

# Inverse Reinforcement Learning and Point of Interest Recommendations\*

Discussion Paper

David Massimo, Francesco Ricci

Free University of Bozen-Bolzano, Italy

## Abstract

We here focus on Points of Interest (POIs) Recommender Systems (RSs), aimed at helping users visiting a city to discover new and relevant POIs. RSs are often assessed in offline settings, hence, measuring the system's precision in predicting previously observed user behaviour. However, when deployed, the system produced recommendations are often of limited use, because they lack novelty. We conjecture that this phenomenon is primarily due to the limited capability of RSs in extracting from the observed behaviour general characteristics of POIs that are relevant for different classes of users (tourist types). We compare an Inverse Reinforcement Learning (IRL) based RS algorithm with more traditional Nearest Neighbour and Popularity-based ones. Through an offline evaluation, we show that the nearest neighbour and popularity-based RSs excel in precision (offline) and are perceived as not novel by users of a live-user study. On the contrary, despite a lower offline precision, the IRL-based RS, which learns the preferences of tourists for POIs characteristics, can give a better support to a tourist.

## Keywords

recommender systems, tourism, user behaviour learning, evaluation

## 1. Introduction

This paper focuses on Recommender Systems (RSs) in the tourism domain and summarises the results of Massimo and Ricci on Points of Interest (POIs) recommendation [1].

In [2] is introduced a next POI-visit recommendation strategy, named Q-BASE, which uses Inverse Reinforcement Learning. It consists of three steps. Firstly, it identifies different tourist clusters based on POI-visit sequence observations. Then, by analysing the sequential consumption pattern of clustered users' POI visits, it learns their preferences for POI features and the reward a (generic) user belonging to a cluster obtains in conducting certain POI visits. Finally, Q-BASE computes the state action-value function  $Q$  which tells how much total reward the user will gain if she selects to visit any POI and keeps choosing POIs according to the optimal visit selection policy which is optimal for their cluster. In an offline evaluation [2] Q-BASE has been compared to the sequence aware recommendation strategy SKNN [3]. Given a set of users' POI-visit sequences and a target user (partial) POI-visit sequence, SKNN recommends to the target

---


*IIR2022: 12th Italian Information Retrieval Workshop, June 29 - July 1, 2022, Milan, Italy*

\*This paper presents some of the results published in Massimo, D., Ricci, F. Popularity, novelty and relevance in point of interest recommendation: an experimental analysis. *Inf Technol Tourism* 23, 473–508 (2021).

✉ davmassimo@unibz.it (D. Massimo); fricci@unibz.it (F. Ricci)



© 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

user next POI visits, which are prevalent in the POI visit sequences of the most similar users. An offline evaluation showed that SKNN offers recommendations that more precisely predict the expected user behaviour, but are not novel. Besides, Q-BASE excelled in suggesting novel POI visits that are also more rewarding than SKNN. This comes at the cost of a lower precision. It was conjectured that in an online RS, when Q-BASE is again compared to SKNN, its recommendations will be perceived as relevant because they contain novel POIs with a larger reward. Besides, it was supposed that SKNN's higher precision is due to its bias in suggesting popular items. We therefore investigated two research questions. (*RQ1*) If SKNN achieves higher precision by being biased towards popular items, can Q-BASE be modified, by biasing its recommendations towards more popular items, to achieve a similar precision of SKNN? (*RQ2*) Will online users like the precise recommendations of SKNN more than those generated by Q-BASE, which are more novel and yet relevant?

In order to investigate these questions we also proposed a novel and adaptable RS, called Q-POP PUSH, derived from Q-BASE and that generates recommendations that simultaneously optimise two criteria: the reward of the recommendations (as for Q-BASE) but also their popularity. Q-POP PUSH computes the harmonic means of the scores given by Q-BASE to a POI visit and the popularity of the POI (visit occurrences in the observed data). To weigh the relative importance of Q-BASE original score and the popularity-based score, Q-POP PUSH uses a parameter  $\alpha$ . With  $\alpha > 0.5$  the popularity of the POI-visit has a higher importance, whereas with  $\alpha < 0.5$  the Q-BASE component is weighted more. Equal importance is given with  $\alpha = 0.5$ . We have tested Q-POP PUSH in an offline experiment by comparing its performance with Q-BASE, a popularity baseline and two Nearest Neighbour next-item RSs: SKNN and s-SKNN [3, 4]. Finally, we have assessed the user perception of the recommendations generated by the best performing (offline) models in a user study.

The rest of this paper summarises the offline experiment, the user study and concludes with a discussion.

## 2. Experimental analysis

The offline experiment was conducted along the classical Machine Learning evaluation approach, with train and test sets [5, 6, 7]. The train set is used to train the models<sup>1</sup> and the test set is used for recommendations generation and evaluation. In particular, 70% of a test POI-visit sequence is used to identify the recommendations and the remaining 30% to assess the recommendation performance. We used a dataset of 1663 POI-visit sequences (over 500 POIs) in the Italian city of Florence, done by 1163 anonymous users. The metrics used to assess Q-BASE, Q-POP PUSH, SKNN, s-SKNN and the popularity baseline POP are: reward as in [2]; popularity intended as a proxy of novelty; precision; and similarity of a recommendation list to the list generated by SKNN. Since SKNN is very precise but also affected by a popularity bias, it is interesting to measure how much Q-BASE and Q-POP PUSH deviates from SKNN. All the metrics range in  $[0, 1]$ , where values close to 0 (1) indicate a low (high) performance. For more details about procedures and metrics, we refer to [1].

---

<sup>1</sup>Nearest neighbour models and POP do not need any training, but use train data at test time.

In Table 1 we report the metric values for Top-5 recommendations, obtained in the offline evaluation. SKNN and s-SKNN perform essentially the same on the reward, precision and popularity

**Table 1**

Performance (top-5) of the considered RSs. Best value for each metric is shown in boldface.

	Q-BASE	Q-POP PUSH				SKNN	s-SKNN	POP
		$\alpha = 0.009$	$\alpha = 0.1$	$\alpha = 0.5$	$\alpha = 0.8$			
Reward	<b>0.032*</b>	0.020	-0.001	-0.009	-0.009	-0.010	-0.010	-0.015
Precision	0.045	0.062	0.063	0.060	0.060	<b>0.068</b>	0.063	0.050
Popularity	<b>0.319*</b>	0.517	0.634	0.643	0.643	0.528	0.570	0.733
SimKNN	0.061	0.307	0.441	0.451	0.450	-	<b>0.530</b>	0.352

\* There is a significant difference ( $p < 0.05$ ) between the best performing model and SKNN (two-tailed paired t-test). The test is run for the metrics Rew, Prec and Pop for 5 repeated train-test split.

metrics. Not surprisingly, s-SKNN produces, on average, recommendations lists that overlap up to 53% to the list of SKNN (*SimKNN*). When comparing SKNN to POP, we note that the overlap of the recommendations is quite low, but still, POP has a reasonably good precision, considering the simplicity of the approach. Q-BASE suggests much less popular items than SKNN and also with higher reward.

The analysis of Q-POP PUSH performance allows to address the research question *RQ1*. By introducing a popularity bias to Q-BASE, as it is done in Q-POP PUSH, the generated recommendations become more similar to SKNN. In fact, when  $\alpha$  is increased, the popularity of the recommendations produced by Q-POP PUSH rises, it reaches and passes that of SKNN. However, it is interesting to note that with a rather small value of  $\alpha = 0.009$  Q-POP PUSH obtains a precision of 0.062, which is very close to the precision of SKNN (0.068) and s-SKNN (0.063). With this setting Q-POP PUSH has still a positive reward (0.02), while SKNN and s-SKNN have negative rewards. This means that a small popularity bias in Q-BASE could be beneficial to improve the system’s precision. We must also note that with larger values of  $\alpha$  the reward of Q-POP PUSH is becoming smaller and smaller and approaching that of the two nearest neighbour methods. Clearly, Q-POP PUSH can be tuned to balance two objectives: precision and reward.

To answer the second research question *RQ2* we have designed a live user study aimed at measuring the user’s perceived novelty and appreciation of the recommendations generated by Q-BASE, Q-POP PUSH, with parameter  $\alpha = 0.5$  (to give equal importance to POIs’ popularity and reward) and the SKNN baseline. We have developed an online system to assess the quality of next-POI recommendations offered to users who have already visited some POIs. The system initially asks to declare visited POIs, which are used to create a hypothetical itinerary the user is supposed to have already visited. Then, the system generates next-POI recommendations with the three tested RS algorithms, combines them in a unique list, and asks the user to evaluate them based on a description of the recommended items. The user does not know which algorithm recommends the displayed recommendations. The training data of the online system is the same we have used in the offline study. The user evaluates each POI by marking it with one or more of the following labels: “I already visited it”, “I like it” for a next visit and “I didn’t know it”. The online user study participants were recruited via social media and mailing lists. Out of

202 users that accessed the application, we identified 158 reliable recommendation sessions.

**Table 2**

Probability to evaluate a POI recommendation as visited, novel, liked and both novel and liked.

<i>Recommender System</i>	<i>Visited</i>	<i>Novel</i>	<i>Liked</i>	<i>Liked &amp; Novel</i>
Q-BASE	0.165*	0.517*	0.361*	0.091
Q-POP PUSH	0.245	0.376	0.464	0.076
SKNN	0.238	0.371	0.466	0.082

\* indicates significant difference from the other two RSs perf. (two proportion z-test,  $p < 0.05$ )

Table 2 shows the estimated probability that a user marks as “visited”, “novel”, “liked” or both “liked” and “novel” a POI recommended by a specific RS. Q-BASE recommends POIs that are less likely to have been already visited by the user and more likely to be novel than those suggested by Q-POP PUSH and SKNN. As in the offline experiment, Q-POP PUSH and SKNN performs similarly. Interestingly, Q-BASE suggests fewer POIs that are liked when compared to the other two strategies. Hence, apparently a more precise RS, based on an offline test, also recommends online items that the user will like more. Besides, the obtained results falsify our hypothesis that optimising the reward of a recommendation, as Q-BASE do, will produce recommendations that the user will like more. However, Q-BASE suggests more novel POIs and, interestingly, more recommendations that are both liked and novel (last column in Table 2).

In a successive analysis we computed the probability that a user will like a recommendation given the following three conditions: she knows the POI but has not yet visited it; she has already seen it; the item is novel (unknown). We have derived the following conclusions. The users liked more the novel POI-visit suggestions generated by SKNN and Q-POP PUSH than those produced by Q-BASE. This is due to the tendency of Q-BASE to suggest POI visits that even if they have the properties typically liked by the user, e.g., they are of the same type of the POIs liked by the user, they are also not popular POIs. Hence, these suggested POIs are hard to be appreciated. The conclusion we derived from the user study is that users tend to like more the items they are familiar with, e.g., previously visited items or items that are not novel.

### 3. Discussion

The results of our experiments seem to confirm that RSs that precisely predict the user choices (offline) are also liked most by real users. In our case, this means that Q-POP PUSH and SKNN are better RSs than Q-BASE. Our explanation of this result is that both high offline precision and large probability of liked recommendations (online) are influenced by the popularity of the recommended items. In fact, these popular items are often in the users’ test sets, and users are likely to be familiar with them.

Despite its lower offline precision performance and a lower extent of liked recommendations in the user study, Q-BASE is the RS that may better accomplish the true goal of a tourism RS: it suggests more next-POIs that are both liked and novel. So, by optimising the reward Q-BASE is capable of discovering novel items that are also appreciated (when users are able to assess them).

## References

- [1] D. Massimo, F. Ricci, Popularity, novelty and relevance in point of interest recommendation: an experimental analysis, *J. Inf. Technol. Tour.* 23 (2021) 473–508. URL: <https://doi.org/10.1007/s40558-021-00214-5>. doi:10.1007/s40558-021-00214-5.
- [2] D. Massimo, F. Ricci, Harnessing a generalised user behaviour model for next-poi recommendation, in: S. Pera, M. D. Ekstrand, X. Amatriain, J. O'Donovan (Eds.), *Proceedings of the 12th ACM Conference on Recommender Systems, RecSys 2018, Vancouver, BC, Canada, October 2-7, 2018*, ACM, 2018, pp. 402–406.
- [3] D. Jannach, I. Kamehkhosh, L. Lerche, Leveraging multi-dimensional user models for personalized next-track music recommendation, in: A. Seffah, B. Penzenstadler, C. Alves, X. Peng (Eds.), *Proceedings of the Symposium on Applied Computing, SAC 2017, Marrakech, Morocco, April 3-7, 2017*, ACM, 2017, pp. 1635–1642. URL: <https://doi.org/10.1145/3019612.3019756>. doi:10.1145/3019612.3019756.
- [4] M. Ludewig, D. Jannach, Evaluation of session-based recommendation algorithms, *User Model. User-Adapt. Interact.* 28 (2018) 331–390.
- [5] A. Bellogín, A. Said, *Recommender Systems Evaluation*, Springer New York, New York, NY, 2018, pp. 2095–2112. URL: [https://doi.org/10.1007/978-1-4939-7131-2\\_110162](https://doi.org/10.1007/978-1-4939-7131-2_110162). doi:10.1007/978-1-4939-7131-2\_110162.
- [6] J. L. Herlocker, J. A. Konstan, L. G. Terveen, J. T. Riedl, Evaluating collaborative filtering recommender systems, *ACM Transactions on Information Systems (TOIS)* 22 (2004) 5–53.
- [7] P. Cremonesi, Y. Koren, R. Turrin, Performance of recommender algorithms on top-n recommendation tasks, in: *Proceedings of the Fourth ACM Conference on Recommender Systems, RecSys '10, Association for Computing Machinery, New York, NY, USA, 2010*, p. 39–46. URL: <https://doi.org/10.1145/1864708.1864721>. doi:10.1145/1864708.1864721.