Dynamic Graph Transformer for Implicit Tag Recognition

Yi-Ting Liou,[1] Chung-Chi Chen,[1] Hen-Hsen Huang,[2,3] Hsin-Hsi Chen[1,3]
[1] Department of Computer Science and Information Engineering
National Taiwan University, Taiwan
[2] Department of Computer Science, National Chengchi University, Taiwan
[3] MOST Joint Research Center for AI Technology and All Vista Healthcare, Taiwan
ytliou, cjchen@nlg.csie.ntu.edu.tw
hhhuang@nccu.edu.tw, hhchen@ntu.edu.tw

Abstract

Textual information extraction is a typical research topic in the NLP community. Several NLP tasks such as named entity recognition and relation extraction between entities have been well-studied in previous work. However, few works pay their attention to the implicit information. For example, a financial news article mentioned "Apple Inc." may be also related to Samsung, even though Samsung is not explicitly mentioned in this article. This work presents a novel dynamic graph transformer that distills the textual information and the entity relations on the fly. Experimental results confirm the effectiveness of our approach to implicit tag recognition.

1 Introduction

Documents on the web deliver and spread lots of most recent information to people worldwide. In order to automatically update the real-world information, textual information extraction is a fundamental issue for NLP researchers. Many downstream tasks such as fake news detection (Wu et al., 2019) and stock movement prediction (Peng and Jiang, 2016) can benefit from the extracted information. However, most of the previous works (Hu et al., 2018; Xu and Cohen, 2018) focus on using explicit information in articles and do not consider the implicit information. For example, two companies, Sprint and T-Mobile, are explicitly mentioned companies in the news article in Figure 1. However, the stock price of a non-mentioned company, SoftBank, may be influenced since Softbank owns shares of Sprint. Although this kind of inference is intuitive for professional analysts, few previous works take such implicit information into consideration. In this paper, we aim to increase the sense of machines toward this kind of implicit information.

Transformer-based (Vaswani et al., 2017) neural networks achieve state-of-the-art performances in many NLP tasks (Devlin et al., 2018; Malmi et al., 2019). To model the relationships between entities, graph neural network (GNN) is a well-known architecture for representing the knowledge and additional information (Fu et al., 2019; You et al., 2020). Furthermore, the models blending these two architectures show their effectiveness (Lu et al., 2020). In this paper, we propose dynamic graph transformer (DGT), a novel blend of Transformer and GNN. In previous work, the weights of the GNN are pre-determined in the training stage and not affected by the given input. Our DGT adjusts the weights depending on the input on the fly. In this way, the representation of the graph information will be more flexible and more specific to the input.

Title:
The Judge Nodded! *Sprint* Rise 70% in After-hour Trading.

News Article:
A federal judge gave his blessing to the US$26.5 billion merger between *T-Mobile* and *Sprint* on Feb. 11, several months after the deal got final antitrust approval from the U.S. government. *Sprint* surges 68.75%. *T-Mobile* rise 7.36%.

Related Stock:
Sprint, *T-Mobile*, **SoftBank**

Figure 1: An example of the implicit information in news articles.

A strategy for pre-training on in-domain data is further proposed. Experimental results show our approach is effective in the task of extracting the implicit information from news articles. The contributions of this work are summarized as follows.

- We point out an important issue of information extraction for implicit entities.
- We propose a novel model that dynamically incorporates textual information and graph in-

Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics, pages 1426–1431
April 19 - 23, 2021.

formation. Our approach outperforms recent works in implicit tag recognition.

- Our pre-training task, masked entity prediction, is helpful for predicting the implicit entities. The pre-trained model can be also applied in other information extraction tasks.

2 Related work

Extracting and using the information in articles is one of the focuses in the NLP community. Some works (Hu et al., 2018; Ding et al., 2019; Ma et al., 2019) adopt the extracted information for stock market prediction. Some of them (Baker et al., 2016; Min and Zhao, 2019) use the information in news articles to construct socio-economic indicators. Most of previous works only focus on explicit information in the articles. In this way, the implicit entities in the articles may be under looked. In this paper, we aim to extract the non-mentioned but related entities from a document.

Recently, GNN has become popular for modeling relationships among multiple entities. Kipf and Welling (2016) use the convolution neural network to learn the node representation by aggregating the features of neighboring nodes. Veličković et al. (2017) employ the attention mechanism to improve the GNN architecture. Recent studies (Berg et al., 2017; Monti et al., 2017; Ying et al., 2018) also show the effectiveness of graph neural networks in various tasks. Inspired by these works, we present a new blend of graph attention network (GAT) (Veličković et al., 2017) with Transformer (Vaswani et al., 2017) and BERT (Devlin et al., 2018) for extracting the implicitly related entities to a given article.

3 Method

3.1 Task Setting

The task is formulated as follows. Given an article, a model is aimed at predicting a list of entities that are not explicitly mentioned but related to the given article. Let a corpus be $\mathcal{D} = \{(\mathbf{d}^1, \mathbf{y}^1), (\mathbf{d}^2, \mathbf{y}^2), ..., (\mathbf{d}^{|\mathcal{D}|}, \mathbf{y}^{|\mathcal{D}|})\}$, where $\mathbf{d}^k$ and $\mathbf{y}^k$ denote k-th article and the implicit entity list of k-th article, respectively. The k-th article can be represented by a word sequence, i.e., $\mathbf{d}^k = (w_1^k, w_2^k, ..., w_{|\mathbf{d}^k|}^k)$. Let the candidate entity list be $\mathcal{C} = \{c_1, c_2, ..., c_{|\mathcal{C}|}\}$, where c_i denotes i-th entity, and $\mathbf{y}^k \in \{0,1\}^{|\mathcal{C}|}$, $\mathbf{y}_i^k = 1$ if the entity c_i is associated with the given article $\mathbf{d}^k$ but not directly mentioned within the content, otherwise $\mathbf{y}_i^k = 0$.

3.2 Graph Attentional Layer

We build a co-occurrence matrix $\mathbf{M}$ to represent the association graph of the entities in $\mathcal{C}$. The frequency of two entities c_i and c_j appearing together in the training corpus can be defined as follows.

$$\mathbf{M}_{i,j} = \sum_{k=1}^{|\mathcal{D}|} [\mathbf{y}_i^k = 1 \text{ and } \mathbf{y}_j^k = 1], \quad (1)$$

where $[\cdot]$ is the Iversion bracket. $\mathbf{M}$ can be viewed as an adjacency matrix representation of an association graph. $\mathbf{M}_{i,j}$ is the value of the edge between nodes i and j, which represents the degree of association between the entities c_i and c_j. The graph attentional layer is a component of the graph attention network (GAT) (Veličković et al., 2017). In order to suit the architecture of Transformers, we modify part of GAT as follows. Firstly, the h-th score matrix $\mathbf{S}^h \in \mathbb{R}^{n \times n}$ is defined as follows.

$$\mathbf{S}^h = \frac{\mathbf{X}\mathbf{W}_Q^h (\mathbf{X}\mathbf{W}_K^h)^\top}{\sqrt{d}}, \quad (2)$$

where $\mathbf{X} \in \mathbb{R}^{n \times d_{\text{input}}}$ denotes the input matrix, and $\mathbf{W}_Q^h \in \mathbb{R}^{d_{\text{input}} \times d}$ and $\mathbf{W}_K^h \in \mathbb{R}^{d_{\text{input}} \times d}$ are learnable matrices.

Secondly, the h-th multi-head attention matrix $\mathbf{A}^h \in \mathbb{R}^{n \times n}$ is defined as follows.

$$\mathbf{A}_{i,j}^h = \begin{cases} \frac{e^{\mathbf{S}_{i,j}^h}}{\sum_{k \in \mathcal{N}[i]} e^{\mathbf{S}_{i,k}^h}} & \text{if } j \in \mathcal{N}[i] \\ 0 & \text{otherwise.} \end{cases}, \quad (3)$$

where $\mathcal{N}[i] = \{j : \mathbf{M}_{i,j} > 0\}$ represents the closed neighborhood set of node i. Lastly, we concatenate all the computational results of multi-attention heads. The output of the graph attentional layer is computed as follows.

$$\text{GAL}(\mathbf{X}) = (\Big\|_{h=1}^{H} \mathbf{A}^h \mathbf{X} \mathbf{W}_V^h) \mathbf{W}_O, \quad (4)$$

where $\mathbf{W}_V^h \in \mathbb{R}^{d_{\text{input}} \times d}$ and $\mathbf{W}_O \in \mathbb{R}^{d_{\text{input}} \times d_{\text{input}}}$ are learnable matrices. H is the number of attention heads and $d_{\text{input}} = d \times H$.

3.3 Dynamic Graph Transformer

We first show Static Graph Transformer (SGT), which incorporates Transformer, BERT, and GAT. Then, we extend SGT to the final model, Dynamic Graph Transformer (DGT) by considering graph information dynamically.

Static Graph Transformer

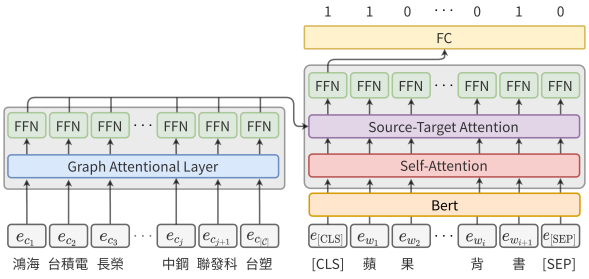
Figure 2a shows the architecture of SGT, which integrates graph and text information. The motivation behind SGT is to treat the Transformer encoder as a variation of GNN by replacing self-attention with a graph attentional layer. The Transformer encoder takes the entity's representation $\mathbf{e}_{c_j}$ as input. It is one of the row vectors in BERT's word embeddings $\mathbf{E} \in \mathbb{R}^{|\mathcal{V}| \times d_{\text{input}}}$, where $\mathcal{V}$ denotes the vocabulary of BERT. We consider the outputs of the Transformer encoder as the entity embeddings $\mathbf{n}_1, \mathbf{n}_2, ..., \mathbf{n}_{|\mathcal{C}|}$. In SGT, the process of generating entity embedding $\mathbf{n}_j$ is static because it is irrelevant to the content information from the input article. In the Transformer decoder, we take the contextual word embeddings as the inputs of the Transformer encoder. The contextual word embeddings are the last hidden state vectors of BERT, which use the word embedding $\mathbf{e}_{w_i}$ in the article as input. Finally, we follow the settings of Devlin et al. (2018) to use the first output embedding of the Transformer decoder for predicting the implicit entity list.

Dynamic Graph Transformer

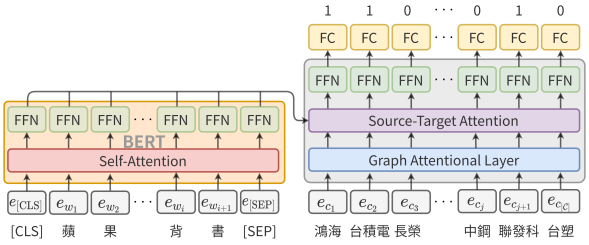
Figure 2b shows the architecture of DGT. From another perspective, since BERT is a kind of Transformer encoder, we move the learning part of entity embedding $\mathbf{n}_j$ to the Transformer decoder. We treat the last hidden state vectors of the Transformer decoder as entity embeddings $\mathbf{n}_1, \mathbf{n}_2, ..., \mathbf{n}_{|\mathcal{C}|}$. In this way, DGT is able to update the entity embedding on the fly because the source-target attention mechanism utilizes the outputs of BERT, making the model more tailored to the input article. In our task, each entity embedding is mapped to the scalar, which indicates the probability of the entity related to the article but not mentioned in it.

3.4 Pre-training by Masked Entity Prediction

Devlin et al. (2018) show that pre-training with masked language modeling and next sentence prediction is effective. Chu et al. (2020) indicate that pre-training with the value process prediction task is useful for generating correct numeric values in news headlines.



(a) Static Graph Transformer (SGT)



(b) Dynamic Graph Transformer (DGT)

Figure 2: Overview of SGT and DGT.

In this work, we propose a new pre-training task, masked entity prediction, to enrich the semantic information of entity names. Building on the original BERT vocabulary, we add a list of entity names in $\mathcal{C}$ and aim to learn their representations. We adopt the *bert-base-chinese* as the initial model and retrofit it on the training data with two sub-tasks at the same time. The first sub-task is a new masked language modeling task. Unlike the masked language model task performed for the original BERT, we not only mask the tokens using the method in Devlin et al. (2018) but also all the entity mentions in the document. The second sub-task is to label all masked entities on the position tagged as `[CLS]`. In this way, we obtain a new pre-trained model tailored to our target corpus.

4 Experiments

4.1 Dataset Description

The dataset[1] consists of 27,716 news articles collected from MoneyDJ[2], a financial newsvendor in Taiwan. Each news article is published with the labels of the related entities. These labels are annotated by professional journalists. The candidate entity list contains 735 company names, i.e., $|\mathcal{C}| = 735$. We split the dataset into the training set and the test set by time. The training set contains

[1]The dataset is available for academic usage by request: http://nlg.csie.ntu.edu.tw/nlpresource/FinTag/

[2]https://www.moneydj.com/

Model	Micro-F1	Macro-F1
ABC	67.11%	31.73%
TAB-LSTM	64.82%	30.47%
ITAG	60.06%	27.44%
BERT	71.72%	34.18%
VGCN-BERT	72.86%	35.65%
SGT	75.94%	42.94%
DGT	**76.16%**	**47.41%**

Table 1: Overall performances.

Model	Micro-F1	Macro-F1
SGT w/o pre-train	-1.67%	-3.78%
DGT w/o pre-train	-1.90%	-7.37%

Table 2: Performance degradation caused by the approach without the proposed pre-training task.

24,640 samples before March 24, 2017. The test set contains 3,076 samples from March 24, 2017.

4.2 Baseline Models

We adopt two kinds of models as our baselines, including the models for tag recommendation and the models for classification. The models ABC (Gong and Zhang, 2016), TAB-LSTM (Li et al., 2016), and ITAG (Tang et al., 2019) are considered as the baselines for tag recommendation. For the classification task, we adopt BERT (Devlin et al., 2018) and VGCN-BERT (Lu et al., 2020) for comparison.

4.3 Experimental Results

We adopt the binary cross-entropy as the loss function and the Adam optimizer (Kingma and Ba, 2014) for training. The learning rate and the batch size are 3e-5 and 8, respectively. Table 1 shows that the proposed models are better than the baseline models in both micro-F1 score and macro-F1 score. In Table 2, we further show the performance degradation when using the original BERT language model instead of the language model with the proposed pre-training task. The results confirm that the proposed pre-training task is useful in the implicit relation learning task.

5 Discussion

5.1 Attention Mechanism

Table 3a shows that the model focuses on the companies and the products that appear in the news article. It indicates that the proposed model captures the relationship between the companies via the related companies and the mentioned products. In Table 3b, we find that even no company names have been mentioned in the news article, DGT can still correctly infer the related entities by the mentioned raw material such as wheat, corn, and soybeans.

宏碁和華碩分別排名第六、第七
... Acer and ASUS rank sixth and seventh, respectively. ...
Surface在連網功能、官方配件
... In terms of Internet connectivity and official accessories, Microsoft Surface ...
Related entities: **HON HAI (鴻海), HTC (宏達電)**, TSMC (台積電), LARGAN (大力光), **CATCHER (可成), TXC (晶技)**

(a) Inferring non-mentioned companies via mentioned companies and products.

400公噸。小麥一週出口量(裝船)648,700公噸
... 400 MT. Weekly wheat (shipping) exports were 648,700 MT, which ...
往日本。美國玉米一週出口淨銷量為1,347,000公噸
... to Japan. Weekly net export sales of U.S. corn were 1,347,000 MT ...
每磅18美分。黃豆期貨的價格預估下調最多每英斗
… 18 cents per pound. Soybean futures prices are expected to fall by at most …
Related entities: **FWUSOW (福壽), UNI-PRESIDENT (統一), TTET (大統益), ST (興泰), TAIROUN (台榮), FOPCO (福懋油), GREATWALL (大成), CPE (卜蜂), TAISUN (泰山), LHIC (聯華), NAMCHOW (南僑), Wei Chuan (味全), S.F.C (佳格), AGV (愛之味)**

(b) Inferring non-mentioned companies only by mentioned products.

Table 3: Examples of attention weights. All related entities are labeled by professional journalists, and the bold entities represent the model predictions.

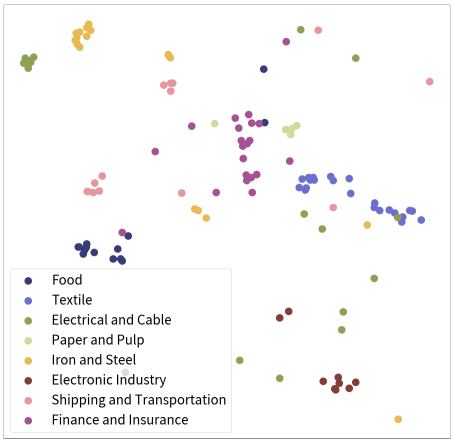


Figure 3: Visualization of entity embeddings by t-SNE.

5.2 Entity Embeddings

We use the last hidden state vectors of DGT as the corresponding entity embeddings and use t-SNE (van der Maaten and Hinton, 2008) to visualize these embeddings. As shown in Figure 3, we find that although we do not directly provide the information about the related industry or product of the entity during training, the model can still capture relationships from the corpus.

6 Conclusion and Future Work

This paper presents a novel dynamic graph transformer model and a pre-training task for extracting the implicit entities in articles. Experimental re-

sults show the usefulness of the proposed methods. We also discuss what kinds of features our model captures.

In our previous work (Liou et al., 2021), we apply the proposed task to accelerate the working process of the journalists and show that using the extracted entities could be useful for downstream tasks such as news aggregation and stock movement prediction. In the future, we plan to apply the proposed approach to datasets with both graphical knowledge and textual content.

Acknowledgments

This research was partially supported by Ministry of Science and Technology, Taiwan, under grants MOST 109-2218-E-009-014, MOST 109-2634-F-002-040, and MOST 109-2634-F-002-034.

References

Scott R Baker, Nicholas Bloom, and Steven J Davis. 2016. Measuring economic policy uncertainty. *The Quarterly Journal of Economics*, 131(4):1593–1636.

Rianne van den Berg, Thomas N Kipf, and Max Welling. 2017. Graph convolutional matrix completion. *arXiv preprint arXiv:1706.02263*.

Jui Chu, Chung-Chi Chen, Hen-Hsen Huang, and Hsin-Hsi Chen. 2020. Learning to generate correct numeric values in news headlines. In *Companion Proceedings of the Web Conference 2020*, pages 17–18.

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

Xiao Ding, Kuo Liao, Ting Liu, Zhongyang Li, and Junwen Duan. 2019. Event representation learning enhanced with external commonsense knowledge. 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 4894–4903, Hong Kong, China. Association for Computational Linguistics.

Tsu-Jui Fu, Peng-Hsuan Li, and Wei-Yun Ma. 2019. GraphRel: Modeling text as relational graphs for joint entity and relation extraction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1409–1418, Florence, Italy. Association for Computational Linguistics.

Yuyun Gong and Qi Zhang. 2016. Hashtag recommendation using attention-based convolutional neural network. In *IJCAI*, pages 2782–2788.

Ziniu Hu, Weiqing Liu, Jiang Bian, Xuanzhe Liu, and Tie-Yan Liu. 2018. Listening to Chaotic Whispers: A Deep Learning Framework for News-oriented Stock Trend Prediction. In *WSDM*, WSDM '18, pages 261–269, New York, NY, USA. ACM.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Yang Li, Ting Liu, Jing Jiang, and Liang Zhang. 2016. Hashtag recommendation with topical attention-based lstm. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3019–3029.

Yi-Ting Liou, Chung-Chi Chen, Tsun-Hsien Tang, Hen-Hsen Huang, and Hsin-Hsi Chen. 2021. Finsense: An assistant system for financial journalists and investors. In *Proceedings of the 14th International Conference on Web Search and Data Mining*.

Zhibin Lu, Pan Du, and Jian-Yun Nie. 2020. Vgcn-bert: Augmenting bert with graph embedding for text classification. In *European Conference on Information Retrieval*, pages 369–382. Springer.

Ye Ma, Lu Zong, Yikang Yang, and Jionglong Su. 2019. News2vec: News network embedding with subnode information. 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 4843–4852, Hong Kong, China. Association for Computational Linguistics.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(11).

Eric Malmi, Sebastian Krause, Sascha Rothe, Daniil Mirylenka, and Aliaksei Severyn. 2019. Encode, tag, realize: High-precision text editing. 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 5057–5068.

Bonan Min and Xiaoxi Zhao. 2019. Measure country-level socio-economic indicators with streaming news: An empirical study. 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 1249–1254, Hong Kong, China. Association for Computational Linguistics.

Federico Monti, Michael Bronstein, and Xavier Bresson. 2017. Geometric matrix completion with recurrent multi-graph neural networks. In *Advances in Neural Information Processing Systems*, pages 3697–3707.

Yangtuo Peng and Hui Jiang. 2016. Leverage financial news to predict stock price movements using word embeddings and deep neural networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 374–379, San Diego, California. Association for Computational Linguistics.

Shijie Tang, Yuan Yao, Suwei Zhang, Feng Xu, Tianxiao Gu, Hanghang Tong, Xiaohui Yan, and Jian Lu. 2019. An integral tag recommendation model for textual content. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 5109–5116.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903*.

Lianwei Wu, Yuan Rao, Haolin Jin, Ambreen Nazir, and Ling Sun. 2019. Different absorption from the same sharing: Sifted multi-task learning for fake news detection. 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 4644–4653, Hong Kong, China. Association for Computational Linguistics.

Yumo Xu and Shay B. Cohen. 2018. Stock movement prediction from tweets and historical prices. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1970–1979, Melbourne, Australia. Association for Computational Linguistics.

Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L Hamilton, and Jure Leskovec. 2018. Graph convolutional neural networks for web-scale recommender systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 974–983.

Renchun You, Zhiyao Guo, Lei Cui, Xiang Long, Yingze Bao, and Shilei Wen. 2020. Cross-modality attention with semantic graph embedding for multi-label classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*.