
CoMeT	71	54	57	64	38	40	73	51	71	46
TF+SF	71	**65**	64	65	48	45	-	-	-	-
LR+BERT	67	52	54	60	42	42	73	60	64	52
SFRN+	**73**	56	58	65	49	47	78	63	67	55
GASAG-LSTM	45	45	39	43	40	33	72	62	62	57
GASAG-BERT	69	58	55	67	**56**	53	83	**76**	76	**68**
GASAG-LSTM+	53	46	42	50	39	37	77	72	64	58
GASAG-BERT+	71	58	56	**70**	**56**	**54**	**85**	72	**77**	62
5-WAY CLASSIFICATION PERFORMANCE										
Lexical Overlap (LO)	43	41	41	37	32	31	51	48	42	41
ETS_2	62	**66**	**63**	**58**	27	39	71	62	61	55
CoMeT	60	43	42	55	20	15	68	56	48	30
TF+SF	62	50	50	47	31	35	-	-	-	-
LR+BERT	61	42	47	45	30	25	69	57	55	51
SFRN+	**69**	47	51	47	35	35	75	56	60	55
GASAG-LSTM	53	39	40	39	29	26	60	61	42	43
GASAG-BERT	67	51	48	52	44	42	76	64	70	63
GASAG-LSTM+	54	42	42	42	31	35	73	66	64	60
GASAG-BERT+	68	53	50	53	**48**	**54**	**77**	**69**	**71**	**66**

Table 1: Performance comparison between the proposed model (GASAG*) and the state-of-art ASAG models. The shaded cells along with boldface represent best performance values.

11 combinations (test item type and performance measure), our proposed method has emerged as winner in 7 cases for 2-way, in 8 cases for 3-way and in 6 cases for the 5-way task.
2) With some minor exceptions, establishing alignment between the dependency graphs has helped in improving the performance of the grading model irrespective of the context representation method (LSTM or BERT). Thus inclusion of co-attention helps in improving the performance of the grading model in general. This validates our hypothesis that joint representation learning of the answer graphs via adding alignment edges helps in developing better grading model.
3) It is also observed that choice of BERT contextual representation over LSTM encoder improves the performance by considerable margin in majority of the cases.

5 Conclusion and Limitation

In this paper, we aimed at exploring the effect of joint representation learning of a given pair of answer graphs for the ASAG task. We have used graph co-attention network to facilitate the proposed joint representation learning. The co-attention mechanism has been implemented on top of GCN-based transformation of aligned dependency graphs corresponding to an input answer pair. It is observed the the inclusion of co-attention has significant positive impact in the performance of the grading model.

As compared to traditional text similarity-based measures, our method relies on dependency graphs of the answer sentences. This limits the applicability of our method for the languages for which dependency parser has not been developed. In many cases, the student answers are ill formed syntactically. This may lead to erroneous dependency graphs and consequently erroneous grading.

References

Dimitrios Alikaniotis, Helen Yannakoudakis, and Marek Rei. 2016. Automatic Text Scoring Using Neural Networks. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 715–725, Berlin, Germany. Association for Computational Linguistics.

Leon Camus and Anna Filighera. 2020. Investigating transformers for automatic short answer grading. In *Artificial Intelligence in Education*, pages 43–48, Cham. Springer International Publishing.

Myroslava O. Dzikovska, Rodney D. Nielsen, Chris Brew, Claudia Leacock, Danilo Giampiccolo, Luisa Bentivogli, Peter Clark, Ido Dagan, and Hoa Trang Dang. 2013. Semeval-2013 task 7: The joint student response analysis and 8th recognizing textual entailment challenge. In *Proceedings of the 7th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2013, Atlanta, Georgia, USA, June 14-15, 2013*, pages 263–274. The Association for Computer Linguistics.

Hadi Ghavidel., Amal Zouaq., and Michel Desmarais. 2020. Using bert and xlnet for the automatic short answer grading task. In *Proceedings of the 12th International Conference on Computer Supported Education - Volume 1: CSEDU,*, pages 58–67. INSTICC, SciTePress.

Sarah Hassan, Aly A. Fahmy, and Mohammad El-Ramly. 2018. Automatic Short Answer Scoring based on Paragraph Embeddings. *International Journal of Advanced Computer Science and Applications*, 9(10). Publisher: The Science and Information Organization.

Qi He, Han Wang, and Yue Zhang. 2020. Enhancing Generalization in Natural Language Inference by Syntax. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4973–4978, Online. Association for Computational Linguistics.

Michael Heilman and Nitin Madnani. 2013. ETS: Domain adaptation and stacking for short answer scoring. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 275–279, Atlanta, Georgia, USA. Association for Computational Linguistics.

Yuwei Huang, Xi Yang, Fuzhen Zhuang, Lishan Zhang, and Shengquan Yu. 2018. Automatic Chinese Reading Comprehension Grading by LSTM with Knowledge Adaptation. In *Advances in Knowledge Discovery and Data Mining*, pages 118–129, Cham. Springer International Publishing.

Sachin Kumar, Soumen Chakrabarti, and Shourya Roy. 2017. Earth Mover's Distance Pooling over Siamese LSTMs for Automatic Short Answer Grading. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 2046–2052.

Zhaohui Li, Yajur Tomar, and Rebecca J. Passonneau. 2021. A semantic feature-wise transformation relation network for automatic short answer grading. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6030–6040, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2018. Fixing weight decay regularization in adam.

Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. 2016. Hierarchical Question-Image Co-Attention for Visual Question Answering. *CoRR*, abs/1606.00061. ArXiv: 1606.00061.

Jiaqi Lun, Jia Zhu, Yong Tang, and Min Yang. 2020. Multiple data augmentation strategies for improving performance on automatic short answer scoring. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(09):13389–13396.

Diego Marcheggiani and Ivan Titov. 2017. Encoding Sentences with Graph Convolutional Networks for Semantic Role Labeling. *CoRR*, abs/1703.04826. ArXiv: 1703.04826.

Michael Mohler, Razvan Bunescu, and Rada Mihalcea. 2011. Learning to Grade Short Answer Questions using Semantic Similarity Measures and Dependency Graph Alignments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 752–762, Portland, Oregon, USA. Association for Computational Linguistics.

Ifeanyi G. Ndukwe, Chukwudi E. Amadi, Larian M. Nkomo, and Ben K. Daniel. 2020. Automatic grading system using sentence-bert network. In *Artificial Intelligence in Education*, pages 224–227, Cham. Springer International Publishing.

Niels Ott, Ramon Ziai, Michael Hahn, and Detmar Meurers. 2013. CoMeT: Integrating different levels of linguistic modeling for meaning assessment. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 608–616, Atlanta, Georgia, USA. Association for Computational Linguistics.

Hao Peng, Jianxin Li, Yu He, Yaopeng Liu, Mengjiao Bao, Lihong Wang, Yangqiu Song, and Qiang Yang. 2018. Large-Scale Hierarchical Text Classification with Recursively Regularized Deep Graph-CNN. In *Proceedings of the 2018 World Wide Web Conference*, WWW '18, pages 1063–1072, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee. Event-place: Lyon, France.

Brian Riordan, Andrea Horbach, Aoife Cahill, Torsten Zesch, and Chong Min Lee. 2017. Investigating neural architectures for short answer scoring. pages 159–168.

Swarnadeep Saha, Tejas I. Dhamecha, Smit Marvaniya, Renuka Sindhgatta, and Bikram Sengupta. 2018. Sentence level or token level features for automatic short answer grading?: Use both. In *Artificial Intelligence in Education*, pages 503–517, Cham. Springer International Publishing.

Sunil Kumar Sahu, Fenia Christopoulou, Makoto Miwa, and Sophia Ananiadou. 2019. Inter-sentence Relation Extraction with Document-level Graph Convolutional Neural Network. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4309–4316, Florence, Italy. Association for Computational Linguistics.

Chul Sung, Tejas Dhamecha, Swarnadeep Saha, Tengfei Ma, Vinay Reddy, and Rishi Arora. 2019. Pre-training BERT on domain resources for short answer grading. 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 6071–6075, Hong Kong, China. Association for Computational Linguistics.

Liangguo Wang, Jing Jiang, Hai Leong Chieu, Chen Hui Ong, Dandan Song, and Lejian Liao. 2017. Can syntax help? improving an LSTM-based sentence compression model for new domains. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1385–1393, Vancouver, Canada. Association for Computational Linguistics.

Xi Yang, Yuwei Huang, Fuzhen Zhuang, Lishan Zhang, and Shengquan Yu. 2018. Automatic Chinese Short Answer Grading with Deep Autoencoder. In *Artificial Intelligence in Education*, pages 399–404, Cham. Springer International Publishing.

Yuhao Zhang, Peng Qi, and Christopher D. Manning. 2018. Graph Convolution over Pruned Dependency Trees Improves Relation Extraction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2205–2215, Brussels, Belgium. Association for Computational Linguistics.