Linking Adaptive Structure Induction and Neuron Filtering: A Spectral Perspective for Aspect-based Sentiment Analysis

Hao Niu∗†**, Maoyi Wang**∗**, Yun Xiong, Biao Yang, Xing Jia, Zhonglei Guo**
Shanghai Key Laboratory of Data Science, School of Computer Science, Fudan University
Shanghai, China
{hniu18, wangmaoyi, yunx, xjia18, guozl18}@fudan.edu.cn
biaoyang22@m.fudan.edu.cn

Abstract

Recently, it has been discovered that incorporating structure information (e.g., dependency trees) can improve the performance of aspect-based sentiment analysis (ABSA). The structure information is often obtained from off-the-shelf parsers, which are sub-optimal and unwieldy. Therefore, adaptively inducing task-specific structures is helpful in resolving this issue. In this work, we concentrate on adaptive graph structure induction for ABSA and explore the impact of neuron-level manipulation from a spectral perspective on structure induction. Specifically, we consider word representations from PLMs (pre-trained language models) as node features and employ a graph learning module to adaptively generate adjacency matrices, followed by graph neural networks (GNNs) to capture both node features and structural information. Meanwhile, we propose the **Neu**ron Fi**lT**ering (NeuLT), a method to conduct neuron-level manipulations on word representations in the frequency domain. We conduct extensive experiments on three public datasets to observe the impact of NeuLT on structure induction and ABSA. The results and further analysis demonstrate that performing neuron-level manipulation through NeuLT can shorten Aspects-sentiment Distance of induced structures and be beneficial to improve the performance of ABSA. The effects of our method can achieve or come close to SOTA (state-of-the-art) performance.

Keywords: Structure Induction, Sentiment Analysis, Neuron Analysis

1. Introduction

Aspect-based sentiment analysis (ABSA) is a fine-grained (token-level) sentiment analysis task for aspects of a given sentence (Vo and Zhang, 2015; Dong et al., 2014). The task aims to detect the sentiment polarities (i.e., POSITIVE, NEGATIVE, NEUTRAL) of given aspects. For instance, in the sentence "The **decor** is not a special at all but their amazing **food** makes up for it" and corresponding aspects "decor" and "food", the sentiment polarity for "decor" is NEGATIVE, while the sentiment for "food" is POSITIVE.

To analyze token-level sentiment in sentences. relevant research often relies on syntactic structures (Zhang et al., 2019b; Tian et al., 2021; Veyseh et al., 2020; Huang and Carley, 2019; Sun et al., 2019; Wang et al., 2020a). In these studies, syntactic structures showed promise in connecting aspects to the corresponding opinion words and assisting in improving the ABSA task's performance. Early works (Vo and Zhang, 2015; Kiritchenko et al., 2014; Schouten and Frasincar, 2016) to deal with ABSA mainly relied on manually designing syntactic features, which is cumbersome. Lately, various neural network-based models (Kiritchenko et al., 2014; Vo and Zhang, 2015; Chen et al., 2017; Zhang et al., 2019b; Wang et al., 2020a; Trusca et al., 2020) have been put forth to deal with the ABSA task, to get rid of hand-crafted feature design. Additionally, some research endeavors (Chen et al., 2020a; Dai et al., 2021; Zhou et al., 2021; Chen et al., 2022) suggest there should exist task-specific induced structures because syntactic structures generated by off-the-shelf dependency parsers are sub-optimal, and not specially designed for ABSA.

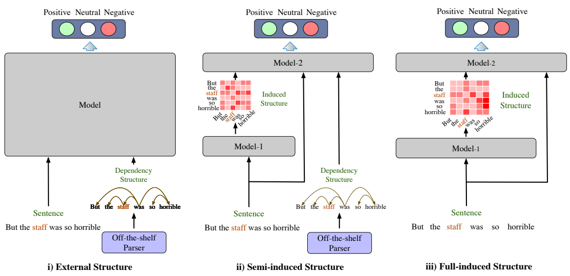

Figure 1: Taxonomy of structure-based ABSA.

By summarizing prior research, we classify the structure-based ABSA works into three categories: **i) external structure**, **ii) semi-induced structure**, and **iii) full-induced structure**. Their patterns are summarized in Figure 1. Studies related to i) external structures utilize syntactic structures generated by off-the-shelf parsers (Zhang et al., 2019b; Sun et al., 2019) or modified syntactic structures (Wang et al., 2020a) to provide structural support for ABSA. Works based on ii) semi-induced struc-

∗Equal Contribution.
†Corresponding Author.

LREC-COLING 2024, pages 10584–10597

tures consider induced structures as a complement to external structures, merging them to offer structural support for ABSA (Chen et al., 2020a). The first two require assistance from external structures, which increases the complexity of preprocessing. The third directly eliminates this burdensomeness, and it still has the potential to perform comparable or superior performance compared to the first two.

Our work follows iii) full-induced structures. Studies in this area plan to eliminate reliance on external structures completely and instead induce task-specific latent structures (Dai et al., 2021; Zhou et al., 2021; Chen et al., 2022). Nevertheless, these studies usually focus on generating tree-based structures, converting them into graph structures, and feeding the graphs' adjacency matrices to Graph Neural Networks (GNNs) to capture structural information. Our research tracks this line of thought but immediately induces graph structures. Moreover, recent studies have suggested that a model's behavior can be controlled by manipulating neurons (Bau et al., 2019; Dai et al., 2022; Suau et al., 2020; Sajjad et al., 2022). For instance, (Bau et al., 2019) were able to modify tense, gender, and other concepts in output translations by manipulating the values of neurons. Neuron-level manipulation in structure induction and ABSA has rarely been investigated in previous studies. Therefore, our research aims to explore the impact of neuron-level manipulation on structure induction for ABSA.

Particularly, we utilize the metric-based graph structure learning (GSL) method (Zhu et al., 2021) to induce latent structures. Moreover, we propose the **Neu**ron Fi**lt**ering (**NeuLT**) as a neuron-level manipulation method to examine its impact on structure induction. Extensive experiments reveal that appropriate heuristic neuron-level manipulation (NeuLT) is beneficial to obtaining suitable graph structures and improving model performance. Meanwhile, further analysis about **Auto**matic **Neu**ron Fi**lt**ering (**NeuLT(Auto)**) demonstrates that NeuLT(Auto) bypasses heuristic manipulations and achieves consistent improvements. Here, we explore three widely used metric functions (Attention-based (*Attn.*), Kernel-based (Kernel), and Cosine-based (Cosine)) and contrast their effects on structure induction for ABSA. Our research is based on three encoder-based PLMs ($\text{BERT}_{\text{base}}$, $\text{RoBERTa}_{\text{base}}$, and $\text{RoBERTa}_{\text{large}}$). We summarize our intriguing findings as follows:

Neuron-level Manipulations. Neuron-level manipulations can influence structure induction. The induced structures of NeuLT obtain lower AsD and better performance compared to the *Attn.* method.

Structure Induction. GSL-based structure induction is effective. The *Attn.* is more suitable for structure induction compared to *Kernel* and Cosine.

Extensive Experiments and Neuron-level Analysis. We conduct extensive experiments and analysis. Results confirm our findings and demonstrate the effectiveness of NeuLT, and neuron-level analysis provides in-depth insights into the approach.

2. Related Work

2.1. Structure Induction in ABSA

Many works in ABSA aim to integrate syntactic structures into neural networks to improve the performance (Zhang et al., 2019b; Sun et al., 2019; Wang et al., 2020a; Niu et al., 2022). Despite advancements in integrating dependency trees, the current state is sub-optimal due to parsing errors in off-the-shelf parsers. Consequently, efforts are being directed towards dynamically learning task-specific tree structures. For example, (Dai et al., 2021) propose to induce tree structure from fine-tuned PLMs. (Chen et al., 2022; Zhou et al., 2021) suggest inducing an aspect-specific latent tree structure by employing policy-based reinforcement learning and aiming to narrow the gap between aspect and opinion. (Chen et al., 2020a) combines dependency trees and automatically induced graph structure by a gate mechanism.

2.2. Controlling Model's Behavior through Neuron-level Manipulation

Recently, some studies have focused on analyzing neurons and controlling the model's performance through neuron-level manipulation (Bau et al., 2019; Dai et al., 2022; Suau et al., 2020). For instance, (Suau et al., 2020) manipulate neurons of concepts in PLMs to generate sentences of specific topics of interest. Additionally, (Dai et al., 2022) updates and erases specific factual knowledge without fine-tuning by manipulating knowledge neurons.

2.3. Spectral Approach in NLP

One line of spectral methods in NLP is used in improving efficiency (Han et al., 2022; Zhang et al., 2018). For instance, (Han et al., 2022) introduces a novel recurrent neural network incorporating the discrete Fourier transformer, resulting in accelerated training. Additionally, some works investigate contextual representation learning from the spectral perspective. (Müller-Eberstein et al., 2022; Tamkin et al., 2020) propose using Frequency filters to constrain neurons to model structures at scales. (Kayal and Tsatsaronis, 2019) proposes a method for creating sentence embeddings that use a spectral decomposition method based on fluid dynamics.

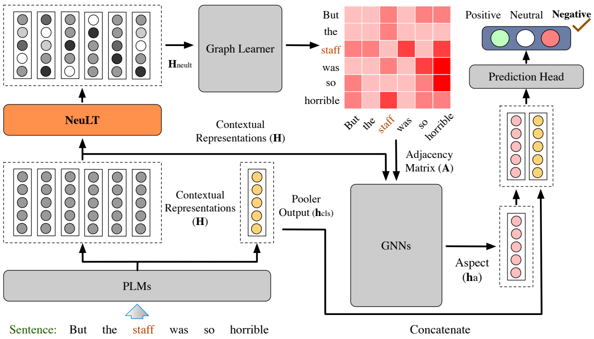

Figure 2: Overall architecture. The aspect (staff) with NEGATIVE sentiment polarity label is in red.

2.4. Metric-based Graph Structure Learning

The metric-based graph structure learning (GSL) determines edge weights by learning a metric function between pairwise representations(Zhu et al., 2021). The method can be categorized into two subgroups according to metric functions: Attention-based and Kernel-based. Attention-based approaches typically employ attention networks or more intricate neural networks to capture the interaction between pairwise representations (Velickovic et al., 2018; Jiang et al., 2019a; Chen et al., 2020b; Zhao et al., 2021a). Kernel-based approaches utilize *Kernel* functions as the metric function to model edge weights (Li et al., 2018; Yu et al., 2020; Zhao et al., 2021b). The Cosine-based method (Chen et al., 2020b) is typically categorized as an Attention-based method.

3. Methodology

We introduce the architecture (in Figure 2) as well as **Neu**ron Fi**lt**ering (NeuLT) to take neuron-level manipulations when adaptively inducing structures.

3.1. Overview

Given an input sentence $S = \{w_1, w_2, \cdots, w_n\}$ and a specific aspect term a, we plan to induce a graph structure g relying on a Graph Learner and utilize the GNNs module as well as Prediction Head to make a judgment about sentiment polarity $y \in \{$POSITIVE, NEURTAL, NEGATIVE$\}$. Firstly, we employ a type of PLMs (BERT$_{base}$ (Devlin et al., 2019), RoBERTa$_{base}$ or RoBERTa$_{large}$ (Liu et al., 2019)) served as the contextual encoder to obtain the hidden contextual representation $\mathbf{H} \in \mathbb{R}^{n\times d}$ of the input sentence S, where d is the dimension of word representations, and n is the length of the given sentence. To facilitate neural-level manipulations, we introduce Module **Neu**ron Fi**lt**ering (NeuLT) to obtain the adjusted contextual representation $\mathbf{H}_{neult}$, which is elaborated in Section 3.3.

Then, we feed $\mathbf{H}_{neult}$ into the Graph Learner module to induce structures g, which serve as adjacency matrices $\mathbf{A}$ for the GNNs module. Simultaneously, the contextual representation $\mathbf{H}$ is waited for inputting into the GNNs module as node representations. Based on $\mathbf{A}$ and $\mathbf{H}$, the GNNs module can extract aspect-specific features $\mathbf{h}_a$ utilizing both structural information from $\mathbf{A}$ and pre-trained knowledge information from $\mathbf{H}$. Finally, we concatenate the representation of [CLS] token $\mathbf{h}_{cls}$ from PLMs as well as $\mathbf{h}_a$, and send them into a Multi-layer Perception (MLP) (served as the Prediction Head) to make predictions.

3.2. Graph Structure Learning (GSL)

We present the configurations of the Graph Learner and the GNNs module. The Graph Learner module is based on Graph Structure Learning (GSL). We investigate the effectiveness of three common GSL methods based on metric learning: Attention-based (*Attn.*), Kernel-based (Kernel), and Cosine-based (Cosine) (refer to (Zhu et al., 2021) for specific descriptions of Kernel-based and Cosine-based methods), and their performance is demonstrated in Section 4.5. Here, our description primarily centers on the Attention-based GSL method. Firstly, we calculate the unnormalized pair-wise edge score ϵ_{ij} for the i-th and j-th words utilizing the given representations $\mathbf{h}_i \in \mathbb{R}^d$ and $\mathbf{h}_j \in \mathbb{R}^d$. Specifically, the pair-wise edge score $\epsilon_{ij} = (\mathbf{W}_i\mathbf{h}_i)(\mathbf{W}_j\mathbf{h}_j)^\top$, where $\mathbf{W}_i, \mathbf{W}_j \in \mathbb{R}^{d\times d_h}$ are learnable weights, and d_h is the hidden dimension.

Then, based on the pair-wise scores ϵ_{ij} for all word pairs, we can construct the adjacency matrices $\mathbf{A}$ for induced graph structures. Concretely,

$$\mathbf{A}_{ij} = \begin{cases} 1 & \text{if} \quad i = j \\ \frac{exp(\epsilon_{ij})}{\sum_{k=1}^{n} exp(\epsilon_{ik})} & \text{otherwise} \end{cases}, \tag{1}$$

where the adaptive adjacency matrix is $\mathbf{A} \in \mathbb{R}^{n\times n}$, and $\mathbf{A}_{ij}$ is the weight score of the edge between the i-th and j-th words.

In addition, we employ commonly used Graph Neural Networks (GCNs) (Kipf and Welling, 2017) as the GNNs module for conciseness without losing generality (other variants of GNNs can also be employed here). Given the word representations $\mathbf{H}$ and the adaptive adjacency matrix $\mathbf{A}$, we can construct an induced graph structure consisting of words (each word acts as a node in the graph). Then, we feed them into GCNs. Specifically,

$$\mathbf{H}^l = \sigma\left(\mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}\mathbf{H}^{l-1}\mathbf{W}^l\right), \tag{2}$$

where σ is an activation function (e.g., ReLU), $\mathbf{W}^l$ is the learnable weight of the l-th GCN layer, and

$\mathbf{D}_{ii} = \sum_j \mathbf{A}_{ij}$ as in (Kipf and Welling, 2017). Therefore, by stacking several layers of Graph Learners and GNNs modules, we can obtain structure information enhanced word representations $\mathbf{H}_g$ for the downstream task. Then, we can get aspect representations $\mathbf{h}_a$ from $\mathbf{H}_g$, and feed $\mathbf{h}_a$ along with the pooler output $\mathbf{h}_{cls}$ of PLMs (the output representation of [CLS] token) into a task-specific Prediction Head to make predictions. It should be noted that the induced graph structure is dynamically updated while training.

3.3. Neuron Filtering (NeuLT)

We explore how neuron-level manipulations affect structure induction and ABSA. Therefore, we introduce NeuLT to achieve this purpose. In addition, as to why we explore manipulating the frequency domain, we found in experiments that manipulations in the frequency domain are easier to optimize than in the time domain. For details, please see the paragraph **Without DFT** in Section 4.9.

Neuron. According to (Sajjad et al., 2022), the term `neuron` refers to the output of a single dimension from any neural network dimension. For instance, in the BERT$_{base}$, a layer block's output comprises 768 neurons, while the output of an attention head has 64 neurons. In this work, we adopt the aforementioned definition of `neuron` to investigate the impact of neuron-level manipulations.

Method Description. The Neuron Filtering (NeuLT) is based on Discrete Fourier Transform (DFT) to conduct disentangling manipulations in the frequency domain. Specifically, given word representations $\mathbf{H} \in \mathbb{R}^{n \times d}$ from PLMs, we send them into the NeuLT before the Graph Learner. Specifically, for the word representations $\mathbf{h}_i \in \mathbb{R}^d$ and $\mathbf{h}_j \in \mathbb{R}^d$, the pair-wise edge score ϵ_{ij} is calculated as follows:

$$\Upsilon^{nlt}(x) = \mathcal{F}^{-1}\Big(\Pi\big(\mathcal{F}(x)\big)\Big), \quad (3)$$

$$\epsilon_{ij} = \Upsilon^{nlt}(\mathbf{W}_i\mathbf{h}_i)\Upsilon^{nlt}(\mathbf{W}_j\mathbf{h}_j)^\top, \quad (4)$$

where $\mathcal{F}(\cdot)$ and $\mathcal{F}^{-1}(\cdot)$ denote the Fast Fourier Transform (FFT) as well as its inverse (IFFT), and Π indicates the filtering operation. Different from (Niu et al., 2023), they conduct filtering at the sentence dimension, but NeuLT's manipulations are all in the neuron dimension. Additionally, Υ^{nlt} denotes the **Neu**ron **Filt**ering (NeuLT).

4. Experiment

To prove the effectiveness of our approach, we demonstrate results conducted on three datasets for ABSA and compare them with baselines.

Table 1: Statistics of datasets.

Dataset	Positive		Neutral		Negative	
	Train	Test	Train	Test	Train	Test
Rest14	2164	728	807	196	637	196
Laptop14	994	341	870	128	464	169
Twitter	1561	173	3127	346	1560	173

4.1. Datasets

We perform experiments on well-established datasets including SemEval 2014 (Rest14 and Laptop14) (Pontiki et al., 2014) and Twitter (Dong et al., 2014). Each dataset comprises three sentiment label categories: POSITIVE, NEUTRAL, and NEGATIVE. Table 1 presents the dataset statistics, where (Train|Test) indicates the number of instances in the training and testing sets for each dataset.

4.2. Implementation Details

We employ popular Encoder-based Pre-trained Language Models (PLMs), namely BERT$_{base}$ (Devlin et al., 2019), RoBERTa$_{base}$, and RoBERTa$_{large}$ (Liu et al., 2019), for word representations. Additionally, all Graph Learners have hidden dimensions of 60, with a batch size of 32. For RoBERTa$base$ and RoBERTa$large$, we train for 60 epochs, and for BERT$base$, we train for 30 epochs. During training, we utilize the Adam optimizer (Kingma and Ba, 2015) with a learning rate of 1e-5. Accuracy and Macro-F1 scores are employed as metrics, consistent with previous studies. All experiments are conducted on an NVIDIA Tesla P100 GPU. [1]

4.3. Baselines

We classify the structure-based ASBA methods into three genres: **i) external structure**, **ii) semi-induced structure**, and **iii) full-induced structure**. Each category is elaborated in the following.

External Structure. These studies utilize syntactic structures generated by external dependency parsers (such as Spacy[2] and Stanford CoreNLP[3]) to provide supplementary structural information for ABSA. Their methodologies typically proceed as follows:

SAGAT (Huang et al., 2020) leverage both graph attention network and BERT to investigate syntax and semantic information for ABSA.

DGEDT (Tang et al., 2020) simultaneously incorporate BERT outputs and dependency syntactic representations using GCNs.

[1] Our code is at https://github.com/hankniu01/NeuLT
[2] https://spacy.io/
[3] https://stanfordnlp.github.io/CoreNLP/

Table 2: The overall performance across the three datasets. The baselines in the 'Structure' column are classified according to the structure categorization (*Dep.*: external structures (dependency syntactic tree), *Semi.*: semi-induced structures, *Full*: full-induced structures, and *None*: no structure information used).

Embedding	Model	Structure	Rest14		Laptop14		Twitter	
			Accuracy	*Macro-F1*	*Accuracy*	*Macro-F1*	*Accuracy*	*Macro-F1*
Static Embedding	depGCN	Dep.	80.77$^\sharp$	72.02$^\sharp$	75.55$^\sharp$	71.05$^\sharp$		
	CDT	Dep.	82.30$^\sharp$	74.02$^\sharp$	77.19$^\sharp$	72.99$^\sharp$		
	kumaGCN	Semi.	81.43	73.64	76.12	72.42	72.45	70.77
	RGAT	Dep.	83.30	76.08	77.42	73.76	75.57	73.82
	FT-RoBERTa(ASGCN)	Full	82.31	73.53	76.33	72.76	73.84	72.66
	FT-RoBERTa(PWCN)	Full	82.40	73.95	76.95	73.21	73.84	71.43
	FT-RoBERTa(RGAT)	Full	82.76	75.25	77.43	74.21	75.43	74.04
BERT$_{base}$	BERT	None	85.62$^\sharp$	78.28$^\sharp$	77.58$^\sharp$	72.38$^\sharp$	75.28	74.11
	SAGAT	Dep.	85.08	77.94	80.37	76.94	75.40	74.17
	DGEDT	Dep.	86.30	80.00	79.80	75.60	77.90	75.40
	depGCN-BERT	Dep.	85.00	78.79	81.19	77.67	75.58	74.58
	RGAT-BERT	Dep.	86.60	81.35	78.21	74.07	76.15	74.88
	KumaGCN-BERT	Semi.	86.43	80.30	81.98	78.81	77.89	77.03
	dotGCN-BERT	Full	86.16	80.49	81.03	78.10	78.11	77.00
RoBERTa$_{base}$	Roberta + MLP	None	87.32	81.01	82.60	79.33	77.17	76.20
	RoBERTa-ASC(Dep)	Dep.	82.82	75.12	74.12	70.52	-	-
	LCFS-ASC-CDW(Dep)	Dep.	86.71	80.31	80.52	77.13	-	-
	Dep(ASGCN)	Dep.	86.90	80.75	81.66	78.31	75.28	74.38
	Dep(PWCN)	Dep.	87.41	81.07	84.16	81.18	76.63	75.60
	Dep(RGAT)	Dep.	87.43	80.61	83.43	80.28	74.42	72.93
	FT-RoBERTa(ASGCN)	Full	86.87	80.59	83.33	80.32	76.10	75.07
	FT-RoBERTa(PWCN)	Full	87.35	80.85	84.01	81.08	77.02	75.52
	FT-RoBERTa(RGAT)	Full	87.52	81.29	83.33	79.95	75.81	74.91
	NeuLT	Full	$\underline{88.93}$	$\underline{83.28}$	$\underline{84.95}$	$\underline{82.26}$	$\underline{78.18}$	$\underline{77.59}$
RoBERTa$_{large}$	**NeuLT**	Full	**89.64**	**84.18**	**86.05**	**84.68**	**78.53**	**77.78**

depGCN (Zhang et al., 2019a) integrates BiLSTM to capture contextual information on word orders along with multi-layered GCNs.

CDT (Sun et al., 2019) leverages both dependency and contextual information through the utilization of GCNs and BiLSTM.

RGAT (Wang et al., 2020a) supplies reshaped syntactic dependency graphs to RGAT to capture aspect-centric information.

LCFS-ASC-CDW (Phan and Ogunbona, 2020) integrate dependency syntactic embeddings, part-of-speech embeddings, and contextualized embeddings to improve ABSA performance.

Semi-induced Structure. Research in this area commonly leverages both dependency syntactic structures from off-the-shelf parsers and induced structures from PLMs, the representative works include:

KumaGCN (Chen et al., 2020a) fuse latent graphs generated by self-attention neural networks with dependency syntactic structures for ABSA.

Full-induced Structure. This research aims to eliminate the need for external parsers entirely by inducing task-specific latent structures for downstream tasks. Its delegate does the following:

dotGCN (Chen et al., 2022) utilize reinforcement learning and attention-based regularization to induce aspect-specific opinion tree structures.

FT-RoBERTa (Dai et al., 2021) employ a dependency probing approach to induce tree structures from a RoBERTa model, which has been pre-trained on ABSA datasets.

4.4. Main Results

The main results of baselines and NeuLT on the three datasets are shown in Table 2. Baselines are categorized based on their embedding type (static embedding (GloVe), BERT$_{base}$, RoBERTa$_{base}$, and RoBERTa$_{large}$) and the structure they utilize (None, Dep., Semi., and Full). The parameters of PLMs are tuned in conjunction with the parameters of the entire model. Compared with all of the baselines, NeuLT obtains the best results. In comparison with FT-RoBERTa-series works (Dai et al., 2021), the most relevant work, NeuLT outperforms them a lot on all three datasets. It's noteworthy that while FT-RoBERTa-series approaches necessitate pre-training of PLMs on ABSA datasets, NeuLT does not. As a result, NeuLT is less complicated and more effective than the FT-RoBERTa-series works.

4.5. Metric Function

From the insight of Graph Structure Learning (Chen et al., 2020b; Zhu et al., 2021), the common options for metric learning include attention mechanism (Vaswani et al., 2017; Jiang et al., 2019a), radial basis function *Kernel* (Li et al., 2018; Yeung and Chang, 2007), and *Cosine* similarity (Wojke and Bewley, 2018). Therefore, in this section, we compare the impact of three representative metric functions on structure induction: Attention-based (*Attn.*),

Table 3: Results of Ablation Studies.

Embedding	Model	Structure	Rest14		Laptop14		Twitter	
			Accuracy	*Macro-F1*	*Accuracy*	*Macro-F1*	*Accuracy*	*Macro-F1*
BERT$_{base}$	*Attn.*	Full	85.43	78.04	80.54	77.06	76.22	75.04
	NeuLT	Full	**86.95**	**81.20**	**81.33**	**77.20**	**77.10**	**75.83**
RoBERTa$_{base}$	*Attn.*	Full	87.59	81.72	83.86	80.53	75.72	73.92
	NeuLT	Full	**88.93**	**83.28**	**84.95**	**82.26**	**78.18**	**77.59**
RoBERTa$_{large}$	*Attn.*	Full	89.46	84.12	84.80	82.19	77.02	75.75
	NeuLT	Full	**89.64**	**84.18**	**86.05**	**84.68**	**78.53**	**77.78**

Table 4: The influence of various metric functions based on RoBERTa$_{base}$.

Metric	Rest14		Laptop14		Twitter	
	Accuracy	*Macro-F1*	*Accuracy*	*Macro-F1*	*Accuracy*	*Macro-F1*
Attn.	**87.59**	**81.72**	**83.86**	**80.53**	<u>75.72</u>	<u>73.92</u>
Kernel	<u>87.14</u>	<u>80.45</u>	<u>83.54</u>	<u>80.44</u>	**76.01**	**73.98**
Cosine	<u>87.14</u>	79.94	83.39	79.93	74.28	72.80

Kernel-based (Kernel), and Cosine-based (Cosine). Following the footsteps of previous works, We implement the counterpart metric functions (*Kernel* and Cosine) for comparison, the results are shown in Table 4. Except for Twitter, the performance of *Attn.* yields the best results. However, the margin between *Attn.* and *Kernel* on Twitter is small (0.29% for Accuracy and 0.06% for Macro-F1), so we chose *Attn.* as the default setting for our work.

4.6. Ablation Study

We perform ablation studies to showcase the effectiveness of NeuLT, built upon the *Attn.* method. We contrast *Attn.* with NeuLT across three PLMs. Results are illustrated in Table 3. Compared to *Attn.*, NeuLT notably enhances consistency across three datasets across different PLMs. Thus, the neuron-level manipulation facilitated by NeuLT demonstrates its effectiveness.

4.7. Aspects-sentiment Distance (AsD)

Method Description. To exhibit the effectiveness of induced structures, in line with (Dai et al., 2021), we introduce the Aspects-sentiment Distance (AsD) metric to measure the average distance between aspect and sentiment words across various structures. AsD is computed as follows:

$$AsD(S_i) = \frac{\sum_{A}^{a_p} \sum_{C^\star}^{c_q} dist(a_p, c_q)}{|A||C^\star|}, \tag{5}$$

$$AsD(D) = \frac{\sum_{D} AsD(S_i)}{|D|}, \tag{6}$$

where $C = \langle c_1, \cdots c_q \rangle$ is a sentiment words set (following the setting from (Dai et al., 2021)), S_i denotes each sentence in dataset D, and $C^\star = S_i \cap C$. Additionally, $A = \langle a_1, \cdots, a_p \rangle$ denotes the set of aspects for each sentence. We utilize $dist(n_1, n_2)$ to calculate the relative distance between two nodes (n_1 and n_2) on the graph structure, and $|\cdot|$ denotes the cardinality of the given set.

Table 5: The Aspects-sentiment Distance (AsD) across different structures in all datasets. The dependency tree structure (Dep.) is derived from the Spacy parser.

Structure	Rest14	Laptop14	Twitter
Dep.	8.19	8.02	8.33
Attn.	2.26	2.55	2.64
NeuLT	**2.04**	**2.39**	**2.48**

Results. As shown in Table 5, the less magnitude indicates the shorter distance between aspects and sentiment words. Compared to dependency structure (Dep.), *Attn.* and NeuLT shorten the AsD greatly, which shows that GSL methods encourage the aspects to find sentiment words. Furthermore, when compared to *Attn.*, NeuLT has a lower AsD score, demonstrating that a reasonable adjustment at the neuron level can result in better structures.

4.8. Automatic Neuron Filtering (NeuLT(Auto))

To further indicate the effectiveness of NeuLT, get rid of the cumbersome heuristic frequency selection, and get consistent improvement, we introduce an **A**utomatic **N**euron **S**election (**ANS**) module to adaptively perform neuron-level manipulations along with the optimization of the overall model. We denote it as textbfAutomatic **Neu**ron Fi**lt**ering (**NeuLT(Auto)**), as a variant of NeuLT.

Method Description. To achieve this goal, we design the ANS module Υ under a probabilistic scenario to replace the filtering operation Π. Specifically, we map the FFT-processed contextual word representations $\mathbf{H}_{fft} \in \mathbb{R}^{n\times d}$ ($\mathbf{H}_{fft} = \mathcal{F}(\mathbf{H})$) into a Bernoulli parameter space by employing a Multi-layer Perceptron (MLP) architecture to parameterize this mapping process. We utilize the MLP architecture (composed of two linear projection layers $Linear_1$ and $Linear_2$, and an activation function σ (i.e., ReLU)) to map each neuron of $\mathbf{H}_{fft}$ into the

Table 6: Results of Automatic Neuron Filtering (NeuLT(Auto)).

Embedding	Model	Structure	Rest14		Laptop14		Twitter	
			Accuracy	*Macro-F1*	*Accuracy*	*Macro-F1*	*Accuracy*	*Macro-F1*
BERT$_{base}$	*Attn.*	Full	85.43	78.04	80.54	77.06	76.22	75.04
	NeuLT(Auto)	Full	**87.04**	**81.19**	**81.96**	**78.70**	**76.96**	**76.01**
RoBERTa$_{base}$	*Attn.*	Full	87.59	81.72	83.86	80.53	75.72	73.92
	NeuLT(Auto)	Full	**88.48**	**83.20**	**84.95**	**82.15**	**78.10**	**77.76**
RoBERTa$_{large}$	*Attn.*	Full	89.46	84.12	84.80	82.19	77.02	75.75
	NeuLT(Auto)	Full	**89.55**	**84.87**	**85.89**	82.19	**78.87**	**78.02**

Bernoulli parameter space to get $\mathbf{H}_{bern} \in \mathbb{R}^{n\times d\times 2}$,

$$\begin{aligned}\mathbf{z}_{bern} &= MLP(\mathbf{H}_{fft}) \\ &= Linear_2\Big(\sigma\big(Linear_1(\mathbf{H}_{fft})\big)\Big), \end{aligned} \tag{7}$$

$$\mathbf{H}_{bern} = \varphi\bigg(\Big(\mathbf{z}_{bern} - log\big(-log(\epsilon)\big)\Big)/\tau\bigg), \tag{8}$$

where the last dimension of $\mathbf{H}_{bern}$ denotes the success probability of Bernoulli distribution for each neuron, and φ denotes the Softmax function.

Here, we utilize the Gumbel reparameterization proposed by (Jang et al., 2017; Maddison et al., 2017) to address the differentiable difficulty during training, where ϵ is a tensor with the same dimension as $\mathbf{z}_{bern} \in \mathbb{R}^{n\times d\times 2}$, where the values of ϵ is random variables of a uniform distribution on the interval $(0, 1)$. The hyperparameter $\tau \to 0$ is the annealing temperature, which is adjusted to zero progressively in practice. Next, we can obtain a mask matrix $\mathbf{M}_{bern} \sim Bern(\mathbf{H}_{bern})$ composed of a set of Bernoulli random variables with the same dimension as $\mathbf{H}_{fft}$, where $\mathbf{M}_{bern} \in \mathbb{R}^{n\times d}$ and the values of $\mathbf{M}_{bern}$ are $\in \{0, 1\}^{n\times d}$. Each value in $\mathbf{M}_{bern}$ indicates whether to manipulate the corresponding neuron. During the non-training phase, we can set a hyperparameter threshold γ to control the sparsity of $\mathbf{M}_{bern}$. Therefore, the ANS module Π^{ans} is to obtain a mask matrix $\mathbf{M}_{bern}$ to indicate which neurons need to be manipulated. Thus,

$$\Pi^{ans}(\mathcal{F}(\mathbf{H})) = \mathbf{M}_{bern} \odot \mathbf{H}_{fft}, \tag{9}$$

where $\odot$ denotes the Hadamard product. For the i-th and j-th word representations $\mathbf{h}_i \in \mathbb{R}^d$ and $\mathbf{h}_j \in \mathbb{R}^d$, we can calculate the pair-wise edge score ϵ_{ij} as follows:

$$\Upsilon^{afs}(x) = \mathcal{F}^{-1}\Big(\Pi^{ans}\big(\mathcal{F}(x)\big)\Big), \tag{10}$$

$$\epsilon_{ij} = \Upsilon^{auto}(\mathbf{W}_i\mathbf{h}_i)\Upsilon^{auto}(\mathbf{W}_j\mathbf{h}_j)^\top, \tag{11}$$

where Υ^{auto} denotes the NeuLT(Auto).

Results. We utilize NeuLT(Auto) instead of NeuLT to conduct experiments, which are shown in Table 6. In this context, the ANS module is also optimized using the Adam optimizer and has an independent learning rate set to 5e-3. Compared to

Table 7: Different frequency filters on RoBERTa$_{base}$. **Bold** indicate improved performance.

Model	Pattern		Rest14(emb)		Laptop14(emb)		Twitter(emb)	
			Acc	F1	Acc	F1	Acc	F1
NeuLT	High	256	87.68	81.28	83.70	80.66	**76.88**	**76.04**
		154	87.50	82.49	83.39	80.22	75.29	74.47
		77	87.59	80.50	83.70	80.99	74.57	72.90
		16	**88.04**	**82.93**	**84.01**	**81.19**	75.29	74.46
		8	**88.13**	**82.09**	83.23	80.43	74.57	74.13
		4	87.32	81.42	**84.33**	**81.69**	74.13	72.50
		2	87.68	81.00	83.86	80.82	74.71	73.13
	Bond	256	87.23	80.83	83.54	80.30	**76.44**	**75.44**
		154	**87.95**	**81.90**	**84.33**	**81.29**	75.58	74.62
		77	87.41	80.64	**84.48**	**81.44**	75.29	74.26
		16	**87.86**	**81.66**	83.70	80.51	75.43	74.45
		8	87.32	81.52	**84.48**	**81.59**	75.14	74.22
		4	**88.04**	**82.19**	83.86	80.89	74.71	73.66
		2	**87.77**	**81.82**	82.92	79.63	75.14	74.00
	Low	256	87.05	80.61	**84.01**	**81.07**	**75.87**	**75.17**
		154	**87.95**	**82.32**	83.54	80.69	75.00	74.07
		77	**88.75**	**82.99**	83.07	79.87	74.28	73.15
		16	**88.84**	**83.33**	83.07	79.22	75.58	74.52
		8	87.32	80.71	**84.33**	**81.71**	74.57	73.46
		4	87.14	80.16	**84.48**	**81.30**	74.86	73.61
		2	87.41	80.96	83.54	80.41	74.71	73.99
Attn.	-		87.77	81.33	84.01	80.94	75.43	74.81
NeuLT(Auto)	-		**87.95**	**81.99**	**84.95**	**82.26**	**76.30**	**75.67**
NeuLT(Auto) w/o DFT	-		87.50	80.54	83.70	80.76	75.87	74.37

Table 8: Neuron statistics in the frequency domain (arranged from high to low frequency).

Pattern	Range	Ratio(%)					
		Rest14		Laptop14		Twitter	
		Train	Test	Train	Test	Train	Test
High	$512 \to 768$	69.42	69.37	44.36	44.58	69.28	69.19
Bond	$256 \to 512$	67.93	67.90	42.31	42.15	67.57	67.61
Low	$1 \to 256$	68.61	69.05	41.93	42.00	67.21	67.18

Attn., NeuLT(Auto) is consistently improved. This further illustrates that neuron-level manipulation is conducive to improving the effectiveness of ABSA. Furthermore, compared with NeuLT, NeuLT(Auto) avoids the burden brought by manual frequency selection, making it more flexible.

4.9. Neuron-level Analysis

Method Description. In this section, we provide an in-depth analysis of the neuron level to investigate how neurons change after NeuLT processing. Firstly, in order to explore the neuron adjustment at the word embedding level, we remove the weight matrices ($\mathbf{W}_i$ and $\mathbf{W}_j$) in Formulae 4 and 11. Hence, in this section, the Formulae are written as follows:

$$\epsilon_{ij} = \Upsilon^{nlt/auto}(\mathbf{h}_i)\Upsilon^{nlt/auto}(\mathbf{h}_j)^\top. \tag{12}$$

Under this setting, we compare the performance with different filtering operations, including High-, Bond-, and Low-based filters, as well as ANS.

Table 9: Statistics in the **frequency** domain. **Bold** indicates distinct neurons in top-N.

Dataset		Top-N Neurons (N=10)
Rest14	Train	61, 344, 305, 227, 211, 88, 71, **256**, **168**, **310**
	Test	61, 305, 344, 227, 88, **229**, **32**, **173**, 211, 71
Laptop14	Train	0, 264, 134, 299, 95, 123, 367, **209**, **384**, 281
	Test	0, 264, 134, 367, 299, **192**, 281, 384, 95, 123
Twitter	Train	0, 113, 256, 134, 49, 332, 339, 264, 3, **111**
	Test	0, 113, 256, 264, 49, 332, 3, 134, 339, **115**

Table 10: Statistics in the **time** domain. Underline indicates distinct neurons, and **bold** indicates the same neurons.

Dataset		Top-N Neurons (N=10)
Rest14	Train	**0**, 181, **1**, 594, 304, 269, <u>*675*</u>, 241, 499, 194
	Test	**0**, 181, **1**, 194, 304, 499, 594, 269, <u>*136*</u>, 241
Laptop14	Train	**0**, 32, 689, 642, 383, 747, <u>*280*</u>, 724, <u>*230*</u>, 464
	Test	32, 689, **0**, 383, <u>*626*</u>, 642, <u>*485*</u>, 724, 747, 464
Twitter	Train	**1**, 767, 3, <u>*510*</u>, 691, 645, 116, <u>*344*</u>, <u>*727*</u>, <u>*146*</u>
	Test	**1**, 767, 3, 691, 116, 645, <u>*96*</u>, <u>*448*</u>, <u>*112*</u>, <u>*533*</u>

Results. The results in Table 7, present that NeuLT can effectively enhance the performance, but it is sensitive to filtering parameters, as inappropriate parameter selection can lead to degraded performance. Moreover, different datasets require different filtering parameters to achieve optimal performance enhancement. In the same setup, we also conduct NeuLT(Auto) experiments under the Formula 12. NeuLT(Auto) achieves consistent improvement compared to *Attn.* It is evident that removing the weight matrices, and directly adjusting the neurons in word embeddings remains effective.

Without DFT. We eliminated the Discrete Fourier Transform (DFT) and performed neuron adjustments directly in the time domain (word embedding level). We denote this setting as NeuLT(Auto) w/o DFT. Actually, we adjusted the parameters of the ANS module with different learning rates of $\{1e-3, 5e-3, 8e-3\}$, but we all got the same results in Table 7. We find that, when omitting the DFT, neuron-level manipulations directly in the time domain by optimizing the ANS module are more challenging than in the frequency domain.

Neuron-level Statistics. We conducted an analysis of the neuron adjustments in both the frequency and time domains while utilizing NeuLT(Auto).

Frequency Domain. We computed the adjustment ratios of NeuLT(Auto) for each frequency band (High, Bond, and Low) across all cases on three datasets, which are in Table 8. The adjustment ratios for each band are generally consistent, indicating that NeuLT(Auto) does not exhibit a specific bias towards any particular band. Additionally, the adjustment ratios of NeuLT(Auto) are adjusted correspondingly when applied to different datasets.

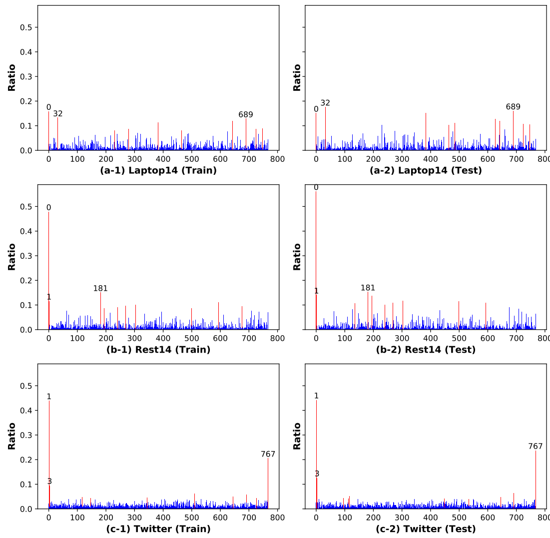

Figure 3: Statistics of Neuron Adjustment. The red line highlights the Top-10 neurons with the highest ratio of adjustment occurrences across all neurons.

Moreover, we similarly conduct a statistical analysis of the top-N neurons selected by NeuLT(Auto), which are presented in Table 9. NeuLT(Auto) selects different top-N neurons for different datasets. While for the training and testing sets within the same dataset, the top-N neurons are generally consistent with slight variations. This suggests that NeuLT(Auto) automatically adapts to different domains of datasets and even within the same domain, it makes fine-tuned adjustments.

Time Domain. We make statistics on the Top-N neurons that underwent the most adjustments in the time domain owing to the manipulations in the frequency domain (NeuLT(Auto)). It is shown in Table 10. Similarly, it is evident that NeuLT(Auto) selects different neurons for datasets from distinct domains, but within the same domain (e.g., Rest14 (Train) and Rest14 (Test)), the neuron selection remains mostly consistent with minor adjustments.

Meanwhile, we visualized the magnitude of adjustments for these neurons and marked the Top-3 neurons with the most significant adjustments, as shown in Figure 3. It is noteworthy that, across all datasets, while the Top-N neurons in the frequency domain are generally different, the 0-th and 1-st position neurons in the time domain consistently rank within the Top-3. It's evident that the adjustments to the 0-th and 1-st position neurons are particularly prominent, especially in Rest14 and Twitter. A rough inference can be made that the adjustments to these neurons at the 0-th and 1-st positions are likely to enhance GSL and performance.

5. Conclusion

In this work, we propose utilizing GSL to induce latent structures for ABSA by performing a neuron-level manipulation (NeuLT and NeuLT(Auto)) in the frequency domain. Extensive experiments and analyses demonstrate that such neuron-level manipulation is effective in structure induction and improvement of ABSA. Furthermore, we conducted an in-depth neural-level analysis to explore this phenomenon. Our exploration is also beneficial to provide inspiration for other similar domains.

Limitations

We have validated the effectiveness of neuron-level manipulations for structure induction as well as ABSA and conducted an in-depth analysis of neuron variations. However, it remains challenging to establish a direct correspondence between specific neurons and their functions. This aspect requires further exploration.

Ethical Considerations

The data we utilized is publicly available, and there are no copyright concerns. The utilization and outputs of our model also do not pose any harm to society.

Acknowledgements

This work is funded in part by the National Natural Science Foundation of China Projects No. U1936213, the Shanghai Science and Technology Development Fund No.22dz1200704.

6. Bibliographical References

Alejandro Fuster Baggetto and Víctor Fresno. 2022. Is anisotropy really the cause of BERT embeddings not being semantic? In *Findings of the Association for Computational Linguistics: EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 4271–4281. Association for Computational Linguistics.

Anthony Bau, Yonatan Belinkov, Hassan Sajjad, Nadir Durrani, Fahim Dalvi, and James R. Glass. 2019. Identifying and controlling important neurons in neural machine translation. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Daniel Bis, Maksim Podkorytov, and Xiuwen Liu. 2021. Too much in common: Shifting of embeddings in transformer language models and its implications. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 5117–5130. Association for Computational Linguistics.

Gianni Brauwers and Flavius Frasincar. 2023. A survey on aspect-based sentiment classification. *ACM Comput. Surv.*, 55(4):65:1–65:37.

BSI. 1973a. *Natural Fibre Twines*, 3rd edition. British Standards Institution, London. BS 2570.

BSI. 1973b. Natural fibre twines. BS 2570, British Standards Institution, London. 3rd. edn.

Razvan C. Bunescu and Raymond J. Mooney. 2005. A shortest path dependency kernel for relation extraction. In *HLT/EMNLP*, pages 724–731. The Association for Computational Linguistics.

A. Castor and L. E. Pollux. 1992. The use of user modelling to guide inference and learning. *Applied Intelligence*, 2(1):37–53.

Chenhua Chen, Zhiyang Teng, Zhongqing Wang, and Yue Zhang. 2022. Discrete opinion tree induction for aspect-based sentiment analysis. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 2051–2064. Association for Computational Linguistics.

Chenhua Chen, Zhiyang Teng, and Yue Zhang. 2020a. Inducing target-specific latent structures for aspect sentiment classification. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 5596–5607. Association for Computational Linguistics.

Peng Chen, Zhongqian Sun, Lidong Bing, and Wei Yang. 2017. Recurrent attention network on memory for aspect sentiment analysis. In *EMNLP*, pages 452–461. Association for Computational Linguistics.

Yu Chen, Lingfei Wu, and Mohammed J. Zaki. 2020b. Iterative deep graph learning for graph neural networks: Better and robust node embeddings. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

J.L. Chercheur. 1994. *Case-Based Reasoning*, 2nd edition. Morgan Kaufman Publishers, San Mateo, CA.

N. Chomsky. 1973. Conditions on transformations. In *A festschrift for Morris Halle*, New York. Holt, Rinehart & Winston.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2022. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 8493–8502. Association for Computational Linguistics.

Junqi Dai, Hang Yan, Tianxiang Sun, Pengfei Liu, and Xipeng Qiu. 2021. Does syntax matter? A strong baseline for aspect-based sentiment analysis with roberta. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 1816–1829. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT (1)*, pages 4171–4186. Association for Computational Linguistics.

Yue Ding, Karolis Martinkus, Damian Pascual, Simon Clematide, and Roger Wattenhofer. 2021. On isotropy calibration of transformers. *CoRR*, abs/2109.13304.

Li Dong, Furu Wei, Chuanqi Tan, Duyu Tang, Ming Zhou, and Ke Xu. 2014. Adaptive recursive neural network for target-dependent twitter sentiment classification. In *ACL (2)*, pages 49–54. The Association for Computer Linguistics.

Timothy Dozat and Christopher D. Manning. 2017. Deep biaffine attention for neural dependency parsing. In *ICLR (Poster)*. OpenReview.net.

Umberto Eco. 1990. *The Limits of Interpretation*. Indian University Press.

Kawin Ethayarajh. 2019. How contextual are contextualized word representations? comparing the geometry of bert, elmo, and GPT-2 embeddings. 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 2019, Hong Kong, China, November 3-7, 2019*, pages 55–65. Association for Computational Linguistics.

Jun Gao, Di He, Xu Tan, Tao Qin, Liwei Wang, and Tie-Yan Liu. 2019. Representation degeneration problem in training natural language generation models. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Bing Han, Cheng Wang, and Kaushik Roy. 2022. Oscillatory fourier neural network: A compact and efficient architecture for sequential processing. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022*, pages 6838–6846. AAAI Press.

Devamanyu Hazarika, Soujanya Poria, Prateek Vij, Gangeshwar Krishnamurthy, Erik Cambria, and Roger Zimmermann. 2018. Modeling inter-aspect dependencies for aspect-based sentiment analysis. In *NAACL-HLT (2)*, pages 266–270. Association for Computational Linguistics.

Paul Gerhard Hoel. 1971a. *Elementary Statistics*, 3rd edition. Wiley series in probability and mathematical statistics. Wiley, New York, Chichester. ISBN 0 471 40300.

Paul Gerhard Hoel. 1971b. *Elementary Statistics*, 3rd edition, Wiley series in probability and mathematical statistics, pages 19–33. Wiley, New York, Chichester. ISBN 0 471 40300.

Binxuan Huang and Kathleen M. Carley. 2019. Syntax-aware aspect level sentiment classification with graph attention networks. In *EMNLP/IJCNLP (1)*, pages 5468–5476. Association for Computational Linguistics.

Lianzhe Huang, Xin Sun, Sujian Li, Linhao Zhang, and Houfeng Wang. 2020. Syntax-aware graph attention network for aspect-level sentiment classification. In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pages 799–810. International Committee on Computational Linguistics.

Eric Jang, Shixiang Gu, and Ben Poole. 2017. Categorical reparameterization with gumbel-softmax. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Otto Jespersen. 1922. *Language: Its Nature, Development, and Origin*. Allen and Unwin.

Bo Jiang, Ziyan Zhang, Doudou Lin, Jin Tang, and Bin Luo. 2019a. Semi-supervised learning with graph learning-convolutional networks. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 11313–11320. Computer Vision Foundation / IEEE.

Qingnan Jiang, Lei Chen, Ruifeng Xu, Xiang Ao, and Min Yang. 2019b. A challenge dataset and effective models for aspect-based sentiment analysis. In *EMNLP/IJCNLP (1)*, pages 6279–6284. Association for Computational Linguistics.

Subhradeep Kayal and George Tsatsaronis. 2019. Eigensent: Spectral sentence embeddings using higher-order dynamic mode decomposition. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 4536–4546. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *ICLR (Poster)*.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Svetlana Kiritchenko, Xiaodan Zhu, Colin Cherry, and Saif M. Mohammad. 2014. Nrc-canada-2014: Detecting aspects and sentiment in customer reviews. In *Proceedings of the 8th International Workshop on Semantic Evaluation, SemEval@COLING 2014, Dublin, Ireland, August 23-24, 2014*, pages 437–442. The Association for Computer Linguistics.

Olga Kovaleva, Saurabh Kulshreshtha, Anna Rogers, and Anna Rumshisky. 2021. BERT busters: Outlier dimensions that disrupt transformers. In *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 3392–3405. Association for Computational Linguistics.

Bohan Li, Hao Zhou, Junxian He, Mingxuan Wang, Yiming Yang, and Lei Li. 2020a. On the sentence embeddings from pre-trained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 9119–9130. Association for Computational Linguistics.

Ruoyu Li, Sheng Wang, Feiyun Zhu, and Junzhou Huang. 2018. Adaptive graph convolutional neural networks. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 3546–3553. AAAI Press.

Yuncong Li, Cunxiang Yin, Sheng-hua Zhong, and Xu Pan. 2020b. Multi-instance multi-label learning networks for aspect-category sentiment analysis. In *EMNLP (1)*, pages 3550–3560. Association for Computational Linguistics.

Bin Liang, Rongdi Yin, Lin Gui, Jiachen Du, and Ruifeng Xu. 2020. Jointly learning aspect-focused and inter-aspect relations with graph convolutional networks for aspect sentiment analysis. In *COLING*, pages 150–161. International Committee on Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Ziyang Luo, Artur Kulmizev, and Xiaoxi Mao. 2021. Positional artefacts propagate through masked language model embeddings. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 5312–5327. Association for Computational Linguistics.

Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2017. Interactive attention networks for aspect-level sentiment classification. In *IJCAI*, pages 4068–4074. ijcai.org.

Xiao Ma, Jiangfeng Zeng, Limei Peng, Giancarlo Fortino, and Yin Zhang. 2019. Modeling multi-aspects within one opinionated sentence simultaneously for aspect-level sentiment analysis. *Future Gener. Comput. Syst.*, 93:304–311.

Chris J. Maddison, Andriy Mnih, and Yee Whye Teh. 2017. The concrete distribution: A continuous relaxation of discrete random variables. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Navonil Majumder, Soujanya Poria, Alexander F. Gelbukh, Md. Shad Akhtar, Erik Cambria, and Asif Ekbal. 2018. IARM: inter-aspect relation modeling with memory networks in aspect-based sentiment analysis. In *EMNLP*, pages 3402–3411. Association for Computational Linguistics.

Max Müller-Eberstein, Rob van der Goot, and Barbara Plank. 2022. Spectral probing. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 7730–7741. Association for Computational Linguistics.

Maximilian Nickel, Lorenzo Rosasco, and Tomaso A. Poggio. 2016. Holographic embeddings of knowledge graphs. In *AAAI*, pages 1955–1961. AAAI Press.

Hao Niu, Yun Xiong, Jian Gao, Zhongchen Miao, Xiaosu Wang, Hongrun Ren, Yao Zhang, and Yangyong Zhu. 2022. Composition-based heterogeneous graph multi-channel attention network for multi-aspect multi-sentiment classification. In *Proceedings of the 29th International Conference on Computational Linguistics, COLING 2022, Gyeongju, Republic of Korea, October 12-17, 2022*, pages 6827–6836. International Committee on Computational Linguistics.

Hao Niu, Yun Xiong, Xiaosu Wang, Wenjing Yu, Yao Zhang, and Zhonglei Guo. 2023. Adaptive structure induction for aspect-based sentiment analysis with spectral perspective. In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, pages 1113–1126. Association for Computational Linguistics.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 2227–2237. Association for Computational Linguistics.

Minh-Hieu Phan and Philip O. Ogunbona. 2020. Modelling context and syntactical features for aspect-based sentiment analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 3211–3220. Association for Computational Linguistics.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. In *SemEval@COLING*, pages 27–35. The Association for Computer Linguistics.

Hassan Sajjad, Nadir Durrani, and Fahim Dalvi. 2022. Neuron-level interpretation of deep NLP models: A survey. *Trans. Assoc. Comput. Linguistics*, 10:1285–1303.

Kim Schouten and Flavius Frasincar. 2016. Survey on aspect-level sentiment analysis. *IEEE Trans. Knowl. Data Eng.*, 28(3):813–830.

Charles Joseph Singer, E. J. Holmyard, and A. R. Hall, editors. 1954–58. *A history of technology*. Oxford University Press, London. 5 vol.

Jannik Strötgen and Michael Gertz. 2012. Temporal tagging on different domains: Challenges, strategies, and gold standards. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, pages 3746–3753, Istanbul, Turkey. European Language Resource Association (ELRA).

Xavier Suau, Luca Zappella, and Nicholas Apostoloff. 2020. Finding experts in transformer models. *CoRR*, abs/2005.07647.

Kai Sun, Richong Zhang, Samuel Mensah, Yongyi Mao, and Xudong Liu. 2019. Aspect-level sentiment analysis via convolution over dependency tree. In *EMNLP/IJCNLP (1)*, pages 5678–5687. Association for Computational Linguistics.

S. Superman, B. Batman, C. Catwoman, and S. Spiderman. 2000. *Superheroes experiences with books*, 20th edition. The Phantom Editors Associates, Gotham City.

Alex Tamkin, Dan Jurafsky, and Noah D. Goodman. 2020. Language through a prism: A spectral approach for multiscale language representations. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu. 2016. Effective lstms for target-dependent sentiment classification. In *COLING*, pages 3298–3307. ACL.

Hao Tang, Donghong Ji, Chenliang Li, and Qiji Zhou. 2020. Dependency graph enhanced dual-transformer structure for aspect-based sentiment classification. In *ACL*, pages 6578–6588. Association for Computational Linguistics.

Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R. Thomas McCoy, Najoung Kim, Benjamin Van Durme, Samuel R. Bowman, Dipanjan Das, and Ellie Pavlick. 2019. What do you learn from context? probing for sentence structure in contextualized word representations.

In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Yuanhe Tian, Guimin Chen, and Yan Song. 2021. Aspect-based sentiment analysis with type-aware graph convolutional networks and layer ensemble. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 2910–2922. Association for Computational Linguistics.

Maria Mihaela Trusca, Daan Wassenberg, Flavius Frasincar, and Rommert Dekker. 2020. A hybrid approach for aspect-based sentiment analysis using deep contextual word embeddings and hierarchical attention. In *Web Engineering - 20th International Conference, ICWE 2020, Helsinki, Finland, June 9-12, 2020, Proceedings*, volume 12128 of *Lecture Notes in Computer Science*, pages 365–380. Springer.

Shikhar Vashishth, Soumya Sanyal, Vikram Nitin, and Partha P. Talukdar. 2020. Composition-based multi-relational graph convolutional networks. In *ICLR*. OpenReview.net.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*, pages 5998–6008.

Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph attention networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Amir Pouran Ben Veyseh, Nasim Nouri, Franck Dernoncourt, Quan Hung Tran, Dejing Dou, and Thien Huu Nguyen. 2020. Improving aspect-based sentiment analysis with gated graph convolutional networks and syntax-based regulation. In *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 4543–4548. Association for Computational Linguistics.

Duy-Tin Vo and Yue Zhang. 2015. Target-dependent twitter sentiment classification with rich automatic features. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, pages 1347–1353. AAAI Press.

Kai Wang, Weizhou Shen, Yunyi Yang, Xiaojun Quan, and Rui Wang. 2020a. Relational graph attention network for aspect-based sentiment analysis. In *ACL*, pages 3229–3238. Association for Computational Linguistics.

Lingxiao Wang, Jing Huang, Kevin Huang, Ziniu Hu, Guangtao Wang, and Quanquan Gu. 2020b. Improving neural language generation with spectrum control. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. 2016. Attention-based LSTM for aspect-level sentiment classification. In *EMNLP*, pages 606–615. The Association for Computational Linguistics.

Nicolai Wojke and Alex Bewley. 2018. Deep cosine metric learning for person re-identification. In *2018 IEEE Winter Conference on Applications of Computer Vision, WACV 2018, Lake Tahoe, NV, USA, March 12-15, 2018*, pages 748–756. IEEE Computer Society.

Wei Xue and Tao Li. 2018. Aspect based sentiment analysis with gated convolutional networks. In *ACL (1)*, pages 2514–2523. Association for Computational Linguistics.

Dit-Yan Yeung and Hong Chang. 2007. A kernel approach for semisupervised metric learning. *IEEE Trans. Neural Networks*, 18(1):141–149.

Donghan Yu, Ruohong Zhang, Zhengbao Jiang, Yuexin Wu, and Yiming Yang. 2020. Graph-revised convolutional network. In *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2020, Ghent, Belgium, September 14-18, 2020, Proceedings, Part III*, volume 12459 of *Lecture Notes in Computer Science*, pages 378–393. Springer.

Chen Zhang, Qiuchi Li, and Dawei Song. 2019a. Aspect-based sentiment classification with aspect-specific graph convolutional networks. In *EMNLP/IJCNLP (1)*, pages 4567–4577. Association for Computational Linguistics.

Chen Zhang, Qiuchi Li, and Dawei Song. 2019b. Syntax-aware aspect-level sentiment classification with proximity-weighted convolution network. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21-25, 2019*, pages 1145–1148. ACM.

Jiong Zhang, Yibo Lin, Zhao Song, and Inderjit S. Dhillon. 2018. Learning long term dependencies via fourier recurrent units. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 5810–5818. PMLR.

Jianan Zhao, Xiao Wang, Chuan Shi, Binbin Hu, Guojie Song, and Yanfang Ye. 2021a. Heterogeneous graph structure learning for graph neural networks. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 4697–4705. AAAI Press.

Pinlong Zhao, Linlin Hou, and Ou Wu. 2020. Modeling sentiment dependencies with graph convolutional networks for aspect-level sentiment classification. *Knowl. Based Syst.*, 193:105443.

Tong Zhao, Yozen Liu, Leonardo Neves, Oliver J. Woodford, Meng Jiang, and Neil Shah. 2021b. Data augmentation for graph neural networks. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 11015–11023. AAAI Press.

Jie Zhou, Jimmy Xiangji Huang, Qinmin Vivian Hu, and Liang He. 2020. Modeling multi-aspect relationship with joint learning for aspect-level sentiment classification. In *DASFAA (1)*, volume 12112 of *Lecture Notes in Computer Science*, pages 786–802. Springer.

Yuxiang Zhou, Lejian Liao, Yang Gao, Zhanming Jie, and Wei Lu. 2021. To be closer: Learning to link up aspects with opinions. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 3899–3909. Association for Computational Linguistics.

Yanqiao Zhu, Weizhi Xu, Jinghao Zhang, Qiang Liu, Shu Wu, and Liang Wang. 2021. Deep graph structure learning for robust representations: A survey. *CoRR*, abs/2103.03036.