Multi-Domain Sentiment Classification Based on Domain-Aware Embedding and Attention

Yitao Cai and **Xiaojun Wan**
Institute of Computer Science and Technology, Peking University
The MOE Key Laboratory of Computational Linguistics, Peking University
Center for Data Science, Peking University
{caiyitao, wanxiaojun}@pku.edu.cn

Abstract

Sentiment classification is a fundamental task in NLP. However, as revealed by many researches, sentiment classification models are highly domain-dependent. It is worth investigating to leverage data from different domains to improve the classification performance in each domain. In this work, we propose a novel completely-shared multi-domain neural sentiment classification model to learn domain-aware word embeddings and make use of domain-aware attention mechanism. Our model first utilizes BiLSTM for domain classification and extracts domain-specific features for words, which are then combined with general word embeddings to form domain-aware word embeddings. Domain-aware word embeddings are fed into another BiLSTM to extract sentence features. The domain-aware attention mechanism is used for selecting significant features, by using the domain-aware sentence representation as the query vector. Evaluation results on public datasets with 16 different domains demonstrate the efficacy of our proposed model. Further experiments show the generalization ability and the transferability of our model.

1 Introduction

Sentiment classification is one of the most fundamental tasks in Natural Language Processing. Over the past decades, many supervised machine learning methods such as Naive Bayes, Support Vector Machines, Neural Networks are applied to this task [McCallum *et al.*, 1998; Choi and Cardie, 2008; Kim, 2014; Yang *et al.*, 2016; Zhang *et al.*, 2018]. However, sentiment classification models are highly domain-dependent, resulting in the demand of a large amount of training data for each domain. The reason is due to that there are usually different words and expressions in different domains and moreover, even the same word in different domains may reflect different sentiment polarities. For example, the word *easy* often exists when the sentence conveys positive sentiment in the domain of baby products (e.g. *It is easy for her to hold ...*). But in the domain of movie reviews, *easy* sometimes may express negative sentiment (e.g. *The ending of this movie is easy to guess.*). Similarly, the word *infantile* cannot show speakers' sentiment when they make a comment on baby products, while this word can convey strong negative sentiment when it comes to book reviews or movie reviews.

Therefore, it is worthwhile to leverage available resources across all domains to improve the performance of sentiment classification on some certain domains. One approach is to first pre-train models on large unlabeled data, and then fine-tune those models like BERT [Devlin *et al.*, 2018] and OpenAI GPT [Radford *et al.*, 2018].

Another approach is multi-task learning [Caruana, 1997], which is the focus of this work and effective in improving the performance of one task with the help of related tasks. Some existing Share-Private models exploit multiple LSTMs, CNNs, Memory Networks and Fully-Connected Networks to represent shared and private layers. Shared layers are trained across all data, while private layers are trained on data from a certain domain [Liu *et al.*, 2016a; Liu *et al.*, 2017; Chen and Cardie, 2018]. However, training these models are difficult and time-consuming due to their large number of parameters. The performance of these models are not very well. Another Share-Private model utilizes shared sentence encoders but private query features to select domain-specific information from shared sentence representation [Zheng *et al.*, 2018]. However, this model only focuses on detecting the important words but ignores identifying sentiment polarity of the same word in different domains.

In order to solve the above problem, we propose a novel completely-shared multi-task learning model in this work. The key characteristic of our model is that it can learn domain-aware word embeddings and make use of domain-aware attention mechanism with completely-shared parameters across different domains. Our model first utilizes BiLSTM for domain classification and extracts domain-specific features for words, which are then combined with general word embeddings to form domain-aware word embeddings. Domain-aware word embeddings are fed into another BiLSTM to extract sentence features. The domain-aware attention mechanism is used for selecting significant features by using domain-aware query vector. As can be seen, we share the general word embeddings across different domains, but learn domain-specific word features for each domain. The use of domain-aware word embeddings and domain-aware attention makes the final representation of an input text in a

domain contain both shared information across domains and private information in the specific domain.

Evaluation results on public datasets with 16 different domains demonstrate the efficacy of our proposed model, which achieves state-of-the-art performance on the multi-domain sentiment classification task. Further experiments show the generalization ability and the transferability of our model. Taking sentences with the word *easy* as example, our model can not only focus on the word *easy* for sentiment classification, but also identify different meanings of it according to different domains and different instances.

Our main contributions are summarized as follows:

- We propose a novel completely-shared neural model based on domain-aware word embeddings and domain-aware attention mechanism to make use of training data among all domains for multi-domain sentiment classification. Our model can not only focus on the significant words in texts but also distinguish their sentiment polarity with the help of domain classifier.
- We conduct experiments on datasets with 16 different domains. The results of multi-domain sentiment classification show the effectiveness of our model.
- We further perform experiment on cross-domain sentiment classification and knowledge transferring. Results reveal the generalization ability and transferability of our model. Our code will be released.

2 Related Work

Multi-task learning (MTL) [Caruana, 1997] is a popular method to solve multi-domain text classification or sentiment classification. Many works of deep learning based multi-task learning [Zhang *et al.*, 2014; Liu *et al.*, 2016b] share first several layers for each task to extract low-level features and generate outputs with task-specific parameters. Recent works apply adversarial training to multi-task learning. Liu et al. [2017] ,Chen and Cardie [2018], Liu et al. [2018] and Li et al. [2017] use LSTMs, CNNs and memory networks to extract features and attempt to confuse domain classifiers by maximizing cost function in order to guarantee the task-independence of features.

Attention mechanism has become popular since it enables models to focus on the more important words and phrases. Bahdanau et al. [2014], Luong et al. [2015], Vaswani et al. [2017] apply attention to the machine translation task to capture long distance dependency. Yang et al. [2016] apply attention at word level as well as sentence level so as to introduce importance of sentences to document classification. In multi-task learning, Zheng et al. [2018], Yuan et al. [2018] share feature extractors but use attention mechanism with task-independent query vectors to generate different representations for different tasks.

Language model[Bengio *et al.*, 2003] can express context information of a word compared to general word embedding. ELMo [Peters *et al.*, 2018] and BERT [Devlin *et al.*, 2018] have proven the reliability of language model containing context information. Inspired by these works, we exploit the output of an LSTM for domain classification to capture domain and context information.

3 Existing MTL Models

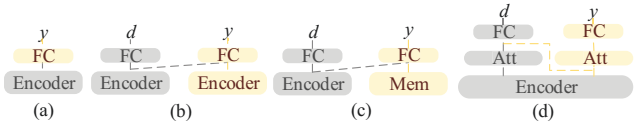


Figure 1: Overview of some existing models (*d* means domain label and *y* is sentiment label. Gray blocks are shared layers and yellow ones are private layers)

Recently, many researches focus on multi-domain sentiment classification or text classification. We will introduce several latest models, which are also considered strong baselines for comparison.

Figure (1a) is the architecture of Fully-Shared Multi-Task Learning framework (**FS-MTL**). This framework is one of the most common multi-task learning frameworks. There is a shared encoder to extract low-level features for all domains, and then a domain-specific classifier is used to make prediction for each domain. However, this model cannot extract domain-specific features since feature extractors are trained on all domains.

Many share-private models utilize multiple encoders to extract domain-specific and domain-independent features. Shared encoders are trained on all training data and private encoders are trained on domain-specific data. Adversarial training is applied to enable shared encoders to extract domain-independent features. **ASP-MTL** (Figure 1b) [Liu *et al.*, 2017] and **DSR-at** (Figure 1c) [Liu *et al.*, 2018] are two specific models of this share-private scheme. ASP-MTL uses another encoder as private feature extractor while DSR-at uses memory network. However, if the number of domains increases, these models require large space and cost a lot of time to train.

Another share-private model **DA-MTL** (Figure 1d) [Zheng *et al.*, 2018] uses a shared BiLSTM encoder to extract low-level features. This model first builds domain representations. Domain representations are used for attention to select domain-specific sentiment information from the shared sentence representation. This model can focus on more significant words and is space-efficient, but cannot represent different meanings and different sentiment polarities of the same word in different domains.

4 Our Proposed Model

The goal of our proposed model is to pay more attention to significant words in a text and identify their sentiment polarities in each specific domain. Our model consists of two parts: domain feature extraction and classification, sentiment feature extraction and classification. The first part aims to extract domain-specific features with a BiLSTM and a domain classifier. This part attempts to capture domain information and context information in the given text. Therefore, the training loss of the domain classifier is one part of the total training loss in order to guarantee the BiLSTM to extract domain information. The second part aims to get the text representation for sentiment classification. It first forms domain-aware

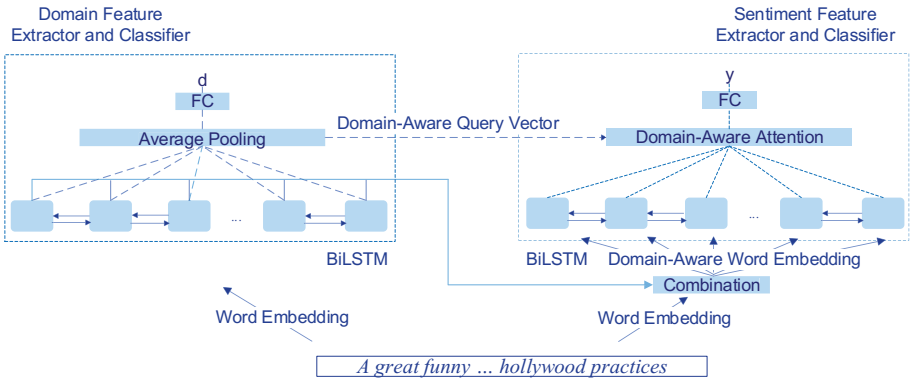


Figure 2: Overview of our proposed model (d means domain label and y is sentiment label)

word embedding by combining domain-specific features and general word embedding, and then obtain the initial representation of the text by a BiLSTM with the domain-aware word embedding. After that, it uses a domain-aware attention mechanism to refine the initial representation and obtains the final representation for sentiment classification. In the attention mechanism, different query vectors are formed for different texts in different domains, aiming at paying attention to important words for sentiment classification. The training loss for sentiment classification and the loss for domain classification are added up, making two objectives learned simultaneously. The architecture of our proposed model is shown in Figure 2. Note that all the parameters in the two parts are completely shared across different domains. We will introduce key components in the following subsections, respectively.

4.1 Domain Feature Extraction and Domain Classification

Representation of each word that involves domain information and context information is required. After the general word embeddings for a sequence of words in a text are taken as input in a BiLSTM, the output vectors of the BiLSTM at different time steps are considered to contain context information of the text. The output vectors are then combined by average pooling to get the text representation, and the training objective of predicting the domain correctly by the text representation forces the BiLSTM to extract domain information. We exploit BiLSTM as domain feature extractor and a fully-connected network as domain classifier. The output vectors of BiLSTM at different time steps are considered domain-specific features for the words. The formulas can be written as follows.

$$\begin{aligned}\overrightarrow{h_{f_t}} &= LSTM(\overrightarrow{h_{f_{t-1}}}, e_t, \theta_f)\\ \overleftarrow{h_{b_t}} &= LSTM(\overleftarrow{h_{b_{t-1}}}, e_t, \theta_b)\\ h_t &= \overrightarrow{h_{f_t}} \oplus \overleftarrow{h_{b_t}}\end{aligned} \tag{1}$$

$$\begin{aligned}h_{text} &= \frac{1}{E}\sum_{t=1}^{E} h_t\\ d &= softmax(W_2 ReLU(W_1 h_{text} + b_1) + b_2)\\ L_d &= -\sum_{i=1}^{N} \hat{d}_i log_2 d_i\end{aligned}$$

where $\overrightarrow{h_{f_t}}$ and $\overleftarrow{h_{b_t}}$ are forward and backward LSTM output vectors at time step t. e_t is the word embedding for the t-th word. θ_f and θ_b are parameters of two LSTMs. $\oplus$ is the concatenation operation. E is the length of text. d is the predicted probabilities (a vector) of domains for the text. W_1, W_2, b_1 and b_2 are weight matrices and bias vectors. L_d is the cross entropy loss function of domain classifier and N is the number of domains. $\hat{d}$ is the gold probability distribution over domains.

4.2 Domain-Aware Word Embedding

We attempt to form domain-aware word embedding that contains domain features as well as context information for sentiment feature extraction. Therefore we combine domain features mentioned in Section 4.1 and general word embedding for each word. The formulas of building domain-aware word embedding x_t are listed as follows.

$$\begin{aligned}\tilde{h_t} &= W_3 h_t + b_3\\ x_t &= e_t \oplus \tilde{h_t}\end{aligned} \tag{2}$$

where W_3, b_3 are weight matrix and bias vector and the linear transformation aims to extract higher-level domain-specific information. h_t is the output vector of domain feature extractor, e_t is the word embedding for the t-th word, as mentioned in Section 4.1, $\oplus$ is concatenation operation.

Domain-aware word embedding x_t is then fed into another BiLSTM to extract sentiment features.

4.3 Domain-Aware Attention and Sentiment Feature Extraction

Our proposed model generates query vectors in the attention mechanism dynamically, expecting the model can leverage

Dataset	Train	Dev	Test	Avg Length
Books	1400	200	400	159
Electronics	1398	200	400	101
DVD	1400	200	400	173
Kitchen	1400	200	400	89
Apparel	1400	200	400	57
Camera	1397	200	400	130
Health	1400	200	400	81
Music	1400	200	400	136
Toys	1400	200	400	90
Video	1400	200	400	156
Baby	1300	200	400	104
Magazine	1370	200	400	117
Software	1315	200	400	129
Sports	1400	200	400	94
IMDB	1400	200	400	269
MR	1400	200	400	21

Table 1: Statistics of datasets

domain information to select important words in a text. The formulas of the attention mechanism are listed below. In the attention mechanism, we apply additive attention instead of dot-product attention, since the performance of former attention is better.

$$\begin{aligned} h'_t &= o_t \oplus h_{text} \\ \beta_t &= W_5 tanh(W_4 h'_t + b_4) + b_5 \\ \alpha_t &= softmax(\beta_t) \\ o_{text} &= \sum_{t=1}^{E} \alpha_t o_t \end{aligned} \tag{3}$$

where o_t is the output vector (by concatenating the forward and backward output vectors) of the BiLSTM for extracting sentiment features, as mentioned in the last paragraph of Section 4.2. h_{text} is the vector mentioned in Equation (1), α_t is the weight of the t-th word, E is the length of text, and o_{text} is the weighted sum of output vector o_t, which is the extracted sentiment features for sentiment classification. W_4, W_5, b_4, b_5 are weight matrices and bias vectors.

4.4 Sentiment Classification

We use a two layer fully-connected neural network as our final sentiment classifier rather than several classifiers corresponding to different domains. The loss function L_s is cross entropy between the predicted labels and true labels.

The formulas can be written as follows.

$$\begin{aligned} p &= \sigma(W_7 ReLU(W_6 o_{text} + b_6) + b_7) \\ L_s &= -(y log_2 p + (1-y) log_2 (1-p)) \end{aligned} \tag{4}$$

where W_6, W_7, b_6, b_7 are weight matrices and bias vectors, p is the probability of predicting the text as positive, and y is 1 if the gold label is positive and 0 otherwise.

The total loss of our model can be computed as follows:

$$L_{all} = \gamma_d L_d + \gamma_s L_s$$

where γ_d and γ_s are weights of loss functions L_d and L_s, respectively.

5 Experiments

5.1 Datasets

We use the datasets released by [Liu *et al.*, 2017] for multi-domain sentiment classification, which consist of product and movie reviews in 16 different domains. The data in each domain is randomly split into training set, development set and test set according to the proportion of 70%, 10%, 20%. Statistics of the 16 datasets are listed in Table 1.

5.2 Training Details

We initialize word embedding with 200-dimension Glove vectors [Pennington *et al.*, 2014]. The word embedding is fixed during training. Other parameters are initialized by sampling from normal distribution whose standard deviation is 0.1. The minibatch is 128. Each batch contains 8 samples from every domain. We use Adam optimizer [Kingma and Ba, 2014] with an initial learning rate of 0.004. The hidden size of each LSTM is 128. Weights γ_d and γ_s of domain classification loss and sentiment classification loss are set to 0.1 and 1 respectively after a small grid search over [1, 0.1, 0.05]. To alleviate overfitting, we use dropout with probability of 0.5 and L2 regularization with parameter of 1e-8. We train the domain classifier and the sentiment classifier jointly at first. After 10 epochs, we only train the sentiment classifier, setting γ_d to 0. At last, we finetune our model on each task.

5.3 Multi-Domain Sentiment Classification

We first conduct experiments of multi-domain sentiment classification. Models are tested on 16 test sets, respectively. Latest models mentioned in Section 3 are used for comparison. In addition, we also compare with the following baselines.

- **Single Task**. BiLSTM, BiLSTM with attention mechanism (att-BiLSTM) and text CNN are three of the most popular models designed for the sentiment classification task. The model is trained on each domain independently.
- **BERT**. BERT [Devlin *et al.*, 2018] is a pre-trained language model with deep bidirectional transformer. Finetuning BERT model can outperform state-of-the-art models among many tasks, including text classification. We use the pre-trained BERT-base model available online and finetune it on each task.
- **SA-MTL**. SA-MTL [Zheng *et al.*, 2018] is the simplified version of DA-MTL. It shares BiLSTM as low-level feature extractor. This model uses different query vector for each domain rather than generating query vector for each instance. The query vector is updated during the training process.

Results of our proposed model and baseline models are listed in Table 2. The table shows that both pre-trained language models and multi-task learning models can improve the accuracy of multi-domain sentiment classification a lot. Multi-task learning baseline models except FS-MTL achieve similar performance with BERT model, but they demand less space and are easier to train. Our proposed model outperforms other models by at least 1.9 points on accuracy.

	BiLSTM	att-BiLSTM	CNN	BERT	FS-MTL	ASP-MTL	SA-MTL	DA-MTL	DSR-at	Our model
Books	81.0	82.0	85.3	87.0	82.5	87.0	86.8	88.5	**89.1**	89.0
Electronics	81.8	83.0	87.8	88.3	85.7	89.0	87.5	89.0	87.9	**91.8**
DVD	83.3	83.0	76.3	85.6	83.5	87.4	87.3	88.0	88.1	**88.3**
Kitchen	80.8	80.3	84.5	**91.0**	86.0	87.2	89.3	89.0	85.9	90.3
Apparel	87.5	86.5	86.3	**90.0**	84.5	88.7	87.3	88.8	87.8	89.0
Camera	87.0	89.5	89.0	90.0	86.5	91.3	90.3	91.8	90.0	**92.0**
Health	87.0	84.3	87.5	88.3	88.0	88.1	88.3	90.3	**92.9**	89.8
Music	81.8	82.0	81.5	86.8	81.2	82.6	84.0	85.0	84.1	**88.0**
Toys	81.5	85.0	87.0	91.3	84.5	88.8	89.3	89.5	85.9	**91.8**
Video	83.0	83.5	82.3	88.0	83.7	85.5	88.5	89.5	90.3	**92.3**
Baby	86.3	86.0	82.5	91.5	88.0	89.8	88.8	90.5	91.7	**92.3**
Magazine	92.0	92.0	86.8	92.8	92.5	92.5	92.0	92.0	92.1	**96.5**
Software	84.5	83.0	87.5	89.3	86.2	87.3	89.3	90.8	87.0	**92.8**
Sports	86.0	84.8	85.3	**90.8**	85.5	86.7	89.8	89.8	85.8	**90.8**
IMDB	82.5	83.5	83.3	85.8	82.5	85.8	87.5	89.8	**93.8**	90.8
MR	74.8	76.0	**79.0**	74.0	74.7	77.3	73.0	75.5	73.3	77.0
Avg	83.7	84.0	84.3	88.1	84.7	87.2	87.6	88.2	87.9	**90.1**

Table 2: Results of multi-domain sentiment classification

	ASP-MTL	DSR-at	Our model
Books	81.5	85.8	**87.3**
Electronics	83.8	**89.5**	85.8
DVD	84.5	86.3	**88.8**
Kitchen	87.5	**88.3**	88.0
Apparel	85.3	85.8	**88.0**
Camera	85.3	88.8	**90.0**
Health	86.0	90.5	**91.0**
Music	81.3	84.8	**86.5**
Toys	88.0	**90.3**	**90.3**
Video	86.8	85.3	**91.3**
Baby	86.5	84.8	**90.3**
Magazine	87.0	84.0	**88.5**
Software	87.0	**90.8**	89.8
Sports	87.0	87.0	**90.5**
IMDB	84.0	83.3	**85.8**
MR	72.0	**76.3**	75.5
Avg	84.6	86.3	**87.9**

Table 3: Results of cross-domain sentiment classification

5.4 Cross-Domain Sentiment Classification

In the task of cross-domain sentiment classification, models are trained on the training data in 15 domains and tested on the test data in the remaining one domain. It means we do not use the sentiment labels of the training data in the test domain, but we still use the data as unlabeled data for domain classification. In other words, all training data in 16 domains are used for training domain feature extractor and domain classifier, but only the training data in 15 domains are used for training sentiment feature extractor and sentiment classifier. We assume that the combination of domain features as well as general word embedding contains domain and context based sentiment features, which is beneficial for sentiment classification and improves the generalization ability of the sentiment classifier.

Previous models ASP-MTL and DSR-at mentioned in Section 5.3 can also be adapted for cross-domain sentiment classification. In ASP-MTL, we use average pooling to combine outputs from all domains. As for DSR-at, the test data is predicted by 15 memory networks, after which the majority voting is applied to classify the sentiment polarity of the test data.

The results are shown in Table 3. Compared to Table 2, the accuracy of cross-domain sentiment classification is worse than that of multi-domain sentiment classification, revealing that the sentiment classification task is highly domain-dependent. Our proposed model outperforms DSR-at by 1.8 points on accuracy and outperforms ASP-MTL by 3.3 points on accuracy, demonstrating the generalization ability of our model.

5.5 Transferability Test

We assume that the shared domain feature extractor can learn enough knowledge about domain and context to be applied to unseen domains. Therefore, we test the transferability of our model, following the transfer learning setting of [Zheng *et al.*, 2018]. We first use the last 10 domains to train the model with multi-task learning. We then keep the parameters of domain feature extractor fixed and finetune other parameters.

We compare our model with previous models SA-MTL and DA-MTL. These two models fix their sentence encoders after pre-training it on the last 10 tasks. We also compare models with transfer learning mentioned above and models with multi-task learning based only on the first 6 domains.

	SA-MTL	DA-MTL	Our model
multi-task learning	84.4	87.0	87.8
transfer learning	86.5	87.7	89.0
Δ acc	+1.9	+0.7	+1.2

Table 4: Results of the first 6 domains with multi-task learning and transfer learning

	SA-MTL	DA-MTL	Our model
origin	87.6	88.2	90.1
+BERT Features	88.7	88.9	90.5
Δ acc	+1.1	+0.7	+0.4

Table 5: Results of original model and model with BERT features

Table 4 shows the results of the first 6 domains with and without transfer learning of three models. According to the table, transfer learning is beneficial to the performance of sentiment classification, because transfer learning with SA-MTL, DA-MTL and our proposed model outperforms multi-task learning a lot. Besides, our model achieves better transfer learning performance, revealing the transferability of domain feature extractor and domain-aware embedding.

5.6 Introducing Language Model Features

In our model, we exploit domain feature extractor to collect domain information as well as context information. Recently, the success of finetuning language models and applying their features to text classification has demonstrated that pre-trained language models can better capture context information of sentences. Therefore, we introduce BERT features to our model to enrich the context information.

We adapt Equation (2) as follows.

$$\begin{aligned} \tilde{h_t} &= W_3 h_t + b_3 \\ \tilde{B_t} &= W_8 B_t + b_8 \\ x_t &= e_t \oplus \tilde{h_t} \oplus \tilde{B_t} \end{aligned} \tag{5}$$

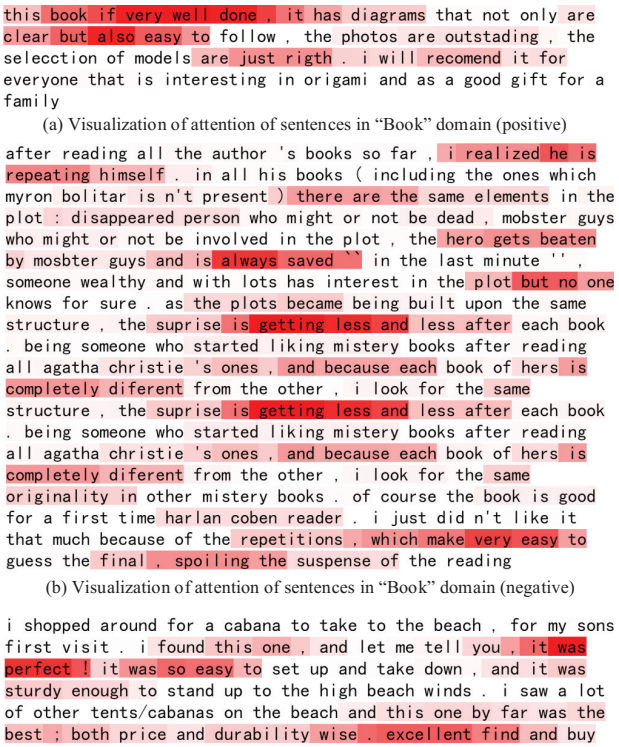
where $\tilde{h_t}$, h_t and so on are the same as that in Equation (2), B_t is the output of BERT model for the t-th word and $\tilde{B_t}$ is the linear transformation of B_t. x_t is the concatenation of $\tilde{B_t}$, e_t, and $\tilde{h_t}$.

We also introduce BERT features to SA-MTL and DA-MTL. In SA-MTL, domain query vectors are replaced by sentence representations extracted by the final layer of BERT. In DA-MTL model, we concatenate generated instance specific query vector and its BERT features as new query vector.

Table 5 shows the results of original models and models with BERT features. The improved performance of models with BERT features shows the positive effect of more context information. The improvement of SA-MTL is larger than DA-MTL and our proposed model, revealing that DA-MTL and our model have already captured more context information than SA-MTL.

6 Visualization Analysis

Figure 3 shows the attention in our model for some running examples. In Figure (3a), our model attends to the words "easy", "rigth"(right), "recommend" and phrase "very well done" to predict positive sentiment for the text, because the phrase "easy to follow", "models are just right", "recommend it" and "well done" convey strong positive polarity to the products. In Figure (3b), our model pays attention to "repeating himself", "getting less and less", "easy" and "spoiling" and succeeds in making correct prediction that the text



(a) Visualization of attention of sentences in "Book" domain (positive)

(b) Visualization of attention of sentences in "Book" domain (negative)

(c) Visualization of attention of sentences in "Baby" domain (positive)

Figure 3: Visualization of Attention (the deeper the red color is, the larger the attention weight is.)

conveys negative sentiment. In Figure (3c), words "perfect", "easy", "best", "excellent" are selected as the evidences of sentiment prediction. Examples above show that our model can choose words and phrases correctly, demonstrating the effectiveness of regarding domain features as query vectors of attention. Besides, in Figure (3a), (3b), (3c), our model succeeds in predicting correct sentiment while selecting the same word "easy", revealing that domain information and context information are beneficial for some instances.

7 Conclusion

In this paper, we propose a novel completely-shared neural model to make use of different training data across all domains. Our model builds domain-aware word embedding to express domain and context information for words, and proposes domain-aware attention mechanism to focus on more significant words in the text. Experiments on multi-domain sentiment classification and cross-domain sentiment classification on 16 different domains demonstrate the effectiveness and advantages of our proposed model.

Acknowledgments

This work was supported by National Natural Science Foundation of China (61772036) and Key Laboratory of Science, Technology and Standard in Press Industry (Key Laboratory of Intelligent Press Media Technology). We thank the anonymous reviewers for their helpful comments. Xiaojun Wan is the corresponding author.

References

[Bahdanau *et al.*, 2014] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[Bengio *et al.*, 2003] Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. A neural probabilistic language model. *Journal of machine learning research*, 3(Feb):1137–1155, 2003.

[Caruana, 1997] Rich Caruana. Multitask learning. *Machine learning*, 28(1):41–75, 1997.

[Chen and Cardie, 2018] Xilun Chen and Claire Cardie. Multinomial adversarial networks for multi-domain text classification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, volume 1, pages 1226–1240, 2018.

[Choi and Cardie, 2008] Yejin Choi and Claire Cardie. Learning with compositional semantics as structural inference for subsentential sentiment analysis. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 793–801. Association for Computational Linguistics, 2008.

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

[Kim, 2014] Yoon Kim. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*, 2014.

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

[Li *et al.*, 2017] Zheng Li, Yu Zhang, Ying Wei, Yuxiang Wu, and Qiang Yang. End-to-end adversarial memory network for cross-domain sentiment classification. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI 2017)*, 2017.

[Liu *et al.*, 2016a] Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. Deep multi-task learning with shared memory. *arXiv preprint arXiv:1609.07222*, 2016.

[Liu *et al.*, 2016b] Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. Recurrent neural network for text classification with multi-task learning. *arXiv preprint arXiv:1605.05101*, 2016.

[Liu *et al.*, 2017] Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. Adversarial multi-task learning for text classification. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1–10. Association for Computational Linguistics, 2017.

[Liu *et al.*, 2018] Qi Liu, Yue Zhang, and Jiangming Liu. Learning domain representation for multi-domain sentiment classification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, volume 1, pages 541–550, 2018.

[Luong *et al.*, 2015] Minh-Thang Luong, Hieu Pham, and Christopher D Manning. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*, 2015.

[McCallum *et al.*, 1998] Andrew McCallum, Kamal Nigam, et al. A comparison of event models for naive bayes text classification. In *AAAI-98 workshop on learning for text categorization*, volume 752, pages 41–48. Citeseer, 1998.

[Pennington *et al.*, 2014] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[Peters *et al.*, 2018] Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*, 2018.

[Radford *et al.*, 2018] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. *URL https://s3-us-west-2. amazonaws. com/openai-assets/research-covers/languageunsupervised/language understanding paper. pdf*, 2018.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017.

[Yang *et al.*, 2016] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489, 2016.

[Yuan *et al.*, 2018] Zhigang Yuan, Sixing Wu, Fangzhao Wu, Junxin Liu, and Yongfeng Huang. Domain attention model for multi-domain sentiment classification. *Knowledge-Based Systems*, 155:1–10, 2018.

[Zhang *et al.*, 2014] Zhanpeng Zhang, Ping Luo, Chen Change Loy, and Xiaoou Tang. Facial landmark detection by deep multi-task learning. In *European conference on computer vision*, pages 94–108. Springer, 2014.

[Zhang *et al.*, 2018] Lei Zhang, Shuai Wang, and Bing Liu. Deep learning for sentiment analysis : A survey. *CoRR*, abs/1801.07883, 2018.

[Zheng *et al.*, 2018] Renjie Zheng, Junkun Chen, and Xipeng Qiu. Same representation, different attentions: Shareable sentence representation learning from multiple tasks. *arXiv preprint arXiv:1804.08139*, 2018.