Graph Adaptive Semantic Transfer for Cross-domain Sentiment Classification

Kai Zhang¹, Qi Liu¹, Zhenya Huang¹, Mingyue Cheng¹, Kun Zhang², Mengdi Zhang³, Wei Wu³, Enhong Chen¹

¹ University of Science and Technology of China
² Hefei University of Technology
³ Meituan

SIGIR 2022
Outline

1. Introduction
2. Related Work
3. The GAST Model
4. Experiment
5. Conclusions
Cross-domain Sentiment Classification (CDSC)

Refers to the use the similarity knowledge learned in the source domain to the target domain.

- **Source → Target**
- **Single → Invariant**

Values

The use of similarities between tasks to apply knowledge learned from a old field to a new field.

- **Unlabeled data**
- **Model versatility**
- **Cold start**
- **Weak computing ...**
- **Text**
- **Picture**
- **Attitude**
Challenges

1. Making full use of domain shared features is pivotal for cross-domain sentiment classification. So, what information do we need to focus on in the whole sentence? How to pay attention?

2. How to mine and utilize this information that is critical for cross-domain classification tasks?
Related Work

Source Domain \( \sim P_S(X, Y) \)

lots of labeled data

\[ D_S = \{(x_i, y_i), \forall i \in \{1, \ldots, N\}\} \]

Target Domain \( \sim P_T(Z, H) \)

unlabeled or limited labels

\[ D_T = \{(z_j,?), \forall j \in \{1, \ldots, M\}\} \]

- Unlabeled problem
  - Sample re-weighting
  - Subspace matching
  - Deep methods

Maximum Mean Discrepancy M. Long, et al. ICML 2015

Reverse Gradient Y. Ganin and V. Lempitsky ICML 2015
Domain-Adversarial Neural Networks (ICML 2015)

- **Input**: $x \in X$, input space of the image data
- **Output**: $y \in Y$, $Y = \{1, 2, 3, \ldots, n\}$, label space of image data
- **Goal**: predict the classification labels of input images in the target domain
IATN model

• Construct a better sentiment transfer network
• Make better use of Aspect information

Contains two parts:

1. S-Net
   • Learn shared information between domains
2. A-Net
   • Modeling Aspect Information
The GAST Model

Limitation of previous methods

1. DNN-based/ PLMs-based methods have become mainstream Encoders, and the model architecture is based on the evolution of the DANN framework
2. Design different deep, dedicated to mining better cross-domain semantic features

There are some problems
1. Semantic representation: Failure to fully utilize the internal information of the text (parsing & relation)
2. Feature transfer: ignoring important syntactic information in the process of cross-domain semantic transfer

GAST: Graph Adaptive Semantic Transfer for Cross-domain Sentiment Classification
1. Sentiment words play a crucial role in CDSC, POS-tags can distinguish sentiment words (e.g., “horrible” and “interesting” in the Figure) via the POS-tag “JJ” in a natural way, i.e., the “JJ” label means the word is an adjective.

This movie is unbelievably **horrible**, please reconsider watching.

The plot of this book was quite **interesting**; it is worth reading!

---

**GAST: Graph Adaptive Semantic Transfer for Cross-domain Sentiment Classification**
1. The sentiment polarity of reviews is largely influenced by the sentiment word’s neighbors, whether they are in-domain or across-domain; Meanwhile, different neighbors’ syntactic relations also have different influences for each sentiment word.

**GAST: Graph Adaptive Semantic Transfer for Cross-domain Sentiment Classification**
As shown in Figure, the syntactic graph structures of sentences in different domains are remarkably similar, which means that the syntactic rules are domain-invariant and can be naturally transferred across domains.
The GAST Model

1. Generally, GAST improves the semantic representation and transferable knowledge between domains by aggregating the information from both word sequences and syntactic graphs.

2. GAST mainly contains two modules to learn comprehensive semantics:
   - The first is **POS-based Transformer** (POS-Transformer, a in Figure).
   - The other is **Hybrid Graph Attention** (HGAT, b in Figure).
The GAST Model

Detail of the GAST model

1. We project the word’s embedding matrix $E$ into the query, key, and value matrices.
2. Apart from word’s embeddings, we also map the whole tag embedding matrix.

$$Z = \text{concat}(z_1, z_2, \ldots, z_I),$$

$$z_i = \text{Att.}(Q_i, K_i, V_i) + \text{Att.}(Q^t_i, K^t_i, V_i),$$

$$\text{Att.}(Q, K, V) = \text{softmax} \left( \frac{QK^T}{\sqrt{d/4}} \right) V,$$
The GAST Model

1. Contains two different calculation methods for better relation representation (i.e., relation-aggregation and relation-activation)
2. Through the above two relational functions, we can obtain two syntax-enhanced word representations
**The GAST Model**

**Detail of the GAST model**

**Integrated Adaptive Strategy**

As shown in Figure (c), the strategy includes three loss functions:

1. a **classifier loss** for sentiment knowledge learning
2. a **discriminator loss** for invariant feature extracting
3. a **feature alignment loss** for syntax-aware feature alignment

\[
L_c = -\frac{1}{n_s} \sum_{i=1}^{n_s} (y_s^i \ln \hat{y}_s^i + (1 - y_s^i) \ln (1 - \hat{y}_s^i)),
\]

\[
L_d = -\frac{1}{N} \sum_{i=1}^{N} (y_d^i \ln \hat{y}_d^i + (1 - y_d^i) \ln (1 - \hat{y}_d^i)),
\]

\[
L_a = -\frac{1}{M} \sum_{i=1}^{M} \sum_{j=1}^{C} \hat{y}_j^i \ln \hat{y}_j^i,
\]
Outline

1. Introduction
2. Related Work
3. The GAST Model
4. Experiment
5. Conclusions
Experiment

Dataset Setup & Data Analysis

1. We evaluate GAST on four widely-used Amazon datasets, i.e., DVD (D), Book (B), Electronics (E) and Kitchen (K)

2. We count the ratio of different syntactic relationships as shown in Figure. The proportions between various domains are close, meaning each sentence’s components might be remarkably similar, even in different domains

| Domains   | Testing set percentage |       |       |       |
|-----------|------------------------|-------|-------|-------|
|           | #Train | #Vali. | #Test | #Unlabel |
| Books     | 1,600  | 400   | 2,000 | 4,000   |
| DVD       | 1,600  | 400   | 2,000 | 4,000   |
| Electronics | 1,600 | 400   | 2,000 | 4,000   |
| Kitchen   | 1,600  | 400   | 2,000 | 4,000   |
Our proposed model (BERT-GAST) gets further improvement and achieves a new SOTA. The reason is that GAST is able to consider the sequential information and syntactic structures jointly.

| Baselines       | DVD (D) | Book (B) | Electronics (E) | Kitchen (K) |
|-----------------|---------|----------|-----------------|-------------|
|                 | D→B    | D→E     | D→K            | B→D        | B→E       | B→K        | E→D        | E→B       | E→K        | K→D       | K→B       | K→E       |
| SCL             | 77.8   | 75.2     | 75.5           | 80.4       | 76.5      | 77.1       | 74.5       | 71.6      | 81.7       | 75.2       | 71.3      | 78.8       |
| SFA             | 78.8   | 75.8     | 75.7           | 81.3       | 75.6      | 76.9       | 75.4       | 72.4      | 82.6       | 74.7       | 72.4      | 80.7       |
| DANN            | 80.5   | 77.6     | 78.8           | 83.2       | 76.4      | 77.2       | 77.6       | 73.5      | 84.2       | 75.1       | 74.3      | 82.2       |
| AMN             | 84.5   | 81.2     | 82.7           | 85.6       | 82.4      | 81.7       | 81.7       | 76.6      | 85.7       | 81.5       | 80.9      | 86.1       |
| HATN            | 86.6   | 86.3     | 87.4           | 86.5       | 85.7      | 86.8       | 84.3       | 81.5      | 87.9       | 84.7       | 84.1      | 87.0       |
| IATN            | 87.0   | 86.9     | 85.8           | 86.8       | 86.5      | 85.9       | 84.1       | 81.8      | 88.7       | 84.4       | 84.7      | 87.6       |
| BERT-DAAT       | 90.8   | 89.3     | 90.5           | 89.7       | 89.5      | 90.7       | 90.1       | 88.9      | 93.1       | 88.8       | 87.9      | 91.7       |
| LSTM            | 75.6   | 73.4     | –              | 78.6       | 75.2      | –          | 72.2       | 69.6      | –          | –         | –        | –          |
| TextGCN         | 80.8   | 77.6     | 79.2           | 85.3       | 81.1      | 79.7       | 82.6       | 78.2      | 82.3       | 83.3       | 84.1      | 81.7       |
| FastGCN         | 81.6   | 80.6     | 81.1           | 86.0       | 82.7      | 82.0       | 83.5       | 78.7      | 84.5       | 84.2       | 85.7      | 83.4       |
| GAST            | 87.9   | 87.3     | 89.1           | 88.2       | 86.2      | 87.4       | 85.6       | 83.4      | 89.3       | 87.7       | 87.5      | 89.4       |
| BERT-GAST       | 91.1   | 90.7     | 92.1           | 90.4       | 91.2      | 91.5       | 90.7       | 89.4      | 93.5       | 89.7       | 89.2      | 92.6       |
| G_Non_Pos-Tran. | 85.9   | 84.7     | 87.6           | 86.8       | 83.4      | 85.5       | 84.2       | 80.4      | 87.8       | 85.8       | 85.5      | 87.4       |
| G_Non_HGAT      | 86.6   | 85.9     | 88.1           | 87.4       | 85.0      | 86.1       | 84.5       | 81.3      | 88.2       | 86.4       | 86.7      | 88.2       |
| G_Non_IDS       | 87.2   | 86.6     | 87.9           | 87.6       | 85.8      | 86.7       | 85.0       | 82.6      | 88.5       | 85.9       | 86.2      | 87.7       |
| G_Non_agg       | 87.5   | 86.7     | 88.9           | 88.0       | 85.9      | 86.9       | 85.2       | 82.6      | 89.0       | 87.3       | 87.2      | 89.1       |
| G_Non_act       | 87.3   | 86.3     | 88.7           | 87.7       | 85.3      | 86.2       | 84.8       | 81.8      | 88.7       | 86.9       | 87.1      | 88.7       |
1. The vanilla transformer makes extra decisions on some unrelated word (e.g., “this”, “please”) and pays much attention to these uncritical words.

2. On the contrary, POS-transformer can alleviate this problem by revising attention scores with the help of POS tags.

3. HGAT could deal with the problem more appropriately through the domain-invariant syntactic relations between words.

(a) Case Example 1: The attention values related to the word “horrible”.

|          | This | movie | is   | unbelievably | horrible | please | reconsider | watching |
|----------|------|-------|------|--------------|----------|--------|------------|----------|
| Transformer | 0.08 | 0.11  | 0.09 | 0.16         | 0.15     | 0.19   | 0.13       | 0.09     |
| POS-Transformer | 0.02 | 0.37  | 0.11 | 0.21         | 0.18     | 0.06   | 0.15       | 0.10     |
| Hybrid GAT  | 0.00 | 0.22  | 0.12 | 0.26         | 0.21     | 0.00   | 0.19       | 0.00     |

(b) Case Example 2: The attention values related to the word “interesting”.

|          | the | plot | of | this | book | was | quite | interesting | it | is | worth | reading |
|----------|-----|------|----|------|------|-----|-------|-------------|----|----|-------|---------|
| Transformer | 0.05 | 0.08 | 0.03 | 0.06 | 0.14 | 0.07 | 0.13  | 0.18        | 0.18| 0.05| 0.03  | 0.14    | 0.04    |
| POS-Transformer | 0.03 | 0.09 | 0.02 | 0.02 | 0.12 | 0.07 | 0.17  | 0.21        | 0.21| 0.02| 0.05  | 0.12    | 0.08    |
| Hybrid GAT   | 0.00 | 0.16 | 0.00 | 0.00 | 0.21 | 0.28 | 0.24  | 0.00        | 0.00| 0.00| 0.11  | 0.00    |
Experiment

Adaptive Efficiency & Adaptive Graph

The GAST model with 40% samples even performs better than IATN with 80% samples, which proves that GAST gains an advanced adaptive ability and efficiency in CDSC.

| Syntax Parser          | $D \rightarrow B$ | $D \rightarrow E$ | $D \rightarrow K$ |
|------------------------|-------------------|-------------------|-------------------|
| (1) Without Graph      | 86.6              | 85.9              | 88.1              |
| (2) Stanford Graph     | 87.1 (+0.5)       | 86.6 (+0.7)       | 88.6 (+0.5)       |
| +compare with (1)      |                   |                   |                   |
| (3) Biaffine Graph     | 87.9 (+1.3)       | 87.3 (+1.4)       | 89.1 (+1.0)       |
| +compare with (1)      |                   |                   |                   |
|                        |                   |                   |                   |
|                        |                   |                   |                   |

The quality of the syntactic graph affect the final results, which shows the effectiveness of syntactic features for CDSC research.
1. Introduction
2. Related Work
3. The GAST Model
4. Experiment
5. Conclusions
Conclusion

In this Work

- Learn what?
  - Adaptive Graph Features (pos-tags & dependency relations)
- How to learn?
  - Graph Adaptive Semantic Transfer (GAST) model
- Experimental analysis

In the Feature

- Further characterization of graph structural information
- Dynamic semantic change & dynamic graph learning
See more details in our paper:

**Graph Adaptive Semantic Transfer for Cross-domain Sentiment Classification**
http://home.ustc.edu.cn/~sa517494/files/sigir22.pdf