Opinion Mining from Customers' Reviews

XIE Yuxiao

Abstract

Opinion mining is the computational work to study people's opinions, including aspect extraction and sentiment analysis. In this project, we implemented a novel RBM model to extract review aspects and sentiment polarities in an unsupervised setting. Through introducing a heterogeneous structure into the hidden layer of a normal RBM and combining informative priors into the model, this model is capable for extracting aspect and sentiment simultaneously. The experimental results showed that our model performs better than both normal RBM and LDA.

1. Introduction

The growing availability and popularity of opinion-rich resources such as online review sites are influencing the way people making decisions, especially when they hesitate of whether to buy a product or not. People now get used to browsing relative information on the internet before making decisions, and leaving some comments after their purchases. However, the social networks nowadays are awash with daunting amount of trivial or even garbage information, making it difficult to obtain useful conclusions from the information floods. Therefore, how to utilize these reviews and comments effectively has become an important NLP research problem.

Opinion mining is the computational study of people's opinions, appraisals, attitudes, and emotions toward entities, individuals, issues, events, topics and their attributes (*B. Liu, A Survey of Opinion Mining and Sentiment Analysis, 2012*). In this context, aspect-based opinion mining, also known as feature-based opinion mining, aims at extracting and summarizing salient aspects of entities and determining relevant sentiment polarities from reviews (*Hu and Liu, 2004*). Not only will this kind of analysis make the formidable amount of information on the internet more accessible to users, but also let the vendors and merchants discern customers' predilections or aversions to improve their products or services.

In this project, a novel model based on Restricted Boltzmann Machines (RBMs) with prior text information was applied to extract aspect and sentiment related words. The model is unsupervised so there are no requirements for labels during the training process, which contributes to a more flexible and efficient model. Compared to LDA (Latent Dirichlet Allocation) or conventional RBMs, this model achieves the following distinct advantages:

- An overall better accuracy. This model achieved a 9.18 % accuracy improvement compared to LDA, and a 12.96% improvement compared to conventional RBMs.
- A more interpretable and informative model. The optimal weight of the model can be interpreted as a specific word feature towards aspects and its sentiment, while LDA cannot offer such features.
- Capability to extract aspect and sentiment simultaneously. Unlike normal RBMs, the hidden layer of this model has a heterogeneous structure, which represents aspects, sentiments and background information respectively.

2. Methodology

The basic idea behind the model is to utilize a RBM model for latent topic modelling and treat some prior information extracted from the text to regularize the optimal weights of the model. This eclectic RBM model combined normal RBMs with LDA, TF-IDF and sentiment scores, and showed a better accuracy compared to both LDA and normal RBMs.

2.1 RBMs

A restricted Boltzmann machine (RBM) is a generative stochastic artificial neural network that can learn a probability distribution over its set of inputs. They are shallow, two-layer neural nets where the first layer is called the visible (input) layer, and the second is the hidden layer. In a basic RBM model for topic modeling, the visible layer consists of a softmax over discrete visible units for words in the text, while the hidden layer captures its topics.

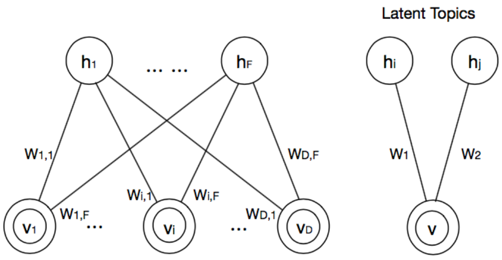

Figure 1: RBM Schema

Conventional RBMs have obvious drawbacks for they cannot distinguish aspects and sentiment in reviews, therefore they may not perform well in aspect extraction work. In this project, a more powerful model was applied to extract aspects and sentiment simultaneously. Unlike standard RBMs, there are three types of hidden units in this model, representing aspect, sentiment, and background, respectively. The optimal weight matrix of this RBM model can exactly reflect individual word features toward aspects and sentiment.

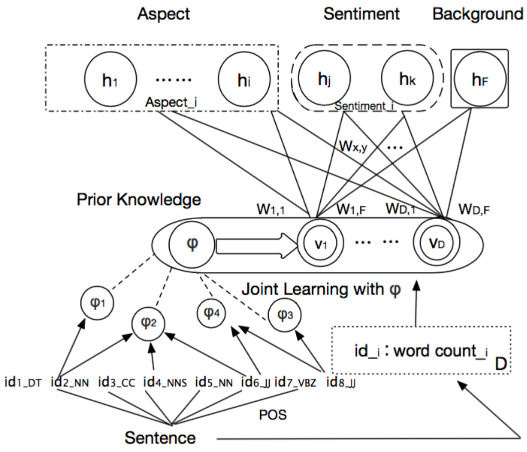

Figure 2: Sentiment-Aspect Extraction Model

2.2 Informative priors

Four different kinds of priors were applied in this model. Three of them are relevant to POS tagging, and the last one utilized LDA's information.

a. POS tagging

The process of classifying words into their parts of speech and labeling them accordingly is known as part-of-speech (POS) tagging. Since the aspects words are often nouns and sentiment words are often adjectives, we will first apply POS tagging to separate these words. For each noun word, we will calculate its TFIDF and rank them according to its TFIDF, denoted by $p_{A,Vk}$, under the assume that more frequent nouns in the documents are more possible to be the aspect words.

For all adjective words, if the words are also included in the online sentiment resource SentiWordNet, we assign prior probability $p_{s,vk}$ to suggest that these words are generally recognized as sentiment words. For each sentiment word in SentiWordNet, we will also assign the negative score and positive score, denoted by $p_{sj,vk}$, respectively to indicate the probability of being a negative word or positive word. For example, the negative score of word "terrible" would be higher than its positive score.

b. LDA

In natural language processing, latent Dirichlet allocation (LDA) is a generative statistical model that allows sets of observations to be explained by unobserved groups that explain why some parts of the data are similar. It is a useful model for topic modelling, and it posits that each document is a mixture of a small number of topics and that each word's creation is attributable to one of the document's topics. In this project, we will use LDA’s result to indicate the probability of a known aspect word belonging to a specific aspect, denoted as $\mathrm{p}_{\mathrm{Aj,vk}}$. For example, word “salad” can be treated as an aspect word related to aspect *food.*

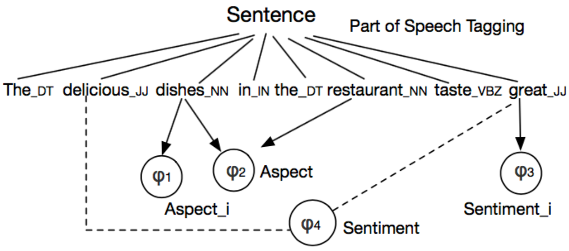
Sentence
Part of Speech Tagging
The_DT delicious_JJ dishes_NN in_IN the_DT restaurant_NN taste_VBZ great_JJ
φ1
Aspect_i
φ2 Aspect
φ4 Sentiment
φ3
Sentiment_i

Figure 3: Prior Feature Extraction

2.3 Training

The log-likelihood objective function with four terms of regularization corresponding to the four kinds of priors of this model was defined as follows:

$$
\begin{aligned}
\ln L_S = \ln \prod_{i=1}^{n_s} P(\mathbf{v}^i) - \sum_{i=1}^{n_s} \Bigg[\\
\lambda_1 \ln \prod_{j=1}^{F_1-1} \prod_{k\in R_1} \left[P(h_j = 1 \mid \hat{v}^k) - p_{A_j,v_k} \right]^2 \\
+ \lambda_2 \ln \prod_{k\in R_2} \left[\sum_{j=1}^{F_1} P(h_j = 1 \mid \hat{v}^k) - p_{A,v_k} \right]^2 \\
+ \lambda_3 \ln \prod_{j=F_2}^{F_2+1} \prod_{k\in R_3} \left[P(h_j = 1 \mid \hat{v}^k) - p_{S_j,v_k} \right]^2 \\
+ \lambda_4 \ln \prod_{k\in R_4} \left[\sum_{j=F_2}^{F_2+1} P(h_j = 1 \mid \hat{v}^k) - p_{S,v_k} \right]^2 \Bigg]
\end{aligned}
$$

where v^{i} is the training object (document) with form: $\mathbf{v}^i = (v_1^i, v_2^i, \ldots, v_K^i)^D$ and K is the dictionary size, and D is the document length. $\hat{v}^k = \sum_{i=1}^{D} v_i^k$ stands for the count for k-th

$P(h_j = 1 \mid \hat{v}^k)$

word in a document, and is the probability of this word belongs to a given hidden unit.

Optimization technique SGD was used to maximize the objective function. We used Contrastive Divergence (CD-m) algorithm to train the RBM. The pseudo code for training a RBM is shown below:

Algorithm 1

```
RBMupdate($x_1, \epsilon, W, b, c$)
*This is the RBM update procedure for binomial units. It can easily adapted to other types of units.*
$x_1$ is a sample from the training distribution for the RBM
$\epsilon$ is a learning rate for the stochastic gradient descent in Contrastive Divergence
$W$ is the RBM weight matrix, of dimension (number of hidden units, number of inputs)
$b$ is the RBM biases vector for hidden units
$c$ is the RBM biases vector for input units

  **for all** hidden units $i$ **do**
    • compute $Q(\mathbf{h}_{1i} = 1|x_1)$ (for binomial units, sigm$(b_i + \sum_j W_{ij} x_{1j})$)
    • sample $\mathbf{h}_{1i}$ from $Q(\mathbf{h}_{1i}|x_1)$
  **end for**
  **for all** visible units $j$ **do**
    • compute $P(x_{2j} = 1|\mathbf{h}_1)$ (for binomial units, sigm$(c_j + \sum_i W_{ij} \mathbf{h}_{1i})$)
    • sample $x_{2j}$ from $P(x_{2j} = 1|\mathbf{h}_1)$
  **end for**
  **for all** hidden units $i$ **do**
    • compute $Q(\mathbf{h}_{2i} = 1|x_2)$ (for binomial units, sigm$(b_i + \sum_j W_{ij} x_{2j})$)
  **end for**
  • $W \leftarrow W + \epsilon(\mathbf{h}_1 x_1' - Q(\mathbf{h}_{2\cdot} = 1|x_2) x_2')$
  • $b \leftarrow b + \epsilon(\mathbf{h}_1 - Q(\mathbf{h}_{2\cdot} = 1|x_2))$
  • $c \leftarrow c + \epsilon(x_1 - x_2)$
```

In over model, despite updating weights only according to the CD-m's results, we will also update the weights according to its 4 priors. In each step, the weight will be updated by ΔW:

$$\lambda \left[P(h_j = 1|\mathbf{v}^{(0)}) v_k^{(0)} - P(h_j = 1|\mathbf{v}^{(\mathrm{cdm})}) v_k^{(\mathrm{cdm})} \right]$$
$$- \lambda_1 \sum_{j=1}^{F_1 - 1} \sum_{k \in R_1} \frac{2 G_j \widehat{v}^k}{(1 + G_j)^2 (\frac{1}{1+G_j} - p_{A_j, v_k})}$$
$$- \lambda_2 \sum_{k \in R_2} \frac{2 \widehat{v}^k}{\sum_{j=1}^{F_1} \frac{1}{(1+G_j)} - p_{A, v_k}} \sum_{j=1}^{F_1} \frac{G_j}{(1 + G_j)^2}$$
$$- \lambda_3 \sum_{j=F_2}^{F_2 + 1} \sum_{k \in R_3} \frac{2 G_j \widehat{v}^k}{(1 + G_j)^2 (\frac{1}{1+G_j} - p_{S_j, v_k})}$$
$$- \lambda_4 \sum_{k \in R_4} \frac{2 \widehat{v}^k}{\sum_{j=F_2}^{F_2+1} \frac{1}{(1+G_j)} - p_{S, v_k}} \sum_{j=F_2}^{F_2 + 1} \frac{G_j}{(1 + G_j)^2},$$

Here $G_j = e^{-(a_j + W_j^k \widehat{v}^k)}$ and $\mathrm{v}^{\mathrm{cdm}}$ is the result of CD-m steps.

3. Experiments

3.1 Dataset and preprocessing

We used a widely-used restaurant review dataset in the training process. Documents in this dataset are annotated with one or more labels from a gold standard label set S = {*Food, Staff, Ambience, Price, Anecdote, Miscellaneous*}. Following the previous studies, we select reviews with less than 50 sentences and remove stop words. Then we used lemmatizer tools to group together the inflected forms of a word so they can be analyzed as a single item. After preprocessing, there are totally 52624 reviews and 179139 tokens in the corpus.

To transform reviews into training objects, we used scikit-learn to convert the text documents to a matrix of token counts. We chose words with top 5000 frequencies to form the dictionary, since it will lead to a very huge but sparse matrix if not doing so. Each document form a training object, and each line of the document is an input for the model.

3.2 Fetching priors

We used POS tagging tools offered in NLTK package to pick out the nouns and adjectives. For all nouns, we will compute its TF in the corpus and use another dataset to compute its IDF. The dataset we chose is Reuters dataset. Then we will compute these word's TFIDF and record them as $p_{A,Vk}$. For example, the value of word "salad" in $p_{A,Vk}$ is 0.3648, indicating its general probability of being an aspect word.

To get a more fine-grained probability of each word belongs to specific aspects, we applied LDA on the dataset and used the result to indicate this probability. For example, the value that word "salad" belongs to aspect *food* is 0.1986, and other values are zero, which means that this word is likely to be a word belongs to aspect *food.*

For each adjective, we used SentiWordNet to get its sentiment score. The SentiWordNet interface in NLTK will give each sentiment word a positive score, a negative score and an objective score. We used the positive score and negative score to form $p_{Sj,Vk}$ and used 1-obj_score to indicate $p_{S,Vk}$. For example, the positive score of word "great" is 1 and its negative score is 0.

3.3 Training and evaluation

We used CD-m algorithm to train the model on mini-batches. An import task in the training process is to tune the hyper-parameters such as learning rate and prior parameters. To measure how well the predictions match the true label and optimize the hyper-parameters, we computed the Precision, Recall, and F1 scores. Precision (P) is defined as the number of true positives (TP) over the number of true positives plus the number of false positives (FP):

$$\mathrm{P} = \frac{TP}{TP + FP}$$

Recall (R) is defined as the number of true positives (TP) over the number of true positives plus the number of false negatives (FN):

$$\mathrm{R} = \frac{TP}{TP + FN}$$

These quantities are also related to the (F1) score, which is defined as the harmonic mean of precision and recall:

$$\mathrm{F1} = \frac{2PR}{P + R}$$

To avoid ambiguity, we used sentences with only one label and choose three aspects {*Food, Staff, Ambience*} to perform evaluation.

4. Results

The final prediction result can be shown as table below:

Aspect	Precision	Recall	F1
Food	0.6765	0.7657	0.7183
Staff	0.5869	0.6009	0.5938
Ambience	0.3965	0.4388	0.4166

Table 1 Accuracy of the aspect extraction task

To compare the accuracy of our model with other models, we also applied training on LDA and normal RBMs, and the comparison can be shown in the following table:

Aspect	RBM	LDA	Our model
Food	0.6359	0.6579	0.7183
Staff	0.4028	0.5802	0.5938
Ambience	0.2065	0.2546	0.4166

Table 2 F1 scores of different models

We also listed the representative aspect words extracted by these models to show their differences:

Aspect	RBM	LDA	Our model

Food	Sauce, duck. appetizer, tuna, shrimp, filet, Excellent, Steak, Good, try, banana, delicious, portion, pepper, grilled, chicken, crust, pork, entree, lamb, small, bland, cheese, salad, ice	Food, good, try, pizza, fresh, excellent, delicious, dessert, sauce, appetizer, salad, steak, small, entree, sushi, fish, tasty, cheese	Sauce, pork, salad, flavour, dessert, outstanding, duck, meat, delicious, appetizer, cheese, bean, grilled, spinach, stick, entrée, filet, dish, pepper, excellent, small, amaze, chicken, tomato
Staff	Service, staff, worth, friendly, help, make, waiter, long, trip, rude, server, attentive, excellent, accommodate, courteous, price, way, come, host, waitress	service, staff, waiter, friendly, ask, tell, waitress, attentive, rude, server, come, leave, manager	Service, staff, friendly, bad, attentive, help, accommodate, bartender, waiter, server, polite, nice, attitude, waitress, professional, quick, rude
Ambience	Loud, music, room, intimate, delicious, décor, absolutely, cozy, want, trendy, dessert, village, try, romantic	Place, great, experience, nice, recommend, wine, new, enjoy, wonderful, décor, ambience	Place, music, ambience, room, loud, portion, hip, try, look, village, atmosphere, romantic, space, cozy, area, east, casual, time, square, noisy

Table 3 Representative words extracted by different models

From the tables above, we could see our model outperforms than other aspect extraction models.

5. Conclusions

In this project, we implemented a novel RBM model to extract review aspects and sentiment polarities in an unsupervised setting. Our approach modifies the standard RBM model by introducing a heterogeneous structure into the hidden layer and incorporating informative priors into the model. The experimental results showed that our model performs better than normal RBM and LDA. We also found that the quality of prior information will influence the final accuracy, so we could try different prior information extracted from different models and integrate them into our model for better results.

6. References

[1] *Sentiment-Aspect Extraction based on Restricted Boltzmann Machines.* L Wang, K Liu, Z Cao, J Zhao, G de Melo, ACL (1), 616-625, 2015

[2] *A Practical Guide to Training Restricted Boltzmann Machines*. Hinton G E. Momentum, 9(1), 599-619, 2012

[3] *Learning Deep Architectures for AI.* Yoshua Bengio. Technical Report 1312, 26-27, 2009