

# Personalizing Type-Based Facet Ranking Using BERT Embeddings

Esraa ALI <sup>a,1</sup>, Annalina CAPUTO <sup>b,2</sup>, Séamus LAWLESS <sup>a</sup> and Owen CONLAN <sup>a</sup>

<sup>a</sup> ADAPT Centre, School of Computer Science and Statistics, Trinity College Dublin

<sup>b</sup> ADAPT Centre, School of Computing, Dublin City University

## Abstract.

In Faceted Search Systems (FSS), users navigate the information space through facets, which are attributes or meta-data that describe the underlying content of the collection. Type-based facets (aka t-facets) help explore the categories associated with the searched objects in structured information space. This work investigates how personalizing t-facet ranking can minimize user effort to reach the intended search target. We propose a lightweight personalisation method based on Vector Space Model (VSM) for ranking the t-facet hierarchy in two steps. The first step scores each individual leaf-node t-facet by computing the similarity between the t-facet BERT embedding and the user profile vector. In this model, the user's profile is expressed in a category space through vectors that capture the users' past preferences. In the second step, this score is used to re-order and select the sub-tree to present to the user. The final ranked tree reflects the t-facet relevance both to the query and the user profile. Through the use of embeddings, the proposed method effectively handles unseen facets without adding extra processing to the FSS. The effectiveness of the proposed approach is measured by the user effort required to retrieve the sought item when using the ranked facets. The approach outperformed existing personalization baselines.

**Keywords.** Type-based Facets, Faceted Search, Personalization.

## 1. Introduction

Faceted Search Systems (FSS) have become one of the main search interfaces used in vertical search systems, offering users meaningful facets to refine their search query and narrow down the results. As the magnitude of data in a collection increases, the number of facets becomes impractical to display on a single page. Ranking the top facets is required as it assists the searcher in locating the target document with minimum effort.

When the information space is structured, type-based facets (t-facets) are extracted from the types of objects (e.g. values of `isA` or `rdfs:type` relationships) [1]. Hierarchical taxonomies of types are derived from ontologies by exploiting the `subClassOf` relationships. When the t-facets belong to a large multilevel hierarchy of categories, it is particularly important to prioritize the most relevant t-facets that can help the user filtering the results by type.

---

<sup>1</sup>Corresponding Author: E-mail: [esraa.ali@adaptcentre.ie](mailto:esraa.ali@adaptcentre.ie)

<sup>2</sup>Corresponding Author: E-mail: [annalina.caputo@adaptcentre.ie](mailto:annalina.caputo@adaptcentre.ie)

In this work, we analyse the role of personalization in t-facet ranking in isolation from other FSS aspects. Existing facet ranking methods rely on attribute frequencies, navigation cost models, textual queries or click logs to order the facets [1]. This study contributes to the research in this area by introducing a light and effective ranking algorithm for type-based facets.

The algorithm exploits the user's past preferences to build a vector which represents the user profile. T-facets are scored according to their similarity with user profile by exploiting their BERT embeddings. The proposed approach works in two steps. First, it generates personalized relevance scores for each t-facet in the last level of the taxonomy. Then, the second stage aggregates these scores to re-arrange the ancestor t-facets and rebuild the final t-facet tree to be rendered to the user. Our experiment aims at answering the following research question: **RQ**: *Does personalizing the t-facet ranking using BERT embeddings minimize users effort to fulfil their search needs?*

The implemented approach is evaluated using the TREC Contextual Suggestion (TREC-CS) track dataset [2]. TREC-CS is a personalized Point-Of-Interest (POI) recommendation task. We solve the POI suggestion problem by ranking the types of venues as t-facets. In our evaluation, we measure the extent to which this ranked tree minimizes the user effort to reach the first relevant POI.

## 2. Personalized Facet Ranking Related Research

Several approaches have been proposed in the literature to solve the problem of personalized facet ranking that make use of individual user models, collaborative filtering (CF), or a mixture between the two. Factic [3] is a FSS that personalizes by building models from semantic usage logs. Several layers of user adaption are implemented and integrated with different weights to enhance the facet relevance model. Koren et al. [4] suggested a CF approach by leveraging explicit user feedback about the facets, which is used to build a facet relevance model for individuals. They also use the aggregated facet ratings to build a collaborative model for the new users in order to provide initial good facets in absence of a user profile. The Adaptive Twitter search system generates user models from Twitter to personalize facet ordering [5]. The user model contains entities extracted from the user's tweets. The facets are weighted higher if they exist in the user profile. Le et al. [6] also collects user profile from social networks. The profile is learned from user activities and preferences using a tf-idf feature vector model. Important facets are then highlighted through a matching with the model. A personalized ranking based on CF features was suggested by Chantamunee et. al [7, 8]. They used user ratings and Matrix Factorization via SVM and Autoencoders to learn facet ranks.

All the aforementioned approaches do not address the special case of t-facets. Sah and Wade [9] employ session-based user interaction to personalize t-facets. When the searcher selects a t-facet, the system re-organizes the other t-facets according to their similarity to the selected one. Neither the user interests nor the hierarchical nature of the t-facets are considered by this method. Our approach exploits VSM to build user profile, which employs users' historical ratings to infer their preferred t-facets, an unexplored area in literature.

### 3. Proposed Approach

Our method works in the context of personalized venue search. When a user submits a query, the underlying search engine retrieves a relevant set of venues for it<sup>3</sup>. In order to achieve query relevance, our method works on this set by collecting the t-facets associated with the retrieved venues. The proposed t-facet ranking approach consists of two steps. The first step assigns a relevance score to each t-facet leaf node. The input to this step are the retrieved venues with their relevancy score, the t-facets to which they belong, and the user profile. The second step constructs the final t-facet tree to be displayed to the user. The input to this step are both the score for each t-facet (from the first step) and the original hierarchical taxonomy from which the t-facets are derived. The output of the t-facet ranking is a sub-tree, which contains the ordered set of relevant t-facets.

*Step 1: T-Facet Scoring using BERT vectors.* At this step, we establish relevance between the t-facet and the user profile through their vector representation. Each t-facet  $f_i$  is represented by a BERT embedding  $\vec{f}_i$ . We employ the t-facet's label to generate its corresponding BERT embedding using pre-trained BERT model<sup>4</sup>. BERT provides a meaningful vector representation of text and has been proven effective in many IR tasks. It is particularly useful in this task as it semantically embeds the meaning of the type in the vector. This is important in handling new and unseen t-facet which did not exist in the user profile before. In this case the model will handle them based on their similarity with existing t-facets. In our early experiments we also experimented word2vec models but BERT-based models gave better results. However we acknowledge that other semantic-based embedding can be used in this approach and improve the results.

The vector representing the user profile is obtained using the Rocchio formula, which is a classical approach to relevance feedback in IR. Since in our case we only have the user feedback (i.e. ratings) at POI level, we assume that the rating can also be transferred to the types associated with the POI. Based on this assumption, we represent the user profile through a vector combining positive, negative and neutral preferences (i.e. rated POI types). We define the positive (pos.) user vector  $\vec{u}_i^{pos}$  as the average of t-facet vectors rated positively by the user  $u_i$ :

$$\vec{u}_i^{pos} = \frac{\sum_{j=1}^{|F_u^{pos}|} \vec{f}_j \times w_{f_j^{pos}}}{|F_u^{pos}|} \quad (1)$$

Where  $F_u^{pos}$  is the set of t-facets rated positively by the user. Each vector is weighted by  $w_{f_j^{pos}}$  (see eq 2). The weight averages the probability that this specific user will rate the t-facet as positive  $Pr(f_j^{pos}|u_i)$  and the probability that any user in the system will rate the t-facet as positive  $Pr(f_j^{pos}|U)$ . In absence of any personal preference only the general probability will be used. The value  $w_{f_j^{pos}}$  affects how each t-facet contributes to the final positive profile vector.

<sup>3</sup>How the venue ranking is performed is outside scope of this research.

<sup>4</sup><https://spacy.io/universe/project/spacy-transformers>, model used: en\_trf\_bertbaseuncased\_lg

$$w_{f_j^{pos}} = \begin{cases} \gamma \times Pr(f_j^{pos}|U) + (1 - \gamma) \times Pr(f_j^{pos}|u) & , \text{if } Pr(f_j^{pos}|u) > 0 \\ Pr(f_j^{pos}|U) & , \text{otherwise} \end{cases} \quad (2)$$

The parameter  $\gamma \in [0, 1]$ , is used to balance between individual user feedback versus general population feedback. In a similar manner, the negative (neg.) and the neutral (neu.) user vectors are computed. In order to model the final vector representation of the user's interest, we experimented with two versions of the Rocchio formula. The first (*Rocchio-BERT*) is the traditional version computed according to equation 3.

$$\vec{u}_i = \alpha \times \vec{u}_i^{neu} + \beta \times \vec{u}_i^{pos} - \lambda \times \vec{u}_i^{neg} \quad (3)$$

The contribution of each user vector is regulated through the weights  $\alpha, \beta, \lambda \in [0, 1]$ . The second (*Ortho-BERT*) is a modified version of Rocchio formula proposed by [10], in which instead of subtracting the negative vector they add the user negative vector's orthogonal complement ( $\vec{u}_i^{neg^\perp}$ ). Equation 4 illustrates the computation.

$$\vec{u}_i = \alpha \times \vec{u}_i^{neu} + \beta \times \vec{u}_i^{pos} + \lambda \times \vec{u}_i^{neg^\perp} \quad (4)$$

The orthogonal complement is obtained using Gram-Schmidt process. The weights are the same as equation 3. Finally, the query t-facets are ranked according to the cosine similarity score between the user profile vector  $\vec{u}_i$  and the the t-facet vector  $\vec{f}_j$ .

**Step 2: T-Facet Tree Building** The tree construction algorithm re-orders the original taxonomy tree by using the generated scores from the previous step. It follows a bottom-up approach where the t-facets at the lower level in the taxonomy are sorted first, then it proceeds by sorting all the ancestors of those t-facets, and so on up to the root of the hierarchy. To build a final t-facet tree with  $v$  levels, we adopted a *fixed level* strategy [11]. The strategy respects the original taxonomy hierarchy and uses a predefined fixed page size for each t-facet level. It starts by grouping t-facets at level- $v$  by their parent. Then, it sorts the parent nodes at level- $(v - 1)$  by aggregating the scores of their top  $k$  children, the children are ordered by their relevance score generated in step 1, and so on up to level-1. We use *Max.* aggregation function to keep the top ranked t-facet at the top of the final tree.

#### 4. Experimental Results

**Experimental setup.** Our approach is evaluated on TREC-CS 2016 dataset [2]. The t-facet taxonomy is derived from the Foursquare venue category hierarchy<sup>5</sup>. To link as much Foursquare venues to TREC-CS POIs as possible, we complement the original data with three Foursquare supplementary datasets from [12, 13] and our own crawled POIs. The final dataset has 58 requests and an average of 208 t-facets per request to be ranked. Only the first two levels of the taxonomy are included. The document search engine implements BM25 with NDCG value of 0.4023, the query is formed by combining user tags weighed by their most common rating. The existence of relevance judgments makes

<sup>5</sup><https://developer.foursquare.com/docs/resources/categories>, version: 20180323

it possible to evaluate our approach against a well established ground-truth. We follow the strategy used in Faceted Search task of INEX 2011 Data-Centric Track [14]. We report two metrics suggested by the organizers. The number of actions (#Actions) counts how many clicks the user has to perform on the ranked facets in order to reach the first relevant document in the top 5 results. The faceted scan (F-Scan) measures the user’s effort to scan facets and documents until user reach the same document. It is calculated as the sum of the t-facet rank in the tree plus the document position in the list after being filtered using this t-facet. We focus on these metrics as a proxy for user’s effort, which will help in answering our research question. To chose the optimal weights for equations 4 and 3, we used hyper-parameter tuning with range from 0 to 1 and step of 0.25. We report results for the Rocchio-BERT and Ortho-BERT methods in addition to KNN-BERT, which uses a trained KNN model for each user to predict the user’s preference to a given t-facet, with  $k = 1$ . All scoring results are reported by adopting the Fixed Level-Max strategy, with level-1 and level-2 page size set to three.

*Results.* Table 1 summarizes our results; the first 3 rows shows our BERT based scoring methods. The Rocchio-BERT method outperformed both Ortho-BERT and KNN-BERT methods. The normalization step by Gram-Schmidt process used in Ortho-BERT neutralises the personalized negative profile weight, which negatively affects the results. Rocchio-BERT achieved best results with  $\alpha = 0$ ,  $\beta \in \{0.25, 0.5, 0.75, 1\}$ , and  $\lambda = 0$ , which means that the positive vector profile is the key component in the formula.  $\gamma = 0$ , which means that the person probability is favoured over the global probability in the t-facets weighting(2).

The second part of Table 1 reports the results of other existing personalization methods. As these methods do not handle the hierarchical nature of the t-facets, we use them as scoring methods with the Fixed Level-Max strategy. Rocchio-BERT still results in the minimum #Actions and F-scan when compared with these models. In addition, the proposed VSM approach has other advantages. It provides a light yet effective personalized ranking. The user profile vectors can be pre-computed and stored offline, reducing the computation at retrieval time of the cosine distance between the query t-facets and the user profile. This can be further optimized by pre-computing a distance matrix for the user and all t-facets in the FSS. User profiles on the other hand, can be updated offline as the user rates new venues. Optimizing this step will reduce the total time needed by FSS to populate its final results page.

## 5. Conclusions

This work presented a two-step t-facet ranking approach employing BERT embeddings to personalize the t-facet ranking. The first step assigns score to t-facets. The second step uses the score to re-arrange and build the final t-facet tree. To personalize the scores, we

Table 1.: Results for our scoring models using Fixed-level(Max) strategy.

Scoring Method	F-Scan	#Actions
KNN-BERT	4.825	1.667
Ortho-BERT	3.754	1.421
Rochio-BERT	<b>3.281</b>	<b>1.281</b>
Prob. Scoring [11]	3.456	1.333
MF-SVM [7]	3.912	1.491
Most Prob. (Person) [4]	3.737	1.6142
Most Prob. (Collab) [4]	3.351	1.333

explored several methods based on vector space model. They showed promising results when combined with the personal preferences of the users. We plan to experiment with additional POI suggestion datasets and explore learning to rank methods for the t-facet scoring step.

### Acknowledgements

This work was supported by the ADAPT Centre, funded by Science Foundation Ireland Research Centres Programme (Grant 13/RC/2106; 13/RC/2106\_P2) and co-funded by the European Regional Development Fund.

### References

- [1] Yannis Tzitzikas, Nikos Manolis, and Panagiotis Papadakos. Faceted exploration of rdf/s datasets: a survey. *Journal of Intelligent Information Systems*, 2017.
- [2] Seyyed Hadi Hashemi, Charles LA Clarke, Jaap Kamps, Julia Kiseleva, and Ellen M Voorhees. Overview of the trec 2016 contextual suggestion track. In *TREC*, 2016.
- [3] Michal Tvarožek and Mária Bieliková. Factic: personalized exploratory search in the semantic web. In *ICWE*. Springer, 2010.
- [4] Jonathan Koren, Yi Zhang, and Xue Liu. Personalized interactive faceted search. In *WWW*. ACM, 2008.
- [5] Fabian Abel, Ilknur Celik, Geert-Jan Houben, and Patrick Siehdnel. Leveraging the semantics of tweets for adaptive faceted search on twitter. *The Semantic Web*, 2011.
- [6] T. Le, B. Vo, and T. H. Duong. Personalized facets for semantic search using linked open data with social networks. In *IBICA*, 2012.
- [7] Siripinyo Chantamunee, Kok Wai Wong, and Chun Che Fung. Collaborative filtering for personalised facet selection. In *IAIT*. ACM, 2018.
- [8] Siripinyo Chantamunee, Kok Wai Wong, and Chun Che Fung. An exploration of user–facet interaction in collaborative-based personalized multiple facet selection. *Knowledge-Based Systems*, 209:106444, 2020. ISSN 0950-7051. doi: <https://doi.org/10.1016/j.knosys.2020.106444>. URL <http://www.sciencedirect.com/science/article/pii/S0950705120305736>.
- [9] Melike Sah and Vincent Wade. Personalized concept-based search and exploration on the web of data using results categorization. In *The Semantic Web:2013*.
- [10] Pierpaolo Basile, Annalina Caputo, and Giovanni Semeraro. Negation for document re-ranking in ad-hoc retrieval. In *ICTIR 2011*.
- [11] Séamus Lawless Esraa Ali, Annalina Caputo and Owen Conlan. A probabilistic approach to personalize type-based facet ranking for poi suggestion. In *ICWE 2021*.
- [12] Mohammad Aliannejadi, Ida Mele, and Fabio Crestani. A cross-platform collection for contextual suggestion. In *SIGIR*. ACM, 2017.
- [13] Mostafa Bayomi and Séamus Lawless. Adapt\_tcd: An ontology-based context aware approach for contextual suggestion. In *TREC*, 2016.
- [14] Qiuyue Wang, Georgina Ramírez, Maarten Marx, Martin Theobald, and Jaap Kamps. Overview of the INEX 2011 Data-Centric Track. In *Focused Retrieval of Content and Structure*. Springer, 2012.