Leveraging Structural and Semantic Correspondence for Attribute-Oriented Aspect Sentiment Discovery

Zhe Zhang
Watson Group
IBM Corporation
Research Triangle Park, NC 27703-9141
zhangzhe@us.ibm.com

Munindar P. Singh
Department of Computer Science
North Carolina State University
Raleigh, NC 27695-8206
singh@ncsu.edu

Abstract
Opinionated text often involves attributes such as authorship and location that influence the sentiments expressed for different aspects. We posit that structural and semantic correspondence is both prevalent in opinionated text, especially when associated with attributes, and crucial in accurately revealing its latent aspect and sentiment structure. However, it is not recognized by existing approaches.

We propose Trait, an unsupervised probabilistic model that discovers aspects and sentiments from text and associates them with different attributes. To this end, Trait infers and leverages structural and semantic correspondence using a Markov Random Field. We show empirically that by incorporating attributes explicitly Trait significantly outperforms state-of-the-art baselines both by generating attribute profiles that accord with our intuitions, as shown via visualization, and yielding topics of greater semantic cohesion.

1 Introduction
Opinionated text is often associated with different attributes—latent variables that serve as reference frames relative to which the underlying aspects and sentiments are expressed. Common attributes in consumer reviews include author type (e.g., business traveler or tourist for hotel reviews; location for reviews of music (McDermott et al., 2016); culture on reviews of food (Bahauddin and Shaarani, 2015)). Whereas current approaches consider attributes in a one-off manner in each application, we posit that attributes can be systematically extracted if we can properly capture the structural and semantic correspondence that is prevalent in opinionated text. We claim that ignoring attributes may lead to biased inference on aspects and sentiments. As evidence, we demonstrate an approach that outperforms the state of the art and yields intuitive and cohesive topics.

We propose Trait, a general model for discovering attribute-oriented aspects and sentiments from text. By incorporating attributes, Trait automatically generates profiles that describe attributes in terms of sentiments and aspects. To leverage structural and semantic correspondence, Trait applies a Markov Random Field as regularization over sentences during inference. We evaluate Trait on four datasets from two domains and consider three attributes. Trait successfully discovers aspects associated with sentiments; the generated word clusters are more cohesive than the state-of-the-art baselines; the generated attribute profiles are well correlated with ground truth.

Motivating Example. Figure 1 presents two hotel reviews from TripAdvisor. We manually assign aspect labels for sentences and calculate pairwise cosine similarity between sentences using sentence embedding from a pretrained sentence encoding model (Cer et al., 2018).

Review A and Review B, which mention aspects Room, Location, and Type, exhibit structural and semantic correspondence. We posit that the correspondence of Location and Type is a result of the attribute value, Las Vegas, common to the two reviews. Location is a crucial aspect for hotels in Las Vegas. On a randomly selected set of 5,000 hotel reviews for Las Vegas, we observe that 4,281 sentences from 2,624 reviews mention the location “Strip.” Using a similarity threshold of 0.6, we obtain 1,929 sentences similar to sentence A3 from 1,519 reviews, including sentence B3 in Review B. We obtain 133 sentences similar to sentence B4 from 128 reviews including A4 in Review A. Figure 1 shows some of these sentences. Likewise, using authorship as an attribute, we observe that users stick to their writing styles. For example, in hotel reviews, some users describe the condition of a room and others share travel tips.
Contributions and Novelty. Our contributions include: (1) a general model that generates attribute profiles associating aspects and sentiments with attributes in text; (2) empirical results demonstrating the benefit of incorporating attributes on a model’s quality; and (3) empirical results demonstrating generalizability using diverse attributes and the quality of the generated attribute profiles.

Trait’s novelty lies in its ability to accommodate attributes. First, it is general across attributes as opposed to being limited to predefined attributes. Second, the handling of attributes means that Trait avoids overfitting to the more prevalent attributes in a dataset. That is, Trait can learn a more refined conditional probability distribution that incorporates specific attributes than otherwise possible. Ignoring the observable attribute variables would relax the constraints on the distribution, meaning that the learned approximate distribution would be biased toward the majority attribute.

Summary of Findings. We demonstrate that incorporating attributes into generative models provides a superior, more refined representation of opinionated text. The resulting model generates topics with high semantic cohesion. We show that Markov Random Field can be used for effectively capturing structural and semantic correspondence.

2 Related Work

Generative probabilistic modeling has been widely applied for unsupervised text analysis. Given the observed variables, e.g., tokens in documents, a generative probabilistic model defines a set of dependencies between hidden and observed variables that encodes statistical assumptions underlying the data. Latent Dirichlet Allocation (LDA) (Blei et al., 2003), a well-known topic model, represents a document as a mixture of topics, each topic being a multinomial distribution over words. The learning process approximates the topic and word distributions based on their co-occurrence in documents.

Many efforts guide the topics learned by incorporating additional information. Rosen-Zvi et al.’s (2004) Author Topic model (AT) captures authorship by building a topic distribution for each author. When generating a word in a document, AT conditions the probability of topic assignment on the author of the document. Kim et al.’s (2012) model captures entities mentioned in documents and models the probability of generating a word as conditioned on both entity and topic. Diao and Jiang (2013) jointly model topics, events, and users on Twitter. Trait goes beyond these models by incorporating sentiments and attributes in a flexible way, which eliminates the model’s dependency on specific attribute types.

Several probabilistic models tackle opinionated text. Titov and McDonald (2008b) handle global and local topics in documents. JST (Lin et al., 2012) and ASUM (Jo and Oh, 2011) model a review via multinomial distributions of topics and sentiments used to condition the probability of generating words. Kim et al.’s (2013) extend ASUM’s probabilistic model to discover a hierarchical structure of aspect-based sentiments. Wang et al.’s (2016) topic model discovers aspect, sentiment, and both general and aspect-specific opinion words. Whereas these models identify aspects and sentiments, they disregard attribute information.

Titov and McDonald (2008a) discover top-
ics using aspect ratings provided by reviewers. Mukherjee et al.’s (2014) JAST considers authors during aspect and sentiment discovery. Poddar et al.’s (2017) AATS jointly considers author, aspect, sentiment, and the nonrepetitive generation of aspect sequences via a Bernoulli process. Zhang and Singh (2018)’s model jointly captures aspect, sentiment, author, and discourse relations. Trait is novel in that, unlike the above models, it is not tied to a specific attribute.

3 Model and Inference

We now introduce Trait’s model and inference mechanism.

3.1 Sentence Embeddings

Measuring semantic similarity between sentences is integral to capturing the structural and semantic correspondence among reviews: high similarity indicates a high degree of correspondence. Cer et al. (2018) propose a pretrained sentence encoding model, Universal Sentence Encoder (USE). USE is based on Vaswani et al.’s (2017) attention-based neural network. Perone et al. (2018) show USE yields the best results among sentence embedding techniques on semantic relatedness and textual similarity tasks. Trait adopts USE to generate sentence embeddings and cosine similarity to measure semantic similarity between sentences.

3.2 Structural and Semantic Correspondence

A Markov Random Field (MRF) defines a joint probability distribution over a set of variables given the dependencies based on an undirected graph. The joint distribution is a factorized product of potential functions.

To capture structural and semantic correspondence, Trait defines an MRF over latent aspects of sentences. Given a set of reviews $D_a$ sharing a common attribute $a$, for sentence $l$ in $D_a$, Trait creates its corresponding sentence set $L$ by adding sentence $l_i$ in $D_a$ if the semantic similarity between sentence $l_i$ and $l$ is larger than a preset threshold $\rho$. For each pair of $l$ and $l_i$, Trait creates an undirected edge between the aspects associated with the two sentences $(t_{l}, t_{l_i})$. To promote $l$ and sentences in $L$ having a high probability of associating with the same aspect, Trait defines a binary edge potential, $\exp\{\mathcal{I}(t_{l}, t_{l_i})\}$, where $\mathcal{I}(\cdot)$ is an indicator function. This binary potential produces a large value if the two sentences have the same aspect and a small value otherwise. Given attribute $a$, sentiment $s$, and a document consisting of $N$ sentences, Trait computes the joint probability of aspect assignments of sentences as:

$$p(t|\psi_{s,a}, \lambda) = \prod_{l} p(t_l|\psi_{s,a}) \exp\left\{\lambda \frac{\sum_{(t_l, t_{l_i}) \in \mathcal{E}_l} \mathcal{I}(t_l = t_{l_i})}{|\mathcal{E}_l|}\right\},$$

where $\psi_{s,a}$ is the aspect distribution given sentiment $s$ and attribute $a$; parameter $\lambda$ controls the reinforcing effects of correspondence regularization; and $\mathcal{E}_l$ is the set of undirected edges for $l$.

3.3 Generative Process

To capture the desired associations, given an attribute type, Trait generates a mixture over sentiments and aspects for each attribute value. Trait assumes that reviews are mixtures of sentiments and considers sentences the basic unit for a sentiment-aspect pair.

Figure 2 shows Trait’s model. Hyperparameter $\alpha$ is the Dirichlet $(\text{Dir}(\cdot))$ prior of the word distribution $\phi$; $\beta$ is the Dirichlet prior of the sentiment distribution $\theta$; and $\gamma$ is the Dirichlet prior of the aspect distribution $\psi$. Given a set of reviews $D$ associated with a set of attribute values $A$ over a set of aspects $T$ and a set of sentiments $S$, each review contains $M$ sentences and each sentence contains $N$ words. Trait’s generative process is as follows.

First, for each pair of aspect $t$ and sentiment $s$, draw a word distribution $\phi_{t,s} \sim \text{Dir}(\alpha)$. Second, for each attribute value $a$ and each sentiment
draw an aspect distribution \( \psi_{s,a} \sim \text{Dir}(\gamma) \).
Third, given a review \( d \) with attribute \( a \), draw a sentiment distribution \( \theta_d \sim \text{Dir}(\beta) \), and for each sentence in \( d \), (1) choose a sentiment \( s \sim \text{Multinomial}(\theta_d) \); (2) given \( s \), choose an aspect \( t \sim \text{Multinomial}(\psi_{s,a}) \); (3) given \( t \) and \( s \), sample word \( w \sim \text{Multinomial}(\phi_{t,a}) \).

Trait estimates \( p(s, t, w, a) \), the posterior distribution of latent variables, sentiments \( s \), and aspects \( t \), given all words used in reviews involving attribute \( a \). We factor the joint probability of assignments of sentiments, aspects, and words for \( a \):

\[
p(s, t, w| a) = p(w| s, t)p(t| s, a)p(s| a).
\] (2)

### 3.4 Inference
We use collapsed Gibbs sampling (Liu, 1994) for posterior inference. By integrating over \( \Phi = \{\phi_i\}_{i=1}^{S \times T} \), we obtain Equation 2’s first term (Section 4.1 explains \( \alpha \)):

\[
p(w| s, t, \alpha) = \int p(w| s, t, \Phi) p(\Phi| \alpha) d\Phi
\]

\[
= \frac{\Gamma(\sum_{v=1}^{V} \alpha_v)}{\prod_{v=1}^{V} \Gamma(\alpha_v)} \prod_{s=1}^{S} \prod_{t=1}^{T} \Gamma(\sum_{v=1}^{V} (n_{s,t,v} + \alpha_v))
\]

where \( W \) is the vocabulary size; \( n_{s,t,v} \) is the number of occurrences of word \( v \) assigned to sentiment \( s \) and aspect \( t \); and \( \Gamma(\cdot) \) is the Gamma function.

Next, by integrating over \( \Psi_{a} = \{\psi_i\}_{i=1}^{S} \), we calculate the second term in Equation 2 as (Section 4.1 explains \( \gamma \)):

\[
p(t| s, \gamma, a) = \int p(t| s, \Psi_{a}, a)p(\Psi_{a}| \gamma) d\Psi_{a}
\]

\[
= \frac{\Gamma(\sum_{t=1}^{T} \gamma_t)}{\Gamma(\gamma)} \prod_{s=1}^{S} \prod_{t=1}^{T} \Gamma(\sum_{i=1}^{S} (n_{s,a,t} + \gamma_t))
\times \prod_{m=1}^{M} \exp \left\{ \lambda \sum_{j=1}^{T_{m}} \mathcal{I}(t_{j} = t) \right\}
\]

where \( n_{s,a,t} \) equals the number of sentences in reviews associated with attribute \( a \), sentiment \( s \), and aspect \( t \); \( M \) is the number of sentences in reviews; \( L_m \) is the set of sentences corresponding to sentence \( m \).

Similarly, for the third term in Equation 2, by integrating over \( \Theta = \{\theta_i\}_{i=1}^{D} \), we obtain (Section 4.1 explains \( \beta \)):

\[
p(s| \beta) = \int p(s| \Theta)p(\Theta| \beta) d\Theta
\]

\[
= \left( \frac{\Gamma(\sum_{s=1}^{S} \beta_s)}{\prod_{s=1}^{S} \Gamma(\beta_s)} \right)^D \prod_{d=1}^{D} \prod_{s=1}^{S} \Gamma(n_{d,s} + \beta_s),
\]

where \( D \) is the number of reviews; \( n_{d,s} \) is the number of times that a sentence from review \( d \) is associated with sentiment \( s \); and \( n_{d} \) is the number of sentences in review \( d \).

For each sweep of a Gibbs iteration, we sample latent aspect \( t \) and sentiment \( s \) as follows:

\[
p(s_i = s, t_i = t| s_{-i}, t_{-i}, w, a)
\]

\[
\propto \frac{n_{s,a,-i} + \gamma_t}{\sum_{t=1}^{T} (n_{s,a,-i} + \gamma_t)} \times \frac{n_{d,-i} + \beta_s}{\sum_{s=1}^{S} (n_{d,-i} + \beta_s)} \times \prod_{v \in W_i} \frac{C_{s,v} - n_v + \alpha_v + c}{C_{i,v} - n_v + c} \times \exp \left\{ \lambda \sum_{j \in L_i} \mathcal{I}(t_j = t) \right\}
\]

where \( n_{s,a,t} \) is the number of sentences from reviews associated with attribute \( a \), sentiment \( s \), and aspect \( t \); \( n_{d,s} \) is the number of sentences from review \( d \) associated with sentiment \( s \); \( W_i \) is the set of words in sentence \( i \); \( C_{s,v} \) is the count of word \( v \) in sentence \( s \); \( C_{i,v} \) is the number of words in sentence \( i \); \( n_{s,a,t} \) is the number of words \( v \) assigned sentiment \( s \) and aspect \( t \); \( n_{s,a} \) is the number of words assigned sentiment \( s \) and aspect \( t \) in all reviews; \( L_i \) is the set of corresponding sentences of sentence \( i \); and an index of \(-i\) indicates excluding sentence \( i \) from the count.

Equations 7, 8, and 9, respectively, approximate the probabilities of word \( w \) occurring given sentiment \( s \) and aspect \( t \); of aspect \( t \) of a sentence occurring given sentiment \( s \) and attribute \( a \); of sentiment \( s \) occurring given document \( d \).

\[
\phi_{s,t,w} = \frac{n_{s,a,w,t} + \alpha_w}{\sum_{v=1}^{W} (n_{s,a,w,t} + \alpha_v)}
\]

\[
\psi_{s,t,a} = \frac{n_{s,a,t} + \gamma_t}{\sum_{t=1}^{T} (n_{s,a,t} + \gamma_t)}
\]

\[
\theta_{d,s} = \frac{n_{d,s} + \beta_s}{\sum_{s=1}^{S} (n_{d,s} + \beta_s)}
\]

The generalized Pólya Urn model (Mahmoud, 2008) has been used for encoding word co-
occurrence information into topic models. Consider an urn containing a mixture of balls, each of which is tagged with a term, for each sampling sweep we draw a ball from the urn. In a standard Pólya Urn model, as used in LDA, we return the ball to the urn with another ball tagged with the same term. This process provides burstiness of the probability of seeing a term but ignores the covariance. The probability increase of one term decreases the probability of the other words. In the generalized Pólya Urn model, when a ball is drawn from the urn, we replace it with two new balls with a set of balls tagged with related terms. Similar to previous models (Mimno et al., 2011; Chen et al., 2013; Fei et al., 2014; Zhang and Singh, 2018), to increase the probability of having semantically related words appear in the same topic, Trait applies a generalized Pólya Urn model in each Gibbs sweep and uses weight $\varepsilon$ to promote related words based on the cosine similarity between their Word2Vec (Mikolov et al., 2013) embeddings.

4 Evaluation

To assess Trait’s effectiveness, we select the hotel and restaurant domains and prepare four review datasets associated with three attributes: author, trip type, and location. HotelUser, HotelType, and HotelLoc are sets of hotel reviews collected from TripAdvisor. HotelUser contains 28,165 reviews posted by 202 randomly selected reviewers, each of whom contributes at least 100 hotel reviews. HotelType contains reviews associated with five trip types including business, couple, family, friend, and solo. HotelLoc contains a total of 136,446 reviews about seven US cities, split approximately equally. ResUser is a set of restaurant reviews from Yelp Dataset Challenge (2019). It contains 23,874 restaurant reviews posted by 144 users, each of whom contributes at least 100 reviews. Table 1 summarizes our datasets. Datasets and source code are available for research purposes (Trait, 2019).

<table>
<thead>
<tr>
<th>Statistic</th>
<th>HotelUser</th>
<th>ResUser</th>
<th>HotelType</th>
<th>HotelLoc</th>
</tr>
</thead>
<tbody>
<tr>
<td># of reviews</td>
<td>28,165</td>
<td>23,873</td>
<td>22,984</td>
<td>136,446</td>
</tr>
<tr>
<td># of sentences</td>
<td>362,153</td>
<td>276,008</td>
<td>302,920</td>
<td>1,428,722</td>
</tr>
<tr>
<td>Sentence/Review</td>
<td>13</td>
<td>12</td>
<td>13</td>
<td>10</td>
</tr>
<tr>
<td>Words/Sentence</td>
<td>8</td>
<td>7</td>
<td>7</td>
<td>7</td>
</tr>
</tbody>
</table>

We remove stop words and HTML tags, expand typical abbreviations, and mark special named entities using a rule-based algorithm (e.g., replace a URL by #LINK# and replace a monetary amount by #MONEY#) and the Stanford named entity recognizer (Finkel et al., 2005). We use Porter’s (1980) stemming algorithm. To handle negation, for any word pair whose first word is no, not, or nothing, we replace the word pair by a negated term, e.g., producing not_work and not_quiet. Finally, we split each review into constituent sentences.

4.1 Parameter Settings

Trait includes three manually tuned hyperparameters that have a smoothing effect on the corresponding multinomial distributions. Hyperparameter $\alpha$ is the Dirichlet prior of the word distribution. We use asymmetric priors based on a sentiment lexicon. Table 2 shows Trait’s sentiment word list as prior knowledge to set asymmetric priors. This list extends Turney and Littman’s (2003) list with additional general sentiment words. For any word in the positive list, we set $\alpha$ to 0 if this word appears in a sentence assigned a negative sentiment, and to 5 if this word appears in a sentence assigned a positive sentiment, and conversely for words in the negative list. For all remaining words, we set $\alpha$ to 0.05. We set hyperparameter $\beta$, the Dirichlet prior of the sentiment distribution, to 5 for both sentiments. We set hyperparameter $\gamma$, the Dirichlet prior of the aspect distribution, to $50/T$ for all models, where $T$ is the number of aspects. We set the reinforcement weight of structural and semantic correspondence $\lambda$ to 1.0; sentence semantic similarity $\rho$ to 0.7; and, related word promoting weights to 0.3 for hotel reviews and 0.1 for restaurant reviews.

4.2 Quantitative Evaluation

Whether topics (word clusters) are semantically cohesive is crucial in assessing topic modeling approaches. As in previous studies (Nguyen et al., 2015a,b; Yang et al., 2017), we adopt Normalized Pointwise Mutual Information (NPMI) (Lau et al., 2014) and W2V (O’Callaghan et al., 2015) as our evaluation metrics. Higher NPMI and W2V scores indicate greater semantic cohesion. We compare Trait with four baselines: AT, JST, ASUM, and AATS. We perform our evaluation on HotelUser and ResUser based on the top 20 words in each sentiment-aspect pair. We split data into five folds.
Table 2: Lists of sentiment words.

**Positive**

amazing, attractive, awesome, best, comfortable, correct, enjoy, excellent, fantastic, favorite, fortunate, free, fun, glad, good, great, happy, impressive, love, nice, not_bad, perfect, positive, recommend, satisfied, superior, thank, worth

**Negative**

annoying, bad, complain, disappointed, hate, inferior, junk, mess, nasty, negative, not_good, not_like, not_recommend, not_worth, poor, problem, regret, slow, small, sorry, terrible, trouble, unacceptable, unfortunate, upset, waste, worst, worthless, wrong

and use training split to train all models. For each number of aspects, we conduct a two-tailed paired t-test for each of the pairwise comparisons. Throughout, *, †, and ‡ indicate significance at 0.05, 0.01, and 0.001, respectively.

Table 3 shows average NMPI and W2V scores for different numbers of aspects. AT performs worst, possibly due to missing conditions on sentiments. ASUM and JST are comparable. Trait outperforms all others, with the highest NMPI and W2V scores for each number of aspects. Table 4 shows similar conclusions for restaurant reviews.

For both datasets, Trait’s improvements of topic coherence over baseline models are statistically significant for HotelUser (p < 0.001) and ResUser (p = 0.002). Trait allows reviews written by the same or similar authors to have idiosyncratic preferences over aspects and sentiments. Trait assigns aspects to sentences by sampling attribute-specific aspect distributions. These distributions are regularized by the Markov Random Fields. Sentences with a high degree of correspondence have a high probability to be assigned the same aspects.

### 4.3 Sentiment Classification

Automatically detecting the sentiment of a document is an important task in sentiment analysis. We compare Trait with JST, ASUM, and AATS for document-level sentiment classification using HotelUser and ResUser. We use integer ratings of reviews to collect ground-truth labels. Reviews with ratings at three and above are labeled as positive and the rest are labeled as negative. Note that our datasets are imbalanced. We conduct five-fold cross-validation with the two-tailed paired t-test. For each user, we use 80% of reviews for training and 20% for testing. For evaluation metrics, we adopt accuracy (Acc) and area under the curve (AUC) of the Receiver Operating Characteristic (ROC) curve. ROC plots the true positive rate against the false positive rate. AUC-ROC is a standard metric for evaluating classifiers’ performance on imbalanced data.

Table 5 reports the results of sentiment classification on hotel reviews. AATS achieves better accuracy but worse AUC scores than JST. ASUM yields better accuracy and comparable AUC scores compared with JST. Trait consistently outperforms all baseline models given different aspect numbers with an average gain in accuracy of 3%. Incorporating attributes and structural and semantic cor-
The correspondence into conditional probability distributions greatly benefit the model in capturing dependencies among attributes, aspects, and sentiments.

<table>
<thead>
<tr>
<th></th>
<th>T=20 Acc</th>
<th>T=20 AUC</th>
<th>T=40 Acc</th>
<th>T=40 AUC</th>
<th>T=60 Acc</th>
<th>T=60 AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>AATS</td>
<td>0.79</td>
<td>0.47</td>
<td>0.80</td>
<td>0.48</td>
<td>0.82</td>
<td>0.50</td>
</tr>
<tr>
<td>JST</td>
<td>0.59</td>
<td>0.71</td>
<td>0.61</td>
<td>0.73</td>
<td>0.64</td>
<td>0.73</td>
</tr>
<tr>
<td>ASUM</td>
<td>0.80</td>
<td>0.78</td>
<td>0.84</td>
<td>0.78</td>
<td>0.87</td>
<td>0.74</td>
</tr>
<tr>
<td>Trait</td>
<td>0.86†</td>
<td>0.79 †</td>
<td>0.87†</td>
<td>0.78 †</td>
<td>0.88  †</td>
<td>0.79 †</td>
</tr>
</tbody>
</table>

4.4.1 Summarization

Trait outputs profiles that summarize attributes in terms of aspects and sentiments in reviews.

Figure 3 shows the profiles of four US cities generated by Trait. We visualize the profiles as aspect-clouds using the top 30 aspects in each sentiment. The size of the aspect label corresponds to its aspect probabilities. Due to space constraints, we place the profiles of Boston, Chicago, and Orlando in Table 7.

These profiles yield salient summaries for each city. For example, Strip and Casino are the top two positive aspects for Las Vegas, a resort city for gambling. We see from the reviews that most hotels with high ratings are located on the Strip. For Boston, Chicago, and New York, Location and PublicTrans are the top positive aspects. These cities rank top on the lists of U.S. cities with high transit ridership (Ridership, 2019) and walkability (Walkability, 2019). Hotels’ proximity to public transportation, shopping, restaurants, and attractions is appealing to several reviewers. We see RoomSize appears in the top five negative aspects. These three cities have among the most expensive hotel room rates (Statista, 2019). Assuming consumers expect more when they pay more, we conjecture that a failed expectation could be caused by room size, especially in New York, where room sizes are smaller than elsewhere in the US (NYC, 2019). For Miami, Transportation is attractive, presumably because many cruises depart from Miami.

Figure 4 shows the results for HotelType: Cleanliness, Internet, TV, and Upgrade are most likely to lead to a negative sentiment for business travelers. For couples, Atmosphere and RestArea are most preferred positive aspects. Family, Tour, and Attraction are most positive aspects for families. Solo travelers, on business or tourism, express most opinions toward both Transportation.

Table 7 (bottom rows) lists the top five aspects for four authors from HotelUser. We can observe strong commonality between Author A and B. They both like to express positive sentiment on Helpfulness, View, Value, and Breakfast. They are like to express not returning a hotel and the negative sentiments are mostly toward Value, Checkin, and Checkout. There is little commonality between Authors C and D. For Author C, Internet and Comfort are most attractive whereas Staff and Room are most appealing aspects for Author D. In terms of negatives, Checkout is the only aspect shared between Authors C and D.

4.4.2 Similarity

Attribute profiles can be used not only for summarization, but also for measuring similarity between attribute values with respect to aspects and sentiments, which can support attribute-based applica-

Table 7: Top five aspects discovered by Trait.

<table>
<thead>
<tr>
<th></th>
<th>Boston (P)</th>
<th>Boston (N)</th>
<th>Chicago (P)</th>
<th>Chicago (N)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PublicTrans</td>
<td>RoomSize</td>
<td>Location</td>
<td>RoomSize</td>
<td>Location</td>
</tr>
<tr>
<td>Location</td>
<td>Service</td>
<td>PublicTrans</td>
<td>Service</td>
<td>PublicTrans</td>
</tr>
<tr>
<td>Decor</td>
<td>Parking</td>
<td>Decor</td>
<td>Parking</td>
<td>Decor</td>
</tr>
<tr>
<td>Bar</td>
<td>RoomNoise</td>
<td>Bar</td>
<td>Elevator</td>
<td>RoomNoise</td>
</tr>
<tr>
<td>Service</td>
<td>StreetNoise</td>
<td>Service</td>
<td>ACHotwater</td>
<td>StreetNoise</td>
</tr>
<tr>
<td>Overall</td>
<td>Bed</td>
<td>Location</td>
<td>Breakfast</td>
<td>Location</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Author A (P)</th>
<th>Author A (N)</th>
<th>Author B (P)</th>
<th>Author B (N)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Helpfulness</td>
<td>NotReturn</td>
<td>Helpfulness</td>
<td>NotReturn</td>
</tr>
<tr>
<td>Value</td>
<td>View</td>
<td>View</td>
<td>Value</td>
</tr>
<tr>
<td>Charge</td>
<td>Breakfast</td>
<td>Checkin</td>
<td>Food</td>
</tr>
<tr>
<td>Checkout</td>
<td>HotelType</td>
<td>Checkin</td>
<td>Restaurant</td>
</tr>
<tr>
<td>Comfort</td>
<td>Transportation</td>
<td>Checkout</td>
<td>Food</td>
</tr>
<tr>
<td>Internet</td>
<td>Elevator</td>
<td>Staff</td>
<td>Value</td>
</tr>
<tr>
<td>Comfort</td>
<td>Room</td>
<td>Room</td>
<td>NotReturn</td>
</tr>
<tr>
<td>TripType</td>
<td>Food</td>
<td>Recommend</td>
<td>Checkout</td>
</tr>
<tr>
<td>View</td>
<td>StreetNoise</td>
<td>Value</td>
<td>Room</td>
</tr>
<tr>
<td>Breakfast</td>
<td>Checkout</td>
<td>TripType</td>
<td>Charge</td>
</tr>
</tbody>
</table>
tions such as recommender systems. Our metric of similarity between distinct values of the same attribute is the Jensen-Shannon distance (JSD) (Endres and Schindelin, 2003), the square root of Jensen-Shannon divergence. We compute, $D_{JS}$, the JSD of attribute profiles $P$ and $Q$ as

$$D_{JS} = \sqrt{\frac{1}{2} D_{KL}(P||M) + \frac{1}{2} D_{KL}(Q||M)}$$

$$M = \frac{1}{2}(P + Q),$$

where $D_{KL}(P||Q)$ is the Kullback-Leibler (KL) divergence of probability distributions $P = \{p_1, \ldots, p_n\}$ and $Q = \{q_1, \ldots, q_n\}$:

$$D_{KL}(P||Q) = \sum_i p_i \log \frac{p_i}{q_i}. \quad (11)$$

As a baseline, we use a vector space model based on USE sentence embeddings. We calculate mean sentence embeddings for each review. Then, given two sets of reviews, $D = \{d_1, \ldots, d_m\}$ and $R = \{r_1, \ldots, r_n\}$, we compute their similarity as follows (here $\text{sim}(d_i, r_j)$ is the cosine similarity between review $d_i$ and $r_j$):

$$S_{D,R} = \frac{1}{m+n} \sum_{i=1}^{m} \sum_{j=1}^{n} \text{sim}(d_i, r_j). \quad (12)$$

Figure 5a shows the similarities among the profiles of the seven cities generated by the baseline model. We see that Boston is close to New York and Chicago; Las Vegas is far away from Los Angeles and Miami but close to New York, Boston, and Chicago. Figure 5b shows the results generated by Trait. Here, dissimilarity corresponds to distance normalized to $[0, 1]$. Boston, Chicago, and New York are close to each other, as are Los Angeles and Miami; Las Vegas is far from each of the others; and, Orlando is far from all except Miami. Trait’s results are arguably more plausible than what the baseline approach produces.

As discussed earlier, hotels in Boston, Chicago, and New York have common characteristics; Las Vegas differs strongly from the others because it is a resort city and its major hotels are combined with casinos; Orlando is a tourism destination but differs from Las Vegas in that it is famous for local attractions, such as theme parks. Orlando’s profile exhibits that travelers there tend to be more aware of aspect $\text{Attraction}$ than elsewhere. An interesting pair is Los Angeles and Miami. We see that $\text{Location}$ appears as the most important aspect on the positive side for both of them. Such a similarity could be partially explained by the fact that both Los Angeles and Miami serve as locations for taking cruises. Also, the common aspect $\text{Safety}$...
Figure 5: Force fields of similarities: cities.

Figure 6a shows similarities among the five trip types generated by the baseline model. Business is closer to Family than Solo and Friend; Solo is closer to Family than Friend. Trait generates more reasonable results, as shown in Figure 6b. Business is far from others but is closer to Friend and Solo than to Couple and Family. Couple, Family, and Friend are relatively close to each other. Business reviewers attend to different aspects from other reviewers. Further, Solo and Friend contain reviews of business trips, although the authors did not select Business as the trip type. However, this situation does not happen for Couple and Family.

Figure 7 shows similarities among 20 authors, including the four authors mentioned in Section 4.4.1. We see that Authors A and B are close to each other whereas Authors C and D are far away from each other. The results are aligned with their aspect and sentiment profiles.

5 Discussion and Conclusion

Trait not only shows that capturing structural and semantic correspondence leads to improved performance in terms of coherence and naturalness of the aspects discovered but can also be realized in an unsupervised framework. Trait outperforms competing approaches across multiple datasets.

These results open up interesting directions for future work. One direction is to learn disentangled latent representations for attributes in neural network’s space, such as for disentangling aspects (Jain et al., 2018), text style (John et al., 2019), and syntax and semantics (Chen et al., 2019; Bao et al., 2019). Another direction is to develop a content-based recommender based on Trait, since it provides an effective unsupervised solution for generating profiles based on different attributes.

6 Acknowledgments

We would like to thank the anonymous reviewers for helpful comments and corrections. MPS would like to acknowledge the US Department of Defense for partial support through the NCSU Laboratory for Analytic Sciences.
References


Hyungsul Kim, Yizhou Sun, Julia Hockenmaier, and Jiawei Han. 2012. ETM: Entity topic models for mining documents associated with entities. In *Proceedings of the 12th IEEE International Conference on Data Mining (ICDM)*, pages 349–358, Brussels.


