# Exploiting Document Knowledge for Aspect-level Sentiment Classification


Ruidan He[1,2], Wee Sun Lee[1], Hwee Tou Ng[1], and Daniel Dahlmeier[2]

[1]National University of Singapore | [2]SAP Innovation Centre Singapore


## Introduction

- Aspect-level sentiment classification aims at identifying the sentiment polarity of one specific aspect term in its context sentence. e.g. in "*great food but the service is dreadful*" the sentiments for "*food*" and "*service*" are positive and negative, respectively.
- Past research suggests that attention-based LSTMs work well on this task, where the LSTM aims to capture sequential patterns and the attention aims to emphasise the target-specific contexts.
- However, aspect-level annotation is costly to obtain and insufficient training data limits the effectiveness of LSTM-based methods.

We propose to improve aspect-level classification performance by transferring knowledge from document-level examples, since:

- Document-level datasets are easily accessible, and come with sentiment labels.
- Document-level classification and aspect-level classification are tasks that are highly related semantically.

## Method

### Aspect-level model:

A vanilla attention-based LSTM (LSTM+ATT) serves as our aspect-level baseline model. We extend it with **pretraining (PRET)** and **multi-task learning (MULT)** for incorporating document-level knowledge.

### Document-level model:

A conventional LSTM is used for document-level classification.

### Three transfer settings:

- PRET: first train the document-level LSTM. Then initialise the relevant parameters of LSTM+ATT with the pretrained weights, and train on aspect-level examples to fine tune those weights and learn attention-relevant parameters.
- MULT: simultaneously train the two tasks. The embedding layer and LSTM layer are shared by both tasks, and other parameters are task-specific.
- PRET+MULT: combine the two transfer methods. The pretrained weights from document-level task are used for parameter initialisation for both models. MULT is then performed.

## Code & References

Code and data available at:

*https://github.com/ruidan/Aspect-level-sentiment*

SCAN ME


References:

[1] Tang et al (2016). Effective LSTMs for target-dependent sentiment classification.
[2] Wang et al (2016). Attention-based LSTM for aspect-level sentiment classification.
[3] Tang et al. (2016) Aspect level sentiment classification with deep memory network.
[4] Chen et al (2017). Recurrent attention network on memory for aspect sentiment analysis.


## Experiments & Results

### Datasets:

- Aspect-level: 4 datasets from SemEval 2014, 2015, and 2016 (see Table 1)
- Document-level: 2 datasets derived from Yelp2014 and Amazon Electronics. Each contains 30k instances with balanced labels.

| | Dataset | Pos | Neg | Neu |
|---|---|---|---|---|
| D1 | Restaurant14-Train | 2164 | 807 | 637 |
| | Restaurant14-Test | 728 | 196 | 196 |
| D2 | Laptop14-Train | 994 | 870 | 464 |
| | Laptop14-Test | 341 | 128 | 169 |
| D3 | Restaurant15-Train | 1178 | 382 | 50 |
| | Restaurant15-Test | 439 | 328 | 35 |
| D4 | Restaurant16-Train | 1620 | 709 | 88 |
| | Restaurant16-Test | 597 | 190 | 38 |

Table 1. Aspect-level dataset statistics

### Model Comparison (Table 2):

- PRET+MULT yields the best results. Overall PRET performs better than MULT
- More improvements on macro-F1 compared to accuracy due to the fact that the aspect-level datasets are quite unbalanced.

| Methods | D1 | | D2 | | D3 | | D4 | |
|---|---|---|---|---|---|---|---|---|
| | Acc. | Macro-F1 | Acc. | Macro-F1 | Acc. | Macro-F1 | Acc. | Macro-F1 |
| TD-LSTM[1] | 75.37 | 64.51 | 68.25 | 65.96 | 76.39 | 58.70 | 82.16 | 54.21 |
| ATAE-LSTM[2] | 78.60 | 67.02 | 68.88 | 63.93 | 78.48 | 62.84 | 83.77 | 61.71 |
| MemNet[3] | 76.87 | 66.40 | 68.91 | 62.79 | 77.89 | 59.52 | 83.04 | 57.91 |
| RAM[4] | 78.48 | 68.54 | **72.08** | 68.43 | 79.98 | 60.57 | 83.88 | 62.14 |
| LSTM | 75.23 | 64.21 | 66.79 | 64.02 | 75.28 | 54.10 | 81.95 | 58.11 |
| LSTM+ATT | 76.83 | 66.48 | 68.07 | 64.82 | 77.38 | 60.52 | 82.73 | 59.12 |
| Ours: PRET | 78.28 | 68.55 | 71.32 | **68.53** | 80.67 | 68.31 | 84.87 | **70.73** |
| Ours: MULT | 77.41 | 66.68 | 68.65 | 64.57 | 81.05 | 65.69 | 83.27 | 64.56 |
| Ours: PRET+MULT | **79.11** | **69.73*** | 71.15 | 67.46 | **81.30*** | **68.74*** | **85.58*** | 69.76* |

Table 2. Average accuracies and macro-F1 scores over 5 runs with random initialisation. The best results are in bold. * indicates that PRET+MULT is significantly better than other baseline methods with $p < 0.05$ according to one-tailed unpaired t-test.

### PRET with different layers (Fig. 1):

- Improvements over LSTM+ATT are observed even when only one layer is transferred.
- Transfers of LSTM and embedding layers are more helpful than the output layer since the output layer is normally more task-specific.
- Transfer of the embedding layer is more helpful on datasets whose label distribution is unbalanced such as D3 and D4.

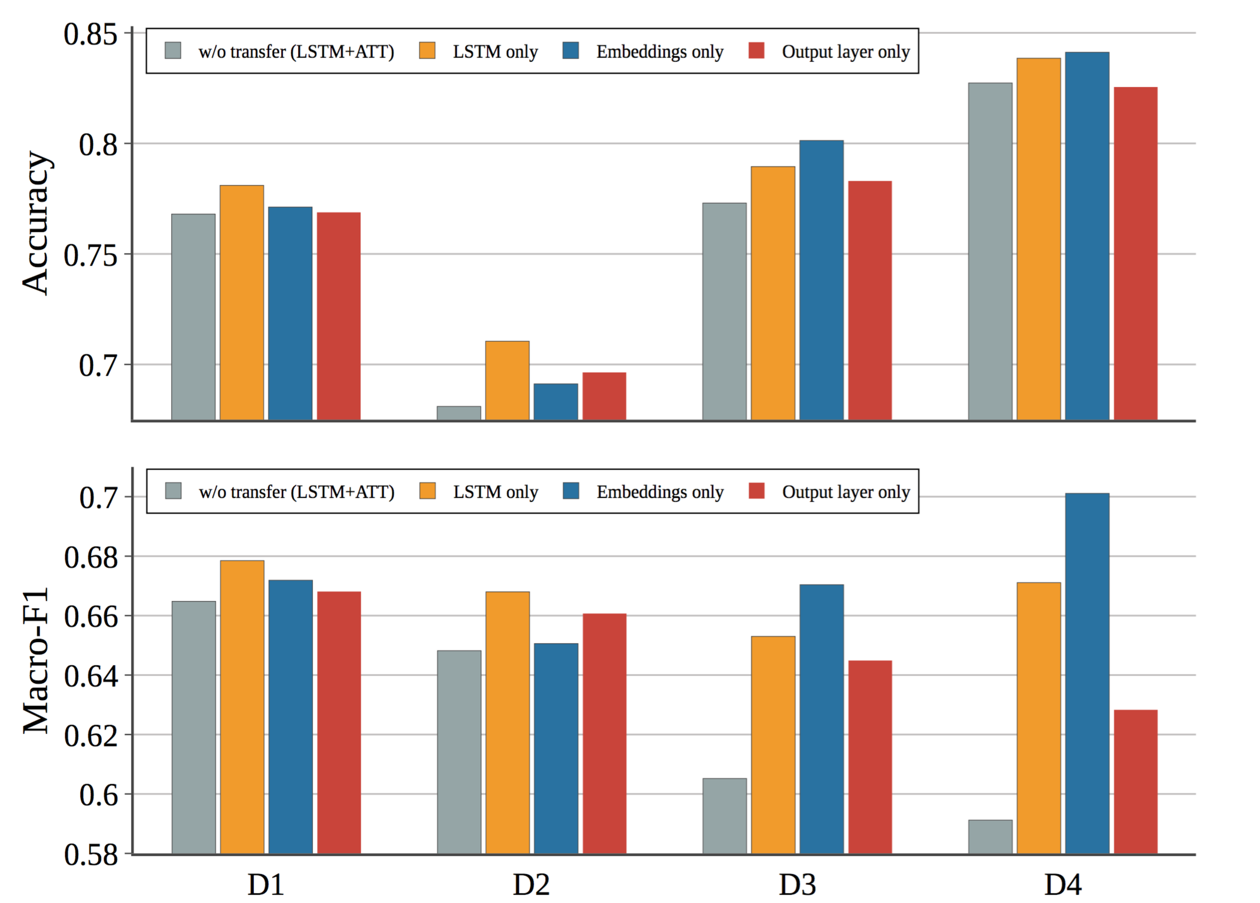


Fig 1. PRET with different layers being transferred. e.g. "LSTM only" denotes the setting where only the LSTM layer is transferred through weight initialisation. Averaged results over 5 runs are reported.

### Varying the size of document-level examples (Fig. 2):

The improvements on accuracies are stable. Macro-F1 scores increase more rapidly.

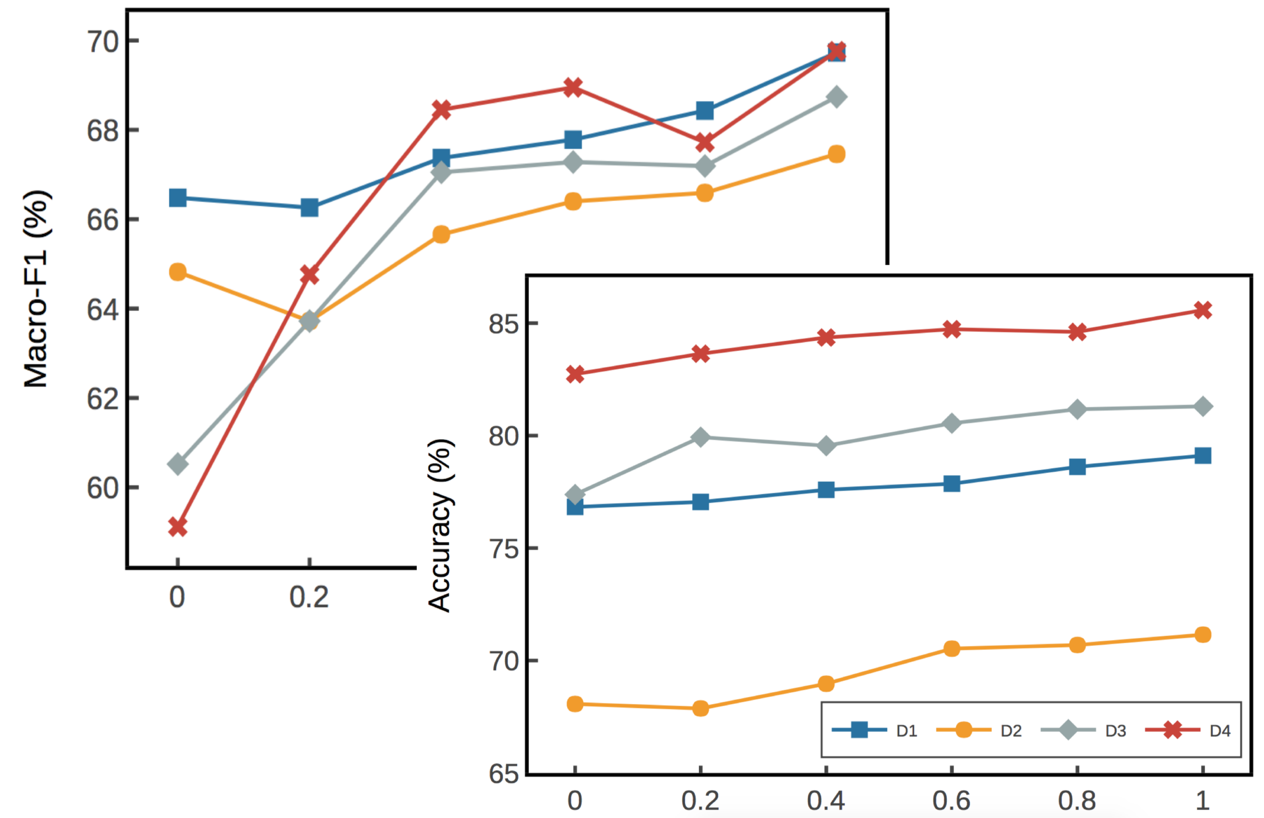


Fig 2. Results of PRET+MULT vs. percentage of document-level training data.

## Analysis & Findings

We find that the benefits brought by transfer learning from document-level examples are typically shown in 4 ways:

- Surprisedly, LSTM+ATT still makes mistakes on instances with common opinion words. One possible reason is that GloVe embeddings do not effectively capture sentiment information, while aspect-level examples are too sparse to capture sentiment for all words. PRET+MULT eliminates this kind of errors by learning from large number of documents.
  - *"I was highly disappointed in the [food]$_{neg}$."*
  - *"This particular location certainly uses substandard [meats]$_{neg}$"*
- PRET+MULT helps to better capture domain-specific opinion words due to additional knowledge from documents that are from a similar domain.
  - *"The smaller [size]$_{pos}$ was a bonus because of space restrictions."*
  - *"The [price]$_{pos}$ is 200 dollars down."*
- LSTM+ATT makes a number of errors on sentences with negation words due to the fact that LSTM may not have enough training data due to the small aspect dataset. PRET+MULT works much better in this case.
  - *"I have experienced no problems, [works]$_{pos}$ as anticipated."*
  - *"[Service]$_{neg}$ not the friendliest to our party!"*
- PRET+MULT makes fewer errors on recognising neutral instances. The lack of training examples makes the prediction of neutral instances very difficult for past works. Our model addresses this issue by learning from additional balanced document-level examples.