Aspect-Based User Preference Analysis

Hanlin Wu, Xi Lu, Sicong Fang, Yunsi Zhang
Data Science Institute
Columbia University
New York, NY 10027
{hw2576, xl2602, sf2796, yz3012}@columbia.edu

Abstract

In this paper, we show how to extract user preference for different aspects of hotel with weight by automatically analyzing reviews. We begin with building a new dataset which contains in total 60,463 unique users and 763,895 hotel reviews from TripAdvisor. Then we adapt a bootstrapping-based algorithm and a Latent Rating Regression (LRR) model (Wang et al., 2010) to segment aspects and calculate weights of the corresponding aspect based on each user's review history. Analysis and visualizations of outputs from different locations highlight the potential of using weight as an indicator of user preference, which can also be used to support hotel recommendation system.

1 Introduction

Using online services, such as purchasing products via Amazon, ordering food via Seamless, and booking hotels via TripAdvisor, has become a growing trend in today's world. As more and more reviews have been posted on the websites, we believe that we can investigate user preference by analysing this data automatically.

One major question that needs to be answered is how we can understand and identify the latent opinions behind reviews. To be more specific, we want to explore which aspect users value most based on the reviews they have written.

We narrow our focus down to hotel reviews from TripAdvisor with assumption that all user reviews are reliable, and state the following hypotheses:

1. **Weight of different aspects in reviews reflects user's preference.**
2. **User preference may vary by places where people share some common characteristics.** In other words, people from adjacent areas usually have similar cultural, financial and other features, and all of these features may potentially influence people's expectation on a specific domain, which is hotel in our case. In contrast, different preference is expected to be seen between people from different places .

In Section 2, we introduce related work in relevant research area. Section 3 shows how we collect hotel review from TripAdvisor, how to process data, and what the obtained data looks like. Then we discuss the aspect segmentation method and LRR model in Section 4 and model implementation in Section 5. After running the model we apply some post data processing steps (Section 5.3) to obtain and combine all the information which we need for analyzing. Section 7 includes data analysis and visualization. Finally, we state some interesting future work which we do not have enough time to explore (Section 8).

2 Related Work

There are considerable work in related area such as reviews analysis, aspect segmentation, user preference study and etc.. Two that have been intensively studied are review aspect identification and rating determination for identified aspect. *SemEval-2015 Task 12: Aspect Based Sentiment Analysis* (Pontiki et al., 2015) aims to identify opinions that expressed about specific entities (e.g. laptops) and their aspects (e.g. price). For the pre-specified entities and attributes, they label the sentiment by positive, neutral or negative, for every combination of entities and attributes. Their goal is to ultimately provide summaries for all aspects with polarity. In *Fine Granular Aspect Analysis using Latent Structural Models* (Fang and Huang, 2012), the authors propose an approach to determine sentiment polarity for reviews, identify sentence which talks about some certain aspect and finally predict sentiment polarity for such aspect. In another paper, *Aspect Extraction with Automated Prior Knowledge Learning* (Chen et al., 2014), the authors propose an aspect extraction method with automated prior knowledge learning system using existing terms of each aspect and exploit the learned knowledge in extracting more coherent aspects. Furthermore, an intelligent system has been built which could accurately tackle the task of sentiment analysis in e-commerce reviews (Khalil and El-Beltagy, 2016). The system is constructed with several aspect extraction models to obtain corresponding sentiment polarities, but with a limitation that model has to train on a separate classifier for other domains. Nevertheless, none of these provide a method to measure the critical indicator weight that people put on different aspects in this paper.

We also look at a novel MIR model on texts with known aspects rating to predict aspects rating (Pappas and Popescu-Belis, 2014) and what attracts our attention is that they calculate weight on each sentence and further use it to analyze reviewers' attitude towards some aspects. This usage of applying weight to analyze user preference inspires us.

More recent work about user preference on sentence structure of reviews is *Textual Demand Analysis: Detection of Users' Wants and Needs from Opinions* (Kanayama and Nasukawa, 2008). The authors create an unsupervised pattern induction method to detect user latent demands in Japanese. They focus on only one type of linguistic structure ”I want N”, concentrating on ”Noun” to extract customers' need, however their work is limited to Japanese.

Eventually the two most relevant work we find to help to achieve our goal are *Latent Aspect Rating Analysis on Review Text Data: A Rating Regression Approach* (LARA) (Wang et al., 2010) and *Latent Aspect Rating Analysis without Aspect Keyword Supervision* (LARAM) (Wang et al., 2011). LARA includes a boot-strapping aspect segmenting method and LRR model, and LARAM introduces a new topic model to segment aspect and keeps same LRR model, so the biggest difference between these two papers is that LARA uses a supervised algorithm to figure out the aspects of reviews, but LARAM uses a new unsupervised method. The models introduced in these two paper are not only able to identify the latent aspects from a review, but also output ratings and weights for each aspect. By adopting their model, we are able to collect aspect weights for each review for each user, and combined these outputs to decide what are the aspects each user values most. Although LARAM which introduced in the second paper overcomes the limitation that aspect segmentation requires pre-decided keywords for each aspect, we choose to use LARA, due to lack of good code source of LARAM and the difficulties to develop one from scratch with the time limitation.

The common purpose of these two paper is using the output of LRR model to predict aspects rating, however, we are more interested in discovering the usage of another output from LRR model, which is the weight of each aspect. Weight values may indicate the degree of emphasis placed on such aspect by the reviewer and a higher weight probably means more emphasis on the corresponding aspect, we may be able to take the weight as a feature to explore reviewer preference. Later we could try to group people by common characteristics to discover potential variations among groups and analyze the possible reasons behind it. Moreover, this discovered preference may contribute to a more accurate recommendation to help people making selections and help companies (e.g. hotels) to increase profit by improving most needed aspect.

3 Data

Due to the limitation of current available TripAdvisor dataset, we decide to build our own. With the goal of calculating aspect weights which we believe reflect preference for each user based on hotel reviews, we program a `Python` based crawler using `Scrapy`[1] module and gather 60,463 unique users and

[1] `https://scrapy.org`

city	content	hotel_classs	hotel_name	hotel_review_qty	hotel_review_stars	place_type	review_stars	reviewer_location	title	user_name
New York City	Very spacious. Excellent...	4	Q&A Residential Hotel	279 Reviews	4	Hotel	5	Appleton, Wisconsin	Excellent hotel	Intrepid_Travel20

Table 1: A sample data of what we collect from web scrawler.

<User Name>daddyoSouthernCA

<Title>Great hotel, great location.
<Content>The Rex is in a good area, close to good restaurants and neighborhood bars. Half block from the Powell St. cable car, also very close to BART, street car and many major bus routes. Staff is very friendly and attentive. Hotel is clean and efficient. Great bar in the back and Cafe Andre is next door.
<Overall>5.0

<Title>Very good location.
<Content>This is a great old hotel located close to Times Square. The hotel is clean and the bathrooms are updated. There are two good delis very close, one is right next door. Plenty of good bars also. NBC, Rockefeller Center, Times Square, Grand Central Station are also very close. It is very close to several subway stations.
<Overall>3.0

Table 2: Sample JSON file.

763,895 reviews (Each user has at least 2 hotel reviews) from some large cities in the world from TripAdvisor. Although we were unable to get aspect ratings of each review, in addition to the review content, we have user location, review title, overall rating and etc. for further analysis. For all the features that we gathered by this approach, see Table 1.

To meet the input requirement of our model, we later process the data by grouping and writing all reviews of every user into each single JSON file and filter out roughly 10,000 users whose reviews are too short or unrecognizable (e.g. Japanese). In total 49,820 JSON files are created after this process. An example of JSON file can be seen in Table 2.

4 Model

We adapt the Latent Rating Regression Model (LRR) constructed by Professor Hongning Huang from University of Illinois at Urbana-Champaign. Unlike his research which aim to use the model to predict ratings that a user might to give to a specific domain (e.g. hotel), instead we are interested in take another part of output from his model, which is weight on each aspect, and explore the possible usage of it.

LRR model assumes that the overall rating of an individual's review is based on the weighted summation of ratings for given aspects. Obtaining ratings for every interested aspect, we are then able to conduct detailed aspect-based analysis for a hotel based on all reviews about it.

Instead of predicting aspect ratings for hotels, we focus on users. More specifically, we utilize the model to estimate weights of each given aspect of a single reviewer based on all reviews that he or she wrote. We assume that the aspect weights in one user's reviews actually reflect his or her extent of caring about such aspect. For example, if the weight of aspect "Location" dominates all reviews from a single user, we should believe that this reviewer cares more about location when choosing or evaluating hotels.

We now demonstrate how the model is constructed and how we obtained the aspect weights from a review. First we pull all reviews including titles from a single reviewer. Take all history review on hotel of each users as input, first we divide reviews into sentences, and assign aspect label to each sentence based on aspect keyword counting. Four aspects are specified, which are Location, Cleanness, Services and Facilities. Next, we rank words in our vocabulary set using Chi-square score and categorize them into aspects. The Chi-Square score is calculated using the following formula:

$$\chi^2(\omega, A_i) = \frac{C \times (C_1C_4 - C_2C_3)^2}{(C_1+C_3)\times(C_2+C_4)\times(C_1+C_2)\times(C_3+C_4)}$$

where C_1 is the number of times a word occurs in the sentences belonging to aspect A_i; C_2 is the number of times a word occur in the sentences does not belonging to aspect A_i; C_3 is the number of sentences of aspect A_i that do not contain this word; C_4 is the number of sentences that neither belong to aspect A_i, nor contain this word and C is the total number of word occurrences. Repeating such procedure, we can finally obtain the $k\dot{n}$ feature matrix W_d , which is the word frequency in each aspect.

LRR model assumes that the aspect rating for a sentence is the summation of word frequency multiplied by sentiment polarity score. The aspect rating for a reviewer then is further assumed to be the weighted sum of rating for each sentence and the overall rating is a Gaussian distribution.

user_10416,0.452,0.181,0.221,0.145
user_10418,0.171,0.343,0.106,0.380
user_10419,0.218,0.323,0.117,0.342
user_10423,0.247,0.339,0.104,0.310
user_10424,0.239,0.316,0.119,0.326
user_10425,0.234,0.261,0.122,0.383
user_10426,0.145,0.279,0.116,0.460
user_10429,0.155,0.457,0.105,0.284

Table 3: Sample output of LRR.

$$\tau_d \sim N(\sum_{i=1}^{k} \alpha_{di} \sum_{j=1}^{n} \beta_{ij} W_{dij}, \delta^2)$$

Since weight for each aspect is not independent, we make another assumption that weight has a prior distribution as multivariate Gaussian distribution.

$$\alpha_d \sim N(\mu, \sum)$$

Finally, we obtain a model as

$$P(r|d) = P(r_d|\mu, N(\mu, \sum, \delta^2, \beta, W_d)$$
$$= \int p(\alpha_d|\mu, \sum)p(r_d| \sum_{i=1}^{k} \alpha_{di} \sum_{j=1}^{n} \beta_{dij} W_{dij}, \delta^2) d\alpha_d$$

To obtain the weights, we apply "maximum a posteriori (MAP) estimation method" and derived the estimated weight for the four aspects on a single reviewer.

5 Experiment

Experiment follows below steps: 1) select aspects and keywords for each aspect; 2) run Bootstrapping method; 3) run LRR model; 4) post-processing of output file for further analysis.

5.1 Keywords Selection

In order to collect key words which are used for matching aspects, we manually select a few seed words for each pre-defined aspect and use them as input to the aspect segmentation algorithm described above, where we set $p = 4$ and $i = 10$ in our experiments.

- **Facility**: bedroom, bathroom, space, view.
- **Cleanliness**: clean, dirty.
- **Service**: service, manager, staff.
- **Location**: Location, locate, distance, airport, close.

akron ohio	['Akron, Summit County, Ohio, United States of America', 41.0830643, -81.5184853]
akron, oh	['Akron, Summit County, Ohio, United States of America', 41.0830643, -81.5184853]
akron, ohio	['Akron, Summit County, Ohio, United States of America', 41.0830643, -81.5184853]
akron,oh	['Akron, Summit County, Ohio, United States of America', 41.0830643, -81.5184853]

Table 4: Sample of high quality address information obtained through GeoPy.

	Accuracy
Same Most valued aspect	62%
Same Least valued aspect	71%

Table 5: Accuracy

5.2 Bootstrapping and LRR

Given the JSON files we generated as input, we then perform simple pre-processing on these reviews as part of bootstrapping approach: 1) aggregate all reviews from one user; 2) convert words into lower cases; 3) remove punctuations, pre-defined stop words, and terms occurring less than 5 times in the corpus; 4) stemming each word to its root with Porter Stemmer. Bootstrapping step output a file that contains 4 dimension vectors for each user and will be use as input for LRR step. During this process, we also discard some sentences that fail to be associated with any aspect. Finally 35,868 users remain.

After Bootstrapping we load the vector file and run LRR model to calculate aspect weight for each reviewer. An sample of how the output looks like can be seen at Table 3. In this table, we can see that each line start with `user_[userindex]` which followed by weight on four aspects.

5.3 Data Post-Processing

There are few problems still pending at this moment. First we need to combine this weight result with more detailed information of each user. To achieve this, we right join the original file (see Table 1) with this weight output on user index. Second, the user location which we scrawled from TripAdvisor lacks a uniform format which leads to much ambiguity (e.g. "akron ohio", "akron, oh", "akron, ohio" and "akron,oh") and inconsistency (e.g. city versus country level location). In order to get location information of a high quality, we use `GeoPy`[2] package to call Google Geocoding API (V3) and get a very

[2] `https://geopy.readthedocs.io/en/1.10.0/`

high quality of address with latitude and longitude (see Table 4) in the end.

6 Evaluation

We believe this model can be an effective way of finding user preference by extracting aspect from reviews and calculating weights. Since it is impossible to calculate aspect weights manually, we conduct an approach to evaluate our method: comparing only the aspects which have highest and lowest weights that computed by model with those gained from manual annotation.

First, we select 1,000 reviews from 100 users (each user posted 10 reviews) for manual annotation. Second, we rate the aspect weights of every review roughly. Notice that the sum of weights in each review should be 1. Third, given 10 reviews of a user with manually rated aspect weights, we compute the average weights of each aspect to get user preference. Again, to simplify the comparison, we only considered the aspects with highest and the lowest weights. Finally, we compare them with values computed by model for same users. Here we define accuracy as the percentage of users who have the same aspect preference in manual weights and model weights.

By looking at accuracy from result of Evaluation (see Table 5), we conclude that this method is much better than random guessing (25%) and it is indeed an effective automatic approach. It is also interesting to notice that model performs better in finding the least valued aspects than most. The reason probably is it is much easier to give a lower weight with few key words matched up than add weights to aspects which have many key words associated and then to decide which one is the most valued.

7 Analysis

Figure 1 - 4 exhibit the distribution of each aspect weight. Apart from cleanness exhibits which shows a skewness on the right side, all other three distributions are approximately normal, which meets our assumption of normality.

Figure 5 indicates that in general location (0.3307) and service (0.3137) gain the highest weight values. Since our model is calculated weight by taking frequency and sentiment polarity of word into account, the higher the weight is the more "preference" people put on the corresponding aspect. In this case, people value location and service more than the other two. In other words, this graph implies that when booking a hotel or living in a hotel, most people would take these two aspects into their considerations, so on the hotels point of view, they probably will gain higher feedback, through improving location (e.g. better transportations) and services.

We further analyze user preference by their geo information to see if there exists geological characteristics in user preference. Figure 6 shows the weight comparison

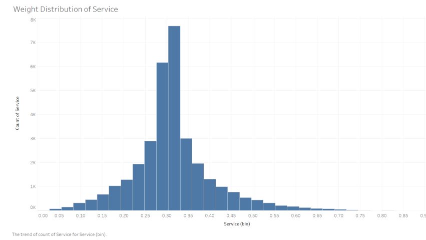

Figure 1: Service distribution.

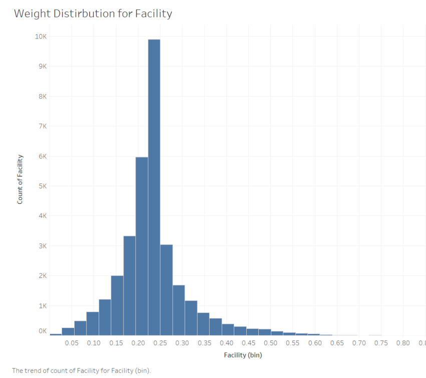

Figure 2: Facility distribution.

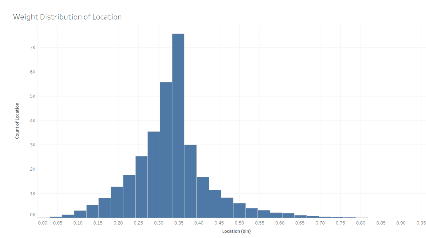

Figure 3: Location distirbution.

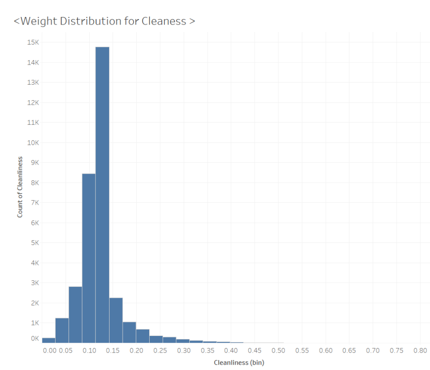

Figure 4: Cleanliness distibution.

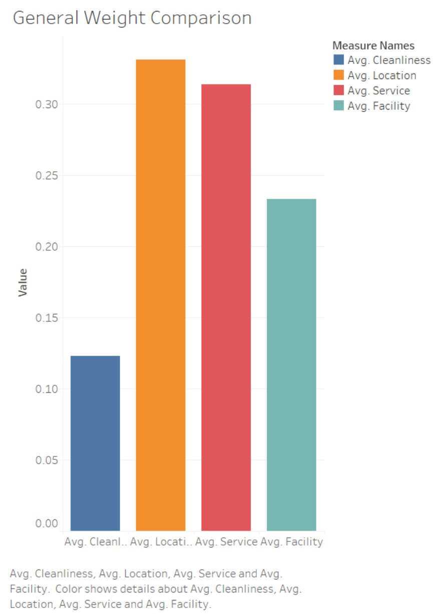

Figure 5: General weight comparison.

among reviewers from different countries. We can see clearly from the picture that aspect weight varies across countries, however, by looking into detail of data, we find that the most significant differences are mainly due to a small sample size. By involving more data and group users in a smaller level, such as state and city level instead of country level in the future, we could obtain a better understanding on the potential location variations among different regions.

Next we only look at countries with a relatively large sample size (see Figure 7) . We see that not surprisingly most of them value location and service most, however, people from Hong Kong value location more than other aspects while British value service as much as location. It is interesting that, by identify and draw data for some big cities , people from New York (see Figure 8) values service more than location, which is different from our general conclusion that people value location most.

8 Future Work

In the experiment, we notice that different seed words may lead to different results, we want to play with different keywords and compare the outputs and further adopt a topic model to gain aspects automatically without giving keywords. Since some features in our data are not used, such as hotel class and hotel type, we expect to add them into analysis in the future and may introduce more information such as gender, frequently stayed hotel level, and age. Eventually, we can provide a framework of customized hotel recommendation system.

The bootstrapping algorithm for aspect segmentation

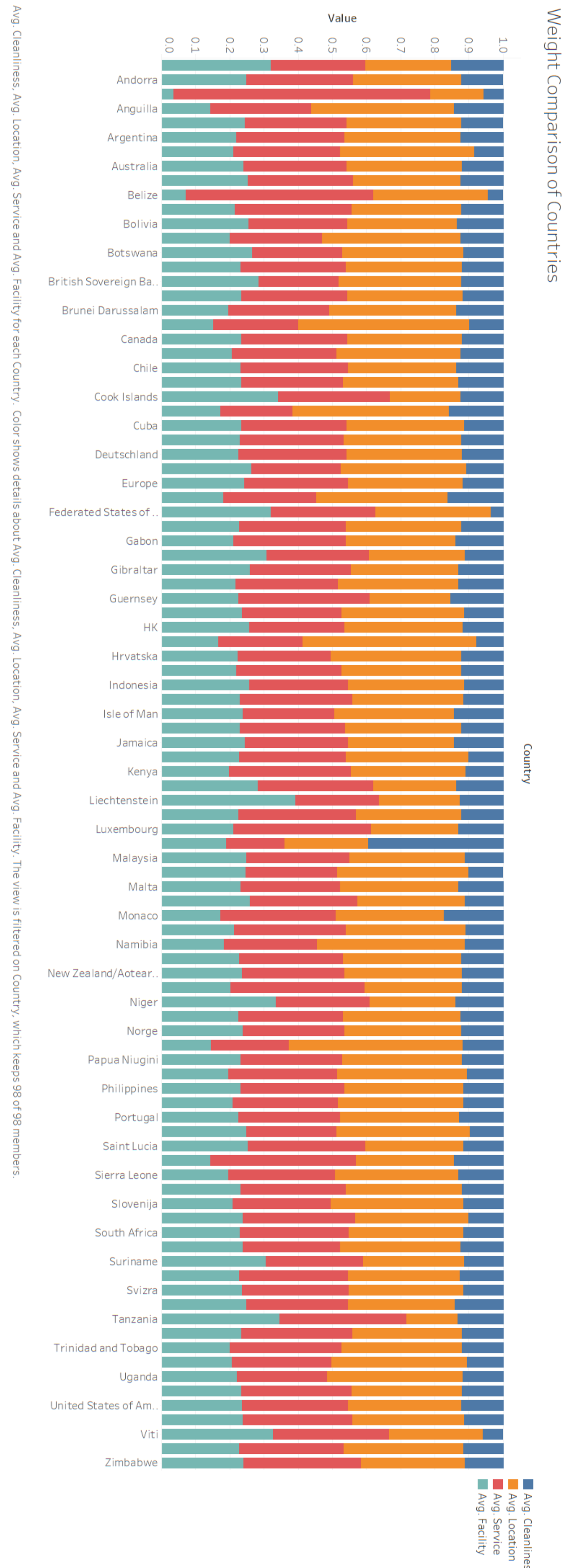

Figure 6: Weight comparison among countries

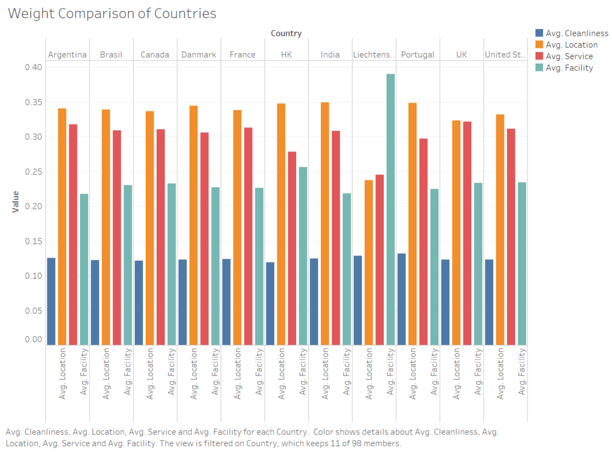

Figure 7: Weight comparison among countries

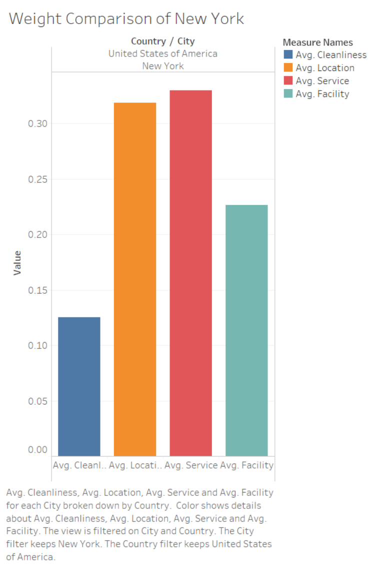

Figure 8: Weight comparison for New York

we adapt is a broadly generalized method that could be used in other domains (e.g. restaurant), so we can apply this approach to other domain other than hotel. For example, Amazon could recommend a particular product to users based on analysing their former reviews and extracting the most concerning aspects for each user.

Moreover, we may apply this model to social topics. For instance, by joining user preference with his/her cultural and financial background, we can explore topics such as whether users from developed countries prefer restaurants with higher environment quality than other countries in the world or not.

Acknowledgments

This is the final paper for COMSE6998 and team members contributions are listed as below:

Hanlin Wu: Web scrawler to gather all original data; Results evaluation; Model interpretation; Part of data cleaning and processing; First two part of Experiment Section and Evaluation Section in the paper.

Xi Lu: Model implementation; Part of data cleaning and processing; Data Section and last part in Experiment Section in the paper; Integrating all the contents including other team members into final Latex version of paper.

Sicong Fang: Helped to run scripts to obtain data; Model interpretation; Model Introduction & Rationale, Result Visualization & Analysis, and part of Related Work in the paper.

Yunsi Zhang: Giving help in running scripts to obtain data; Created some visualization and analysis in presentation; Future Work Section and part of Related Work in the paper.

References

Zhiyuan Chen, Arjun Mukherjee, and Bing Liu. 2014. Aspect extraction with automated prior knowledge learning. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, pages 347–358*.

Lei Fang and Minlie Huang. 2012. Fine granular aspect analysis using latent structural models. *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics, pages 333–337*.

Hiroshi Kanayama and Tetsuya Nasukawa. 2008. Textual demand analysis: Detection of users' wants and needs from opinions. *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008), pages 409–416*.

Talaat Khalil and Samhaa R. El-Beltagy. 2016. Niletmrg at semeval-2016 task 5: Deep convolutional neural networks for aspect category and sentiment extraction. *Proceedings of SemEval-2016, pages 271–276*.

Nikolaos Pappas and Andrei Popescu-Belis. 2014. Explaining the stars: Weighted multiple-instance learning foraspect-based sentiment analysis. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 455–466*.

Maria Pontiki, Dimitrios Galanis, Haris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. Semeval-2015 task 12: Aspect based sentiment analysis. *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), pages 486–495*.

Hongning Wang, Yue Lu, and Chengxiang Zhai. 2010. Latent aspect rating analysis on review text data: A rating regression approach. *KDD'10*.

Hongning Wang, Yue Lu, and Chengxiang Zhai. 2011. Latent aspect rating analysis without aspect keyword supervision. *KDD'11*.