Venue Appropriateness Prediction for Contextual Suggestion

Mohammad Aliannejadi, Ida Mele, and Fabio Crestani
Università della Svizzera italiana (USI)
Via G. Buffi 13
Lugano, Switzerland
{mohammad.alian.nejadi,ida.mele,fabio.crestani}@usi.ch

ABSTRACT

This technical report presents the work of Università della Svizzera italiana (USI) at TREC 2016 Contextual Suggestion track. The goal of the Contextual Suggestion track is to develop systems that could make suggestions for venues that a user will potentially like. Our proposed method attempts to model the users' behavior and opinion by training a SVM classifier for each user. It then enriches the basic model using additional data sources such as venue categories and taste keywords to model users' interest. For predicting the contextual appropriateness of a venue to a user's context, we modeled the problem as a binary classification one. Furthermore, we built two datasets using crowdsourcing that are used to train a SVM classifier to predict the contextual appropriateness of venues. Finally, we show how to incorporate the multimodal scores in our model to produce the final ranking. The experimental results illustrate that our proposed method performed very well in terms of all the evaluation metrics used in TREC.

KEYWORDS

Contextual Suggestion, User Modeling, SVM, TREC

1 INTRODUCTION

This paper describes the participation of Università della Svizzera italiana (USI) at TREC 2016 Contextual Suggestion[1] track[6]. This year's Contextual Suggestion track consisted of two phases, namely, *Phase 1* and *Phase 2*. We participated in both of them. For Phase 1, the participants were given a list of 442 users who had visited from 30 to 60 venues in 1 or 2 cities. For each user, context and profile were defined. The task consisted in producing for each user a ranked list of 50 venues to visit in a new city. The suggested venues were limited to be from the collection of venues that was provided by the organizers. As for Phase 2, the contexts and profiles were the same. However, for each user there was also a list of candidate suggestions. The task consisted in reranking the list of candidate suggestions to produce the best ranking according to each user's profile and context.

For both phases we followed a similar approach to our participation at TREC 2015 [1, 2] to create an initial user model. However, this year we focused more on the user context and explored how we could predict the level of appropriateness given a venue and a user's context. We modeled it as a classification problem and picked 10% of the data randomly as our training set. More specifically, we built two datasets: one containing the labels for training data and the other one containing the features. The reported results of TREC shows that the base user model was able to capture user interest and opinion effectively and the contextual appropriateness model enabled the model to suggest venues that were more appropriate to the users context.

The remainder of the paper is organized as follows. Section 2 details our approach to gather information and model the users. Then, Section 3 describes how we predict the contextual appropriateness of venues. Section 4 presents our experimental evaluation. Finally, Section 5 concludes the study.

2 CONTEXTUAL SUGGESTION

In this task we modeled users by leveraging data mainly from Yelp. We further enriched these models using data from Foursquare and the contextual appropriateness prediction model. We used classifiers to model users' behavior and we enriched user models by combining other measures. At a high level, our system consists of five modules:

- Information gathering
- User modeling
- User model enrichment
- Contextual appropriateness prediction
- Suggestion ranking

The system's execution cycle starts with the module for data collection. This module collects information from the most important data sources, such as Yelp and Foursquare. Using the gathered information, user modeling creates a basic model for each user. Then, model enrichment improves the user models by adding two additional measures based on *user-category profiles* and Foursquare's *venue-taste keywords*. The contextual appropriateness prediction model predicts how appropriate each venue is to each context, providing an additional score in our model. Subsequently, suggestion ranking module ranks all candidate places. In the following sections we provide more details for each component.

2.1 Information Gathering

Since we participated in both phases, we were provided with users' history and a large number (virtually 600K) of venues to be ranked for Phase 1. However, since a list of candidate suggestion was included in the dataset for Phase 2, the number of links presenting venues that we had to crawl was reduced to virtually 19K. Moreover, the task of suggesting places to each user in Phase 2 was limited to a certain number of venues, hence it was crucial not to miss any information relevant to the target venues. Additionally, although almost half of the URLs was from known location-based social networks (LBSNs), such as Yelp and Foursquare, another half of the URLs was from less known websites (e.g., the places' official web pages). Consequently, effort was made to find the corresponding profiles of these places on Yelp and Foursquare, too.

[1] https://sites.google.com/site/treccontext/

To collect data we performed the following steps:

1. We discarded the attractions that were rated by the users with a score of '−1' or '2'. This is due to the fact that these places either were not assigned any ratings or their rating was neutral, thus insignificant.
2. We detected and discarded broken links.
3. We downloaded the links from the target LBSNs, namely, Yelp and Foursquare.
4. For each venue on each of the above-mentioned LBSNs we found the corresponding profiles on the other LBSN (e.g., for a given Yelp profile, we found its corresponding profile on Foursquare).
5. For the other attractions with unknown links, we downloaded the web pages, and analyzed their contents to find their corresponding profiles on the two above-mentioned LBSNs.

For Phase 1, due to the large number of venues to be crawled, we only crawled Foursquare using their API. For Phase 2, we managed to crawl both Yelp and Foursquare for all the venues.

2.2 User Modeling

We modeled each user by training a classifier using example suggestions. Our intuition was that a user's opinion regarding a venue could be learned based on the opinions of other users who gave the same rating as the target user to the same venue [2]. To train a classifier per user we extracted negative and positive samples as explained in the following:

- **Positive samples:** We elicited the positive reviews of positive-example suggestions.
- **Negative samples:** Likewise, we elicited the negative reviews of negative-example suggestions.

We defined the positive example suggestions as the venues that a user rated as 3 or 4, so the positive reviews were those reviews that were rated accordingly. Analogously, negative example suggestions and reviews were defined as taking ratings of 0 and 1.

We adopted a binary classifier per user to learn why she might have liked/disliked some venues and to assign a score for a new venue[2]. The binary classifier was trained using the positive and negative profiles for each user. Since the users' reviews may contain a lot of noise and off-topic terms, we calculated a TF-IDF score and used it as the feature vector for training the classifier. As classifier we used Support Vector Machine (SVM) [5] and considered the value of the SVM's decision function as the score (S_{rev}) since it gives us an idea on how close and relevant a venue is to a user profile.

2.3 User Model Enrichment

We used some frequency-based scores to enrich the user model. Frequency-based scores are based on the assumption that a user visits the venues that she likes more frequently than other venues and rates them positively. We created positive and negative profiles based on categories of venues that a user had visited and calculated their corresponding normalized frequencies. A new venue was then compared with the user's profiles to compute a similarity score. We explain the score for venue categories. The same method was applied for other frequency-based scores.

Given a user u and her history of rated venues $h_u = \{v_1, \ldots, v_n\}$, each venue has a corresponding list of categories $C(v_i) = \{c_1, \ldots, c_k\}$. We define the user category profiles as follows:

Definition 2.1. A **Positive-Category Profile** is a set of all distinct categories belonging to venues that a particular user has previously rated positively. A **Negative-Category Profile** is defined analogously for the venues that are rated negatively.

We assigned a user-level-normalized frequency value to each category in the positive/negative category profile. The user-level-normalized frequency for a positive/negative category profile is defined as follows:

Definition 2.2. A **User-level-Normalized Frequency** for an item (e.g., category) in a profile (e.g., positive-category profile) is defined as:

$$\mathrm{cf}_u^+(c_i) = \frac{count(c_i)}{\sum_{v_k \in h_u} \sum_{c_j \in C(v_k)} 1} .$$

A user-level-normalized frequency for negative category profile, cf^-, is calculated analogously.

Based on Definitions 2.1 and 2.2 we created positive/negative category profiles for each user. Let u be a user and v be a candidate venue, then the category-based similarity score $S_{cat}(u, v)$ is calculated as follows:

$$S_{cat}(u, v) = \sum_{c_i \in C(v)} \mathrm{cf}_u^+(c_i) - \mathrm{cf}_u^-(c_i) . \quad (1)$$

The frequency-based category similarity score was calculated using the data from both Yelp (denoted as S_{cat}^Y) and Foursquare (denoted as S_{cat}^F).

The venue taste keywords on Foursquare are special terms extracted from users' tips[3] and are very informative. For example, 'Central Park' in 'New York City' is described by these taste keywords: *picnics, biking, scenic views, trails, park, …* These terms are very informative and often express characteristics of an attraction as well as the users sentiment. Figure 1 shows a snapshot from Foursquare with the venue taste keywords and categories for Central Park in New York. We created the positive and negative profiles using venue taste keywords for each user following Definition 2.1 and calculated the user-level-normalized frequencies in the profiles following Definition 2.2. The frequency-based similarity score for venue taste keywords was then calculated in a similar way to Equation 1(S_{key}^F).

3 CONTEXTUAL APPROPRIATENESS PREDICTION

As described in the guidelines of the track, in addition to the users' history of preferences, the users' context is also available. The context consists of 4 dimensions:

- Trip type: business, holiday, other
- Trip duration: night out, day trip, weekend trip, longer
- Group type: alone, friends, family, other

[2] An alternative to binary classification would be a regression model, but it is inappropriate due to the data sparsity, thus degrading the accuracy of venue suggestion.

[3] Tips on Foursquare are short reviews written by users.

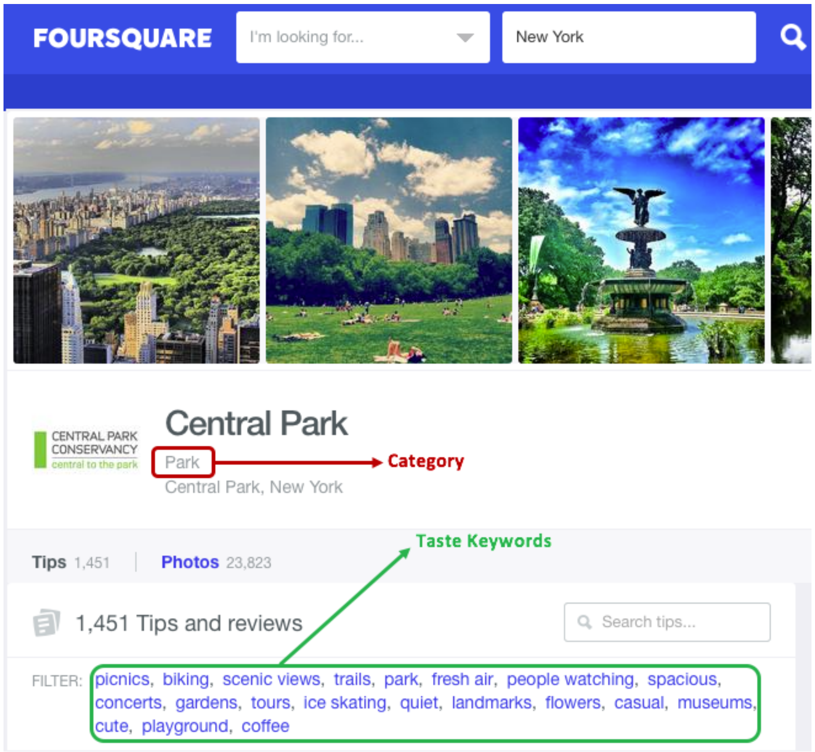


Figure 1: A sample snapshot of Foursquare.com illustrating the difference of venue categories and venue taste keywords.

- Trip season: winter, summer, autumn, spring

Given a venue v and a user's context ux, our aim was to predict how appropriate is v for ux: $F_{ap}(v, ux)$. We assume that each venue is represented by its corresponding categories. Therefore, our problem is to predict the appropriateness of a given list of venue categories ($\mathbf{c} = \langle c_1, \dots, c_n \rangle$) for ux. We broke the problem into a set of simpler sub problems, i.e., for each $c_i \in \mathbf{c}$ we predicted the appropriateness score of c_i to ux assuming that categories are independent: $F_{ap,c}(c_i, ux)$.

Furthermore, we calculated the appropriateness score as follows: $F_{ap}(v, ux) = min\{F_{ap,c}(c_1, ux), \dots, F_{ap,c}(c_n, ux)\}$. For instance, assume that a user is going to visit a venue whose categories ($\mathbf{c}$) are: *burger joint* and *bar*. The user context (ux) is: *business* (trip type), *day trip* (trip duration), *alone* (group type), and *autumn* (trip season). We calculate $F_{ap,c}(\text{'burgerjoint'}, ux)$ and $F_{ap,c}(\text{'bar'}, ux)$. $F_{ap}(v, ux)$ would be the minimum value of the two functions.

We trained a SVM classifier to calculate $F_{ap,c}$. As training samples, we picked 10% of samples in the dataset and asked human assessors to judge them. For each instance, we assigned three assessors. Figure 2 shows the sample questions we asked the workers. We considered a venue category appropriate to a context if more than 2 users agreed on that.

3.1 Features

As features for classification, we considered the appropriateness of each venue category to each contextual dimension. Therefore, for all pairs of category-context, we needed to define the appropriateness of the pairs. This is not a trivial task since it could be very subjective. For instance, for a 'family' (group type), it is supposedly **not** appropriate to visit a 'nightlife spot' (objective). While on a 'business trip' (trip type), visiting a 'pharmacy' depends mostly on the user and other subjective factors. In order to determine how subjective is a pair, we asked human workers to assess the *appropriateness* of each pair. For each pair we made sure that at least 5 different workers assessed it. The level of agreement between workers was considered as the level of subjectivity of each pair. Figure 3 shows some example questions we asked the workers. It is worth noting that we included almost all category-context pairs for this job irrespective of their availability in the TREC collection. This makes this dataset more general.

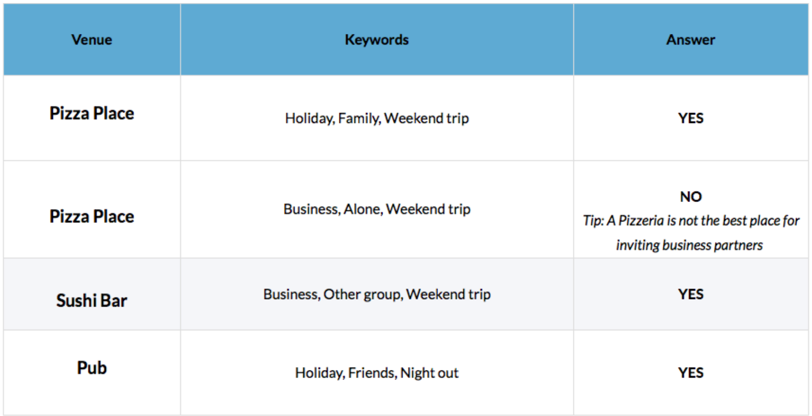
We trained a SVM classifier using these features and called the value of the decision function of the classifier $F_{ap,c}$. The value of $F_{ap}(v, ux)$ was considered as another similarity score in our model, referred to as S^F_{cxt}.

4 EXPERIMENTAL RESULTS

We estimated the similarity between user and candidate suggestion using the following equation:

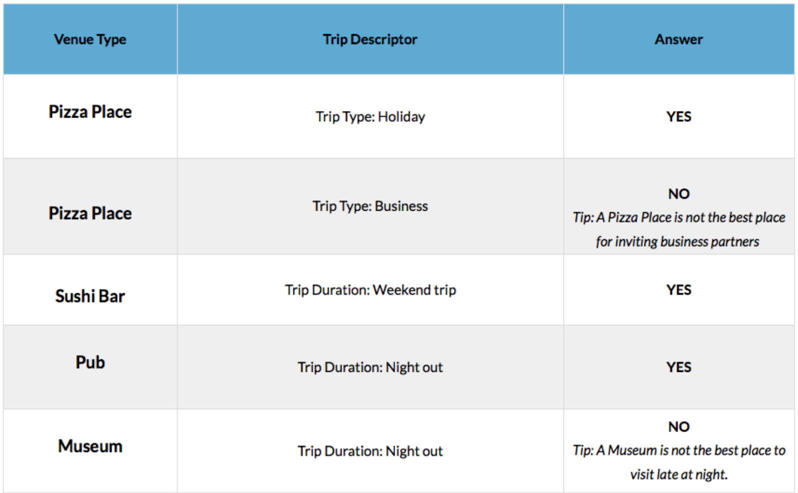
$$\begin{aligned}\text{Similarity}(u, v) = \omega_1 S^Y_{rev}(u, v) + \omega_2 S^Y_{cat}(u, v) + \\ \omega_3 S^F_{cat}(u, v) + \omega_4 S^F_{key}(u, v) + \omega_5 S^F_{cxt}(u, v) \,,\end{aligned} \tag{2}$$

where $\omega_{1\dots5}$ are the weights assigned to these scores, u is a given user and v is a given venue. To find the optimum setting for the



Venue	Keywords	Answer
Pizza Place	Holiday, Family, Weekend trip	YES
Pizza Place	Business, Alone, Weekend trip	NO *Tip: A Pizzeria is not the best place for inviting business partners*
Sushi Bar	Business, Other group, Weekend trip	YES
Pub	Holiday, Friends, Night out	YES

Figure 2: Sample questions for crowdsourcing. Each question is answered by at least three workers.



Venue Type	Trip Descriptor	Answer
Pizza Place	Trip Type: Holiday	YES
Pizza Place	Trip Type: Business	NO *Tip: A Pizza Place is not the best place for inviting business partners*
Sushi Bar	Trip Duration: Weekend trip	YES
Pub	Trip Duration: Night out	YES
Museum	Trip Duration: Night out	NO *Tip: A Museum is not the best place to visit late at night.*

Figure 3: Sample questions for crowdsourcing. Each question is answered at least by 5 workers. The agreement between the workers reveals the level of subjectivity of each pair.

weights associated with each score, we conducted a 5-fold cross validation. Note that different submitted runs consisted of different sets of scores; subsequently, the weights and Equation 2 would be different for each run. We ranked the candidate suggestions according to the similarity measure computed by this module. The higher the similarity score, the higher the rank.

By applying the method to our gathered dataset, we submitted two runs for Phase 1: '*USI1*' and '*USI2*' and three runs for Phase 2: '*USI3*', '*USI4*', and '*USI5*'. We used different combinations of scores for each run as described below:

- USI1: We only used S^F_{cat} to rank the candidate suggestions.
- USI2: We reranked the top 10 venues of USI1, using S^F_{key}.
- USI3: We used Factorization Machines (FM) [7] to combine all the crawled data.
- USI4: We used all the scores except for the contextual appropriateness score, namely, S^Y_{rev}, S^Y_{cat}, S^F_{cat}, and S^F_{key}.
- USI5: We used all the scores, namely, S^Y_{rev}, S^Y_{cat}, S^F_{cat}, S^F_{key}, and S^F_{cxt}.

In this year's task, we were given 442 profile-context pairs. For each pair a user's history consisted of 30 to 60 venues and the number of candidate suggestions to be ranked varied for each user (Phase 2). Track organizers evaluated all submitted runs using three evaluation metrics, namely, nDCG@5 (normalized discounted cumulative gain at 5), P@5 (precision at 5), and MRR (mean reciprocal rank). A suggestion is considered relevant if it is rated 3 or 4 by user.

Table 1 demonstrates the overall average performances of our runs. It could be seen that for Phase 1 both our runs outperformed the median of all submitted runs. This confirms the effectiveness of the user model enrichment method we proposed in this work. However, our run 'USI2' performed better than the other, suggesting that venue taste keywords (S^F_{key}) was effective for modeling users. As for Phase 2, 'USI3' did not perform very well compared to TREC

Table 1: Overall Average Performances. Bold values denote the best performing runs w.r.t each evaluation metric.

		nDCG@5	P@5	MRR
Phase 1	USI1	0.2578	0.3934	0.6139
	USI2	**0.2826**	**0.4295**	**0.6150**
	Median	0.2133	0.3508	0.5041
Phase 2	USI3	0.2470	0.4259	0.6231
	USI4	0.3234	0.4828	**0.6854**
	USI5	**0.3265**	**0.5069**	0.6796
	Median	0.2562	0.3931	0.6015

median, while both 'USI4' and 'USI5' performed very well. In fact, 'USI5' performed the best, indicating that the proposed contextual appropriateness prediction model was able to effectively predict the appropriateness of a venue given a user context.

5 CONCLUSIONS

In this technical report we presented the methodology we applied for our participation in the TREC 2016 Contextual Suggestion track. In this track, we showed that our method for suggesting venues to users based on their profiles and context is very effective. Results suggest that our approach combining multimodal information from multiple LBSNs is able to model users effectively as in our previous works [1–4]. In this work, moreover, the results indicate that the contextual appropriateness score is able to effectively predict the appropriateness of a given venue with respect to a user context.

As future work, we plan to explore other ways to incorporate the contextual appropriateness score into our base model.

ACKNOWLEDGMENTS

This research was partially funded by the RelMobIR project of the Swiss National Science Foundation (SNSF).

REFERENCES

[1] Mohammad Aliannejadi, Seyed Ali Bahrainian, Anastasia Giachanou, and Fabio Crestani. 2015. University of Lugano at TREC 2015: Contextual Suggestion and Temporal Summarization Tracks. In *TREC 2015*. NIST.

[2] Mohammad Aliannejadi, Ida Mele, and Fabio Crestani. 2016. User Model Enrichment for Venue Recommendation. In *AIRS 2016*. Springer.

[3] Mohammad Aliannejadi, Ida Mele, and Fabio Crestani. 2017. Personalized Ranking for Context-Aware Venue Suggestion. In *SAC 2017*. ACM.

[4] Mohammad Aliannejadi, Dimitrios Rafailidis, and Fabio Crestani. 2017. Personalized Keyword Boosting for Venue Suggestion based on Multiple LBSNs. In *ECIR 2017*. Springer.

[5] Corinna Cortes and Vladimir Vapnik. 1995. Support-Vector Networks. *Machine Learning* 20, 3 (1995), 273–297.

[6] Seyyed Hadi Hashemi, Charles L. A. Clarke, Jaap Kamps, Julia Kiseleva, and Ellen M. Voorhees. 2016. Overview of the TREC 2016 Contextual Suggestion Track. In *TREC 2016*. NIST.

[7] Steffen Rendle. 2010. Factorization Machines. In *ICDM 2010*. IEEE.