Enhancing explanations in recommender systems with knowledge graphs

Vincent Lully, Philippe Laublet, Milan Stankovic, Filip Radulovic

Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 137 (2018) 211–222

Procedia Computer Science

www.elsevier.com/locate/procedia

SEMANTiCS 2018 – 14th International Conference on Semantic Systems

Enhancing explanations in recommender systems with knowledge graphs

Vincent Lully[a,b,*] , Philippe Laublet[a], Milan Stankovic[a,b], Filip Radulovic[b]

[a]*Sorbonne Université, 28 Rue Serpente, 75006, Paris, France*
[b]*Sépage, 38 Avenue de l'Opéra, 75002, Paris, France*

Abstract

Recommender systems are becoming must-have facilities on e-commerce websites to alleviate information overload and to improve user experience. One important component of such systems is the explanations of the recommendations. Existing explanation approaches have been classified by style and the classes are aligned with the ones for recommendation approaches, such as collaborative-based and content-based. Thanks to the semantically interconnected data, knowledge graphs have been boosting the development of content-based explanation approaches. However, most approaches focus on the exploitation of the structured semantic data to which recommended items are linked (e.g. actor, director, genre for movies). In this paper, we address the under-studied problem of leveraging knowledge graphs to explain the recommendations with items' unstructured textual description data. We point out 3 shortcomings of the state of the art entity-based explanation approach: absence of entity filtering, lack of intelligibility and poor user-friendliness. Accordingly, 3 novel approaches are proposed to alleviate these shortcomings. The first approach leverages a DBpedia category tree for filtering out incorrect and irrelevant entities. The second approach increases the intelligibility of entities with the classes of an integrated ontology (DBpedia, schema.org and YAGO). The third approach explains the recommendations with the best sentences from the textual descriptions selected by means of the entities. We showcase our approaches within a tourist tour recommendation explanation scenario and present a thorough face-to-face user study with a real commercial dataset containing 1310 tours in 106 countries. We showed the advantages of the proposed explanation approaches on five quality aspects: intelligibility, effectiveness, efficiency, relevance and satisfaction.

Peer-review under responsibility of the scientific committee of the SEMANTiCS 2018 – 14th International Conference on Semantic Systems.

Keywords: Recommender system, explanation, knowledge graph, hierarchical data, category, ontology, DBpedia, natural language, e-tourism

* Corresponding author. *E-mail address:* vincent.lully@sorbonne-universite.fr

1877-0509
Peer-review under responsibility of the scientific committee of the SEMANTiCS 2018 – 14th International Conference on Semantic Systems.
10.1016/j.procs.2018.09.020

1. Introduction

Recommender systems [1] are becoming must-have facilities on e-commerce websites to alleviate information overload and to improve user experience. Apart from the traditional top-N item recommendation functionality, another important functionality of such systems is the explanations of the recommendations. An explanation is sometimes a justification of why items have been recommended, while sometimes an item description which helps users to understand the qualities of the item well enough to decide whether it is relevant for them or not [2]. Like the classification of recommendation approaches, explanation approaches are also classified to several categories such as demographic-based, collaborative-based, content-based etc.

Since several years, with large-scale publicly available interconnected semantic data, knowledge graphs have boosted content-based recommendation approaches [3] in various domains [4-6]. Some systems provide content-based explanations for the recommendations of movies, music artists and books [4, 7, 8]. Their approaches explain with the structured semantic metadata in knowledge graphs to which the recommended items are linked (e.g. actor, director, genre).

In this paper, we address the under-studied problem of leveraging knowledge graphs to explain the recommendations with items' unstructured textual description data. The problem concerns items like books, news articles, touristic tours which often have rich textual contents valuable for explanation use. The state of the art content-based approach consists of explaining with a certain number of words or knowledge graph entities extracted from the texts and ranked with a certain method [9-11].

We showcase and evaluate our novel propositions within the scenario of explaining tourist tour recommendations. A touristic tour typically contains a title, a period, a price and a detailed textual description of the itinerary. In Fig.1., we give an example of a tourist tour to Bavaria, Switzerland and Austria proposed by the agency "trailfinders[1]". Imagine the tour in Fig.1. is recommended, the state of the art approach may explain it by listing several entities like "dbr:Marienplatz", "dbr:Neuschwanstein_Castle", "dbr:Passion_Play".

We point out 3 shortcomings of the state of the art entity-based explanation approach:

1. **Absence of entity filtering**. Current approaches rely on off-the-shelf entity linking tools like DBpedia Spotlight and Babelfy which cannot achieve 100% accuracy for the moment [12]. The outputted entities are not filtered. Incorrect and irrelevant entities may be extracted and selected for explanations. From the sentence "Be sure to always have the battery on your camera to capture the beautiful scenery.", a tool may extract the wrong entity "dbr:Artillery_battery", another tool may extract the correct entity "dbr:Battery_pack" but irrelevant for the current recommendation domain.
2. **Lack of intelligibility**. For many people, travelling is sometimes about discovering new cultures and civilizations. They might not be very familiar with the tours that they are browsing or being recommended and thus might have difficulties understanding the mere entities. In the example of Fig.1., the meaning of the entity "dbr:Passion_Play" may not be obvious for everyone.
3. **Poor user-friendliness**. This style of explanation consisting of a mere listing of entities may appear to be not very user-friendly.

Accordingly, we try to alleviate these 3 shortcomings with 3 approaches:

1. We leverage a DBpedia category tree for filtering out incorrect and irrelevant entities.
2. We increase the intelligibility of entities with an integrated ontology (DBpedia, schema.org and YAGO).
3. We explain the recommendations with the best sentences from the textual descriptions selected by means of the entities.

The rest of the paper is organised as follows. In Section 2, we discuss some related work. In Section 3, we detail our entity filtering approach. Section 4 describes our solution for the lack of intelligibility. In Section 5, we present our sentence-based explanation approach. In Section 6, we report the evaluation. Section 7 concludes the paper.

[1] https://www.trailfinders.com/

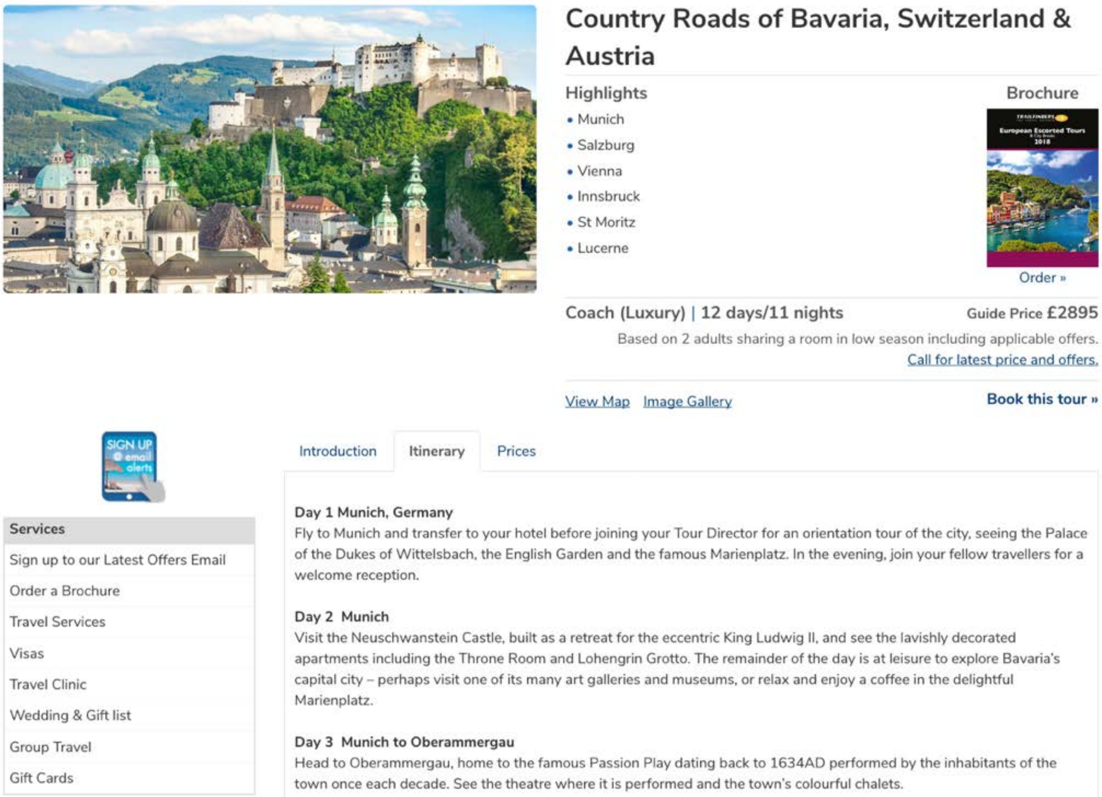

Fig. 1. Screenshot of a touristic tour on “trailfinders.com”

2. State of the art

Given that the novel propositions of our work concern content-based explanation of touristic tour recommendations, we present some related work in 3 directions:

1. approaches using structured data (knowledge graphs) and semi-structured data (processed folksonomy)
2. approaches using unstructured data (textual description data)
3. approaches explaining travel recommendations

2.1. Explaining recommendations with (semi-)structured data

In [4], the author explains music artist recommendations with features from DBpedia. The provided explanations aim to answer the question “*Why are the recommended music artist is related to a music artist I like?*”. They explain by showing human-readable labels of the property and their values. For example, for a user who likes Johnny Cash, the recommendation of Elvis Presley can be explained by “There are shared ‘associated acts’ between Johnny Cash and Elvis Presley: Charlie McCoy, Buddy Harman, The Jordanaires.”

In [13], the authors explain travel destination (city) recommendations with DBpedia entities coming from a feature selection and a travel folksonomy entity mapping. For a set of recommended cities, they search for the most common features shared by them and diversify with regards to the properties with link cities and features. For a user who likes Rome, Florence and Amsterdam, the system recommends The Hague, Haarlem, Naples, Milan and Turin accompanied by five explanation features *dbc:Clothing*, *dbr:Food*, *dbr:David_de_Haen*, *dbr:Italy* and *dbr:History*.

In [5], the authors search for top properties which link directly or indirectly recommended movies and the ones liked by the user. The found top properties are used to fill a template-based structure to generate the final explanation. All properties involved in the movie domain are mapped to a natural language expression, for example, the property *dbp:starring*, which links movies to actors, is mapped to *starred by*. For a user who likes the movies Da Vinci Code, Saving Private Ryan, the recommendation of Cloud Atlas can be explained by "I recommend you Cloud Atlas since you often like movies starred by Tom Hanks as Da Vinci Code and Saving Private Ryan. Moreover, I recommend it because you sometimes like Dystopian Movies as The Matrix and American Epic Films as Saving Private Ryan". This work is extended in [6] and applied also on music artists and books.

Some other efforts focus on the movie domain and rely on the use of the tag genome dataset. This dataset is made by the MovieLens community. The tag genome dataset contains 11 million computed tag-movie relevance scores from a pool of 1,100 tags (e.g. comedy, police, space) applied to 10,000 movies.

In [14], the authors present the *tagsplanations* system. For each user, based on his or her past movie ratings, the system calculates a preference score for each tag. Given a recommended movie, for each tag, the system knows a relevance score and a preference score. The recommendation of the movie Rushmore can be explained by the most relevant tags: wes anderson, deadpan, quirky, witty, off-beat comedy, notable soundtrack, stylized. The authors compare four different explanation interfaces: *RelSort* and *PrefSort* where both relevance and preferences scores are displayed but tags are either sorted by relevance or by preference, *RelOnly* and *PrefOnly* where only relevance or preference scores are displayed.

Similar to [14], in [15], the authors introduce another way of visualizing explanations named tag cloud. One tag cloud is generated for one recommended movie. Within the tag cloud, the tags are arranged in alphabetical order. Larger font sizes indicate higher relevance score of a term. The tag cloud can be non-personalised or personalied. In the non-personalised version, the score is determined by the number of items a tag is applied to an item. In the personalised version, the sorting and the font size remain the same, the preferences of a user are considered, depending on the sentiment polarity (positive, neutral, negative) of a user towards tags, tags are displayed with different colors.

In [16], the authors explain movie recommendations in natural language for MovieLens users. All possible explanations are pre-constructed by using the tag genome dataset and by asking crowd workers. For each movie, they select the top 20 most relevance tags. Then they cluster the tags and ask crowd workers to refine and label each cluster (a label being one of the tags in a cluster, they name a label a key topical aspect). After that, crowd workers are asked to select, from a list of movie reviews containing a key topical aspect, the one that best promote the movie on the aspect. Finally, crowd workers are asked to edit the review to make it more appropriate for an explanation. The explanation is personalised. It is about an aspect that a user likes the most according to his/her past movie ratings. For example, for a user who likes very much movies which are tagged with *space*, the recommendation of the movie Gravity can be "From your MovieLens profile it seems that you prefer movies tagged as space, this movie takes you in space and it feels claustrophobic to be there. It keeps you on the edge of your seat the whole time.".

2.2. Explaining recommendations with unstructured data

In [9], the authors suggest explaining the recommendation of a news article (that they call a story) with the template "This story received a [high | low] relevance score, because it contains the words fl, f2 and f3." In [10], the authors present a system named of LIBRA (Learning Intelligent Book Recommending Agent). They build a book database by crawling book data from Amazon's website. Each type of data (title, authors, synopses, reviews, comments etc) is put in a separate slot. They extract keywords for all slots. They use a Bayesian learning algorithm to build a user profile which consists of slots, words and strength scores. Provided a recommended book, the explanation aims to answer the question "What is it about the book that speaks to the user's interests?" by a list of keywords sorted by their strength scores. For example, the recommendation of the book "The Fabric of Reality" by David Deutsch can be explained by "multiverse – 75.12", "universes – 25.08", "reality – 22.96" etc. These explanations are evaluated in a later work [2] against collaborative-based ones and are shown to perform better on the effectiveness of convincing users to adopt recommendations and the helpfulness on making more accurate decisions. The development of Semantic Web technologies has seen not only the rapid growth of knowledge graphs in many domains but also the advent of entity linking tools [12] which map mentions in texts written in natural language to corresponding knowledge graph entities. In [11], the authors make use of entities to explain news article recommendations. They extract knowledge graph entities from all news articles. For a user, the system compares the entities in the news articles already read by the user and in the one recommended to him/her. The system explains with entities selected with different strategies, for example selecting the shared entities, or selecting distinct entities to show the differences between the read articles and the current recommended article. To the best of our knowledge, the entity-based explanation is the state-of-the-art approach in this under-studied area.

2.3. Explaining travel recommendations

Despite a large body of recommender systems in the travel domain [17, 18], especially on the recommendation of points of interest, only a few systems provide explanations.

Some of them use demographic-based approaches. In [19], the authors explain the recommendation of Lingotto by “For children, it is much eye-catching, it requires low background knowledge... For yourself it is much eye-catching and it has high historical value. For the impaired...”. In [20], the authors provide an example “The spa resort VIVAT is very apt for families with children. They are offering for instance day nursery and animators for children”. In order to function correctly, such systems need to have a mapping between the recommended items and demographic characteristics, and also know this information about the user.

The listing of features is a common practice on today’s travel websites. In Fig.2., we show some examples that we found on several websites. However, many upstream manual efforts need to be made in order to create these displayed features. In our work, we extract semantic entity features from the textual descriptions and try to alleviate the 3 shortcomings mentioned in the introduction with help of knowledge graphs.

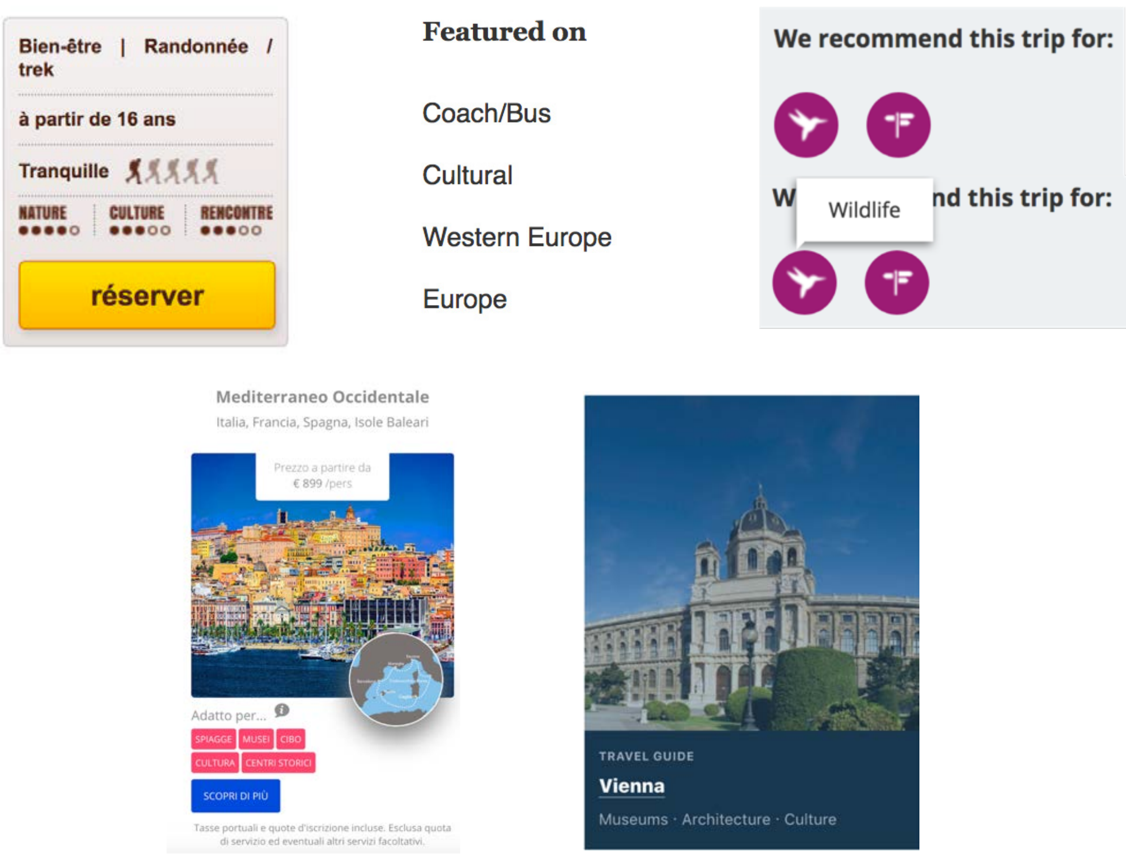

Fig. 2. Screenshots on multiple travel websites, respectively: Nomade Aventure, Tour Radar, Audely Travel, Costa Crociere, Booking

3. Entity filtering with a DBpedia category tree

In DBpedia, entities are linked to categories with the property "dct:subject". The categories reflect the subjects of the entities. The main idea behind our entity filtering method is to retain entities whose subjects are relevant for the recommendation domain. For example, the entity "dbr:Musée_d'Orsay" is linked to the category "dbc:Art_museums_and_galleries_in_Paris". The category is relevant to the travel domain so the entity should be retained.

Our filtering method requires the knowledge about the relevance of the categories with regards to the travel domain. To avoid manually annotating all the categories, we make advantage of the hierarchical relationships between categories (skos:broader). We constructed a DBpedia category tree by following the steps described in [21]. As shown in Table 1, our tree contains 1,023,155 categories spread over 15 levels and has as root category "dbc:Main_topic_classifications". In Fig.3, we show the distribution of categories by level of depth in the tree.

In the current work, we manually annotated the relevance of the 43 categories at level 2 (just under the root category). 12 of them are annotated as relevant to the travel domain, such as "dbc:Arts", "dbc:Culture" and "dbc:Nature". Their relevance is propagated to all their sub-categories. As we can see in Fig.3, there are 1183 categories at level 3. It remains feasible to annotate this number of categories if we want to have a more fine-grained filtering. For example, among the 21 sub-categories of "dbc:Culture" at level 3, we may annotate "dbc:Natural_monuments" as relevant and "dbc:Nature_writers" as irrelevant.

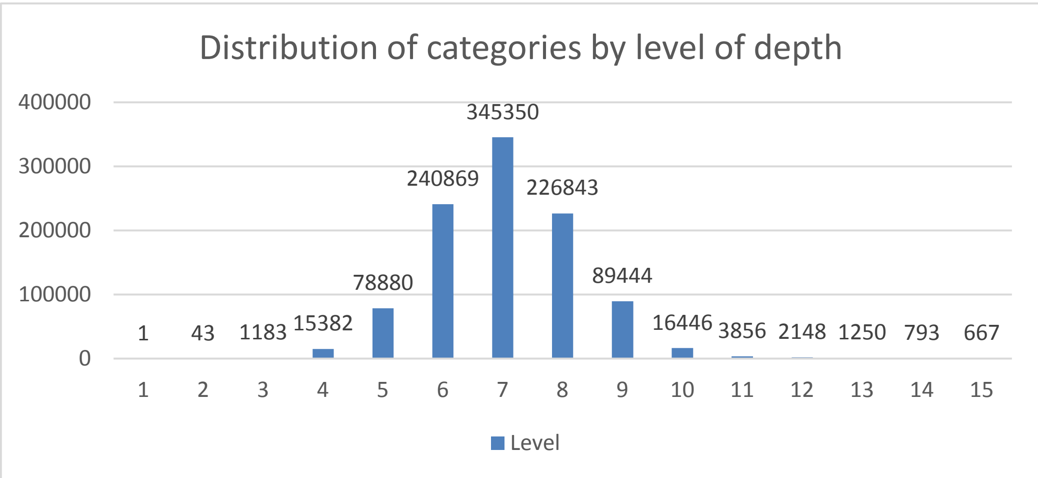

Fig. 3. Distribution of categories by level of depth in DBpedia

At the entity filtering phase, we query DBpedia endpoint to retrieve all the categories to which the entity is linked. If at least one category is relevant, then the entity is relevant. Otherwise, the entity should be filtered out.

4. Increasing the intelligibility with ontology class

Our approach consists of providing a hint to help users better understand what an entity is about. To this end, we make use of another important knowledge graph hierarchy namely the ontology class taxonomy. Ontology classes are abstract objects that are defined by values of aspects that are constraints for being member of the classes. We use more precisely an integrated ontology constructed for the work [22]. It integrates 447,250 classes from DBpedia, YAGO (whose ontology integrates Wordnet synsets and conceptual Wikipedia categories [23]) and schema.org, rooted on *owl:Thing* and with a depth of 19 (Table 1). We are convinced these ontology classes can serve as helpful hints for entities.

Table 1. Statistics about the two knowledge graph hierarchies used in Sections 3 and 4

	Integrated ontology (DBpedia, YAGO, schema.org)	DBpedia category tree
root	owl:Thing	dbc:Main_topic_classifications
depth	19	15
# classes/categories	447,250	1,023,155

Given an entity, we retrieve the ontology class which is the deepest in the hierarchy. The idea is that a deeper class conveys more detailed information and is thus more appropriate as a hint in our use case. If multiple classes have the same level of depth, for practical reasons, we select the one with the shortest string length to make it easier to display in a recommendation banner. In Fig.4, we show a part of the categories and ontology classes to which the entity "dbr:Saimaa" is linked. In this case, our method would select as hint "yago:LakesOfFinland".

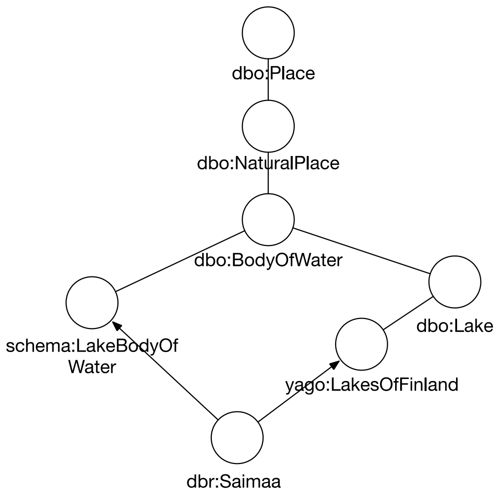

Fig. 4. A part of the ontology classes to which the entity "dbr:Saimaa" is linked

5. Better user-friendliness with sentence-based explanations

To augment the user-friendliness, we propose a sentence-based explanation approach built on two intuitions:

1. Explaining with sentences in natural language is more user-friendly than a mere listing of entities.
2. The more important entities a sentence contains, the stronger the sentence's explanation ability is.

Our approach consists of 3 main steps:

1. Entity extraction and scoring. Given an item dataset, we use DBpedia Spotlight to extract entities from the textual description of each item. We then apply the widely-adopted TF-IDF method to calculate a score for all the entities and all the items.
2. Sentence tokenization. The description of each item is tokenized into sentences.
3. Sentence scoring. For each item, we calculate a score for each tokenized sentence as the average of the TF-IDF scores of all the entities it contains. This strategy would reduce the bias towards longer sentences which may contain more entities. Shorter sentences would be selected and be displayed in recommendation banners where space is limited.

In Fig.5, we illustrate the main components and steps of different explanation approaches.

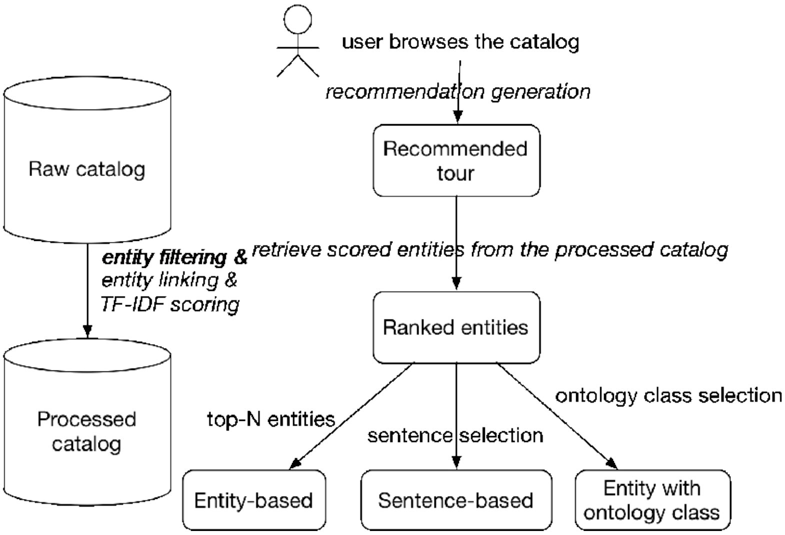

Fig. 5. Main components and steps of different explanation approaches

6. Evaluation

We conducted a qualitative user study to observe whether the entity-based explanations with ontology classes and the sentence-based explanations yield better results than the state of the art pure entity-based explanations. We describe successively the dataset, the recommendation algorithm, the parametrization, the candidate approaches, the experiment protocol and the results.

6.1. Dataset

We ran our experiment on a real commercial catalogue of a popular French tour-operator. This catalogue contains 1310 tours in 106 countries. The entity linking step resulted in 5161 distinct entities in the whole catalog and 20 entities per tour on average.

6.2. Recommendation algorithm and parametrization

We used an internal implementation of a knowledge graph-based recommendation algorithm [3]. This choice was made because of two reasons: (i) recommendation algorithm is not the focus of this paper (the recommendation and the explanation are dissociated as in many approaches [2]), and (ii) this existing algorithm has been proved to be efficient in terms of click-through rate improvement on multiple commercial websites. In the final explanations, we chose to display the top-3 best scored elements (entities, classes, sentences).

6.3. Candidate approaches

In Table 2, we enumerate 4 candidate approaches which are compared in our study. We assign them abbreviations which are used in the following subsections.

Table 2. Four candidate explanation approaches compared in the experiment

Abbreviation	Description
EN	entity-based (baseline)
NL	sentences in natural language
PC	pure class-based (all entities are replaced by classes)
CC	companion class-based (entities are accompanied by classes)

6.4. Experimental protocol

Our protocol is inspired by other similar studies [2, 8, 16]. The participants were asked to put themselves in the scenario of searching for a tour and to imagine a vague travel idea. Then they were asked to browse the tours of our evaluation dataset and to select several tours they are interested in. After that, they were recommended a tour with some basic information like the title, the photo, the duration and the price. This recommendation was accompanied by the 4 sets of explanations generated by the candidate approaches. They were asked to make a two-stage judgement. During the first stage, they were not allowed to check the details of the recommended tour and could only see the recommendation banners with different explanations (like in Fig. 6). They were asked to give a score on a five-point Likert scale on the statements related to intelligibility, effectiveness and efficiency (Table 3). They were told the meaning of the score: 1. strongly disagree 2. disagree 3. neither agree nor disagree 4. agree 5. strongly agree. During the second stage, they were asked to read the detailed descriptions of the recommended tour (like in Fig. 1). They then rate on the same scale the statements related to relevance and satisfaction (Table 3). They were encouraged to give free comments at any moment during the rating process. We designed this two-stage judgement because we wanted the statements of stage 1 to be assessed only based on a limited view on the recommendation. For the statements of stage 2, the relevance requires a complete knowledge about the recommended tour and we wanted to measure the overall satisfaction after the whole experience when the user has an idea about their perceived relevance.

Table 3. Summary of the rated statements in the 2-stage user study.

Aspect	Statement	Stage
Intelligibility	The explanation is easy to understand.	1
Effectiveness	The explanation conveys enough information about this tour to help me decide whether to discover more about it or not.	1
Efficiency	The explanation helps me to decide more rapidly whether to discover more about it or not.	1
Relevance	The explanation is relevant to the tour.	2
Satisfaction	Overall I am satisfied with explanation.	2

6.5. Results and discussion

We conducted through face-to-face interviews with 30 participants. The interview of each participant lasted on average 15 minutes. All the participants have experience in planning and purchasing travel products online. In Fig.6, we give two explanation examples (CC and NL) of a user who has been recommended a tour to Finland.

For the ratings, we considered that on a scale of 5 points, ratings of 4 and 5 are considered as positive. The percentages of participants who gave positive ratings on different aspects and on different explanation approaches are shown in Fig.7. Apart from the ratings on the five aspects, the participants left some free comments.

On the natural language side, we can clearly see that NL outperforms the baseline and achieves high scores on all aspects. Participants found that NL explanations contain a better quantity of information to help them understand the recommendation and make decisions more efficiently. Notwithstanding this, some negative free comments showing the drawbacks and limits of this approach are worth mentioning. We classify the badly perceived explanations into three categories. Firstly, some sentences are too long. For example, *A few kilometers from Apt, beautiful hiking day … Bastide des Claparèdes* which has 335 characters in total. Even though this sentence contains entities which are important to the tour and it is nicely written, its length is too important and its reading requires too many user efforts. Secondly, some sentences contain temporal expressions which do not make much sense without their surrounding sentences. For example, “then” in *You will then take the road to Gyeongju, capital of the kingdom Shilla*. Thirdly, some sentences are uninteresting as an explanation of the recommendation. For example, *São Filipe was the name of this island until 1680* which is a historic fact but does not directly help the user understand the tour. In any case, the results on NL should be viewed cautiously because the approach is dependent on the quality of the textual descriptions. In the travel domain, the descriptions are often well written in order to attract users.

Nordic skiing and pulka in Finnish Lapland
8 days
Starting from 1499 €

- **Earless seal (Pinnipeds)**
- **Saimaa (Lakes of Finland)**
- **Art Nouveau (Architectural Styles)**

- **Clear waters, and a varied wildlife including bears, wolves, bird eagles and the famous and very rare earless seal**
- **Two hours from Helsinki, Saimaa is the largest lake zone in the country**
- **In full autonomy, we will leave for 5 days to explore this natural environment and enjoy when we will return the attractions of the capital, between architecture style "art nouveau", green spaces, and omnipresence of the sea**

Fig. 6. Explanation examples of a recommended tour to Finland, the one on the top is generated by CC and the one on the bottom by NL

On the ontology class side, PC performs slightly worse than the baseline. Participants did not appreciate a simple juxtaposition of classes. On the contrary, CL outperforms the baseline on most of the aspects. This shows that it is indeed better to combine entities with ontology classes than to use entities or classes alone.

One notable negative point is that in certain cases, classes not very useful. Two cases can be distinguished.

The first case is that the entity does not need any hint, either the entity is self-explanatory like dbr: World_Heritage_Site, either it is very known. For example, the entity dbr:Italy is a European country and this is a shared knowledge for many people. A possible solution to this problem is to use the pagerank values[2] of the entities and to set a threshold upon which entities can be considered as known enough and do not need to be clarified.

The second case is that the ontology class is not interesting enough for explaining the touristic tour. For example, the entity dbr:Vienna has best ranked class yago:PopulatedPlacesInAustria while other classes (less profound in the hierarchy) like yago:WineRegionsOfAustria and yago:WorldHeritageSitesInAustria might be more interesting for the travel domain. This lack of domain interestingness might also partially explain the poor perceived quality of PC. We are currently experimenting with more sophisticated ontology class selection methods.

[2] http://people.aifb.kit.edu/ath/

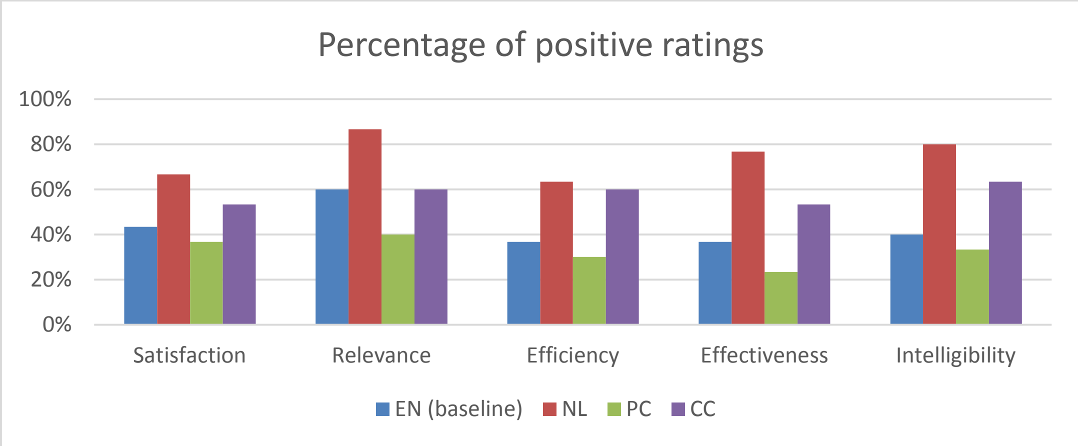

Fig. 7. Percentages of participants having given positive ratings (4 or 5) on different aspects and approaches

7. Conclusion

The explanation is an important component in recommender systems. It justifies the recommendation, helps users to better understand it and decide whether to take it or not. Thanks to the semantically interconnected data, knowledge graphs have been boosting the development of content-based explanation approaches since several years. However, most approaches focus on the exploitation of the structured semantic data to which recommended items are linked (e.g. actor, director, genre for movies).

In this paper, we have addressed the under-studied problem of leveraging knowledge graphs to explain the recommendations with items' unstructured textual description data. We point out 3 shortcomings of the state of the art entity-based explanation approach: absence of entity filtering, lack of intelligibility and poor user-friendliness. Accordingly, 3 novel approaches are proposed to alleviate these shortcomings. The first approach leverages a DBpedia category tree for filtering out incorrect and irrelevant entities. The second approach increases the intelligibility of entities with an integrated ontology (DBpedia, schema.org and YAGO). The third approach explains the recommendations with the best sentences from the textual descriptions selected by means of the entities.

We showcase our approaches within a travel tour recommendation explanation scenario. We conducted a thorough face-to-face user study with 30 participants. We used a real tour catalog from a popular French tour-operator which contains 1310 different tours in 106 different countries. Different explanation approaches have been evaluated on 5 aspects in two stages. Intelligibility, effectiveness and efficiency have been evaluated in stage 1, relevance and satisfaction in stage 2. The results showed that our proposed approaches (usage of natural language sentences containing important entities and usage of ontology classes to accompany important entities) outperform the state-of-the-art entity-based approach. They are easier to understand, contain more appropriate information to help users make decisions more efficiently, they are more relevant with respect to the explained recommendation and they yield better overall satisfaction. Multiple problems and limits of the approaches have been pointed out by our participants such as the need of reformulating long sentences, calculating the interestingness of sentences, determining the necessity of accompanying entities with ontology classes etc. We believe that the design of future explanation approaches could be guided by our study.

As future work, we consider testing our approaches on another tour catalog and in other domains. We also envisage studying the personalisation of the explanations.

References

[1] Bobadilla, J., Ortega, F., Hernando, A., & Gutiérrez, A. (2013). Recommender systems survey. Knowledge-based systems, 46, 109-132.

[2] Tintarev, N., & Masthoff, J. (2015). Explaining recommendations: Design and evaluation. In Recommender Systems Handbook (pp. 353-382). Springer US.

[3] de Gemmis, M., Lops, P., Musto, C., Narducci, F., & Semeraro, G. (2015). Semantics-aware content-based recommender systems. In Recommender Systems Handbook (pp. 119-159). Springer, Boston, MA.

[4] Passant, A. (2010, November). dbrec—music recommendations using DBpedia. In International Semantic Web Conference (pp. 209-224). Springer Berlin Heidelberg.

[5] Di Noia, T., Mirizzi, R., Ostuni, V. C., Romito, D., & Zanker, M. (2012, September). Linked open data to support content-based recommender systems. In Proceedings of the 8th International Conference on Semantic Systems (pp. 1-8). ACM.

[6] Piao, G., & Breslin, J. G. (2016, September). Exploring dynamics and semantics of user interests for user modeling on Twitter for link recommendations. In Proceedings of the 12th International Conference on Semantic Systems (pp. 81-88). ACM.

[7] Musto, C., Narducci, F., Lops, P., De Gemmis, M., & Semeraro, G. (2016). ExpLOD: A Framework for Explaining Recommendations based on the Linked Open Data Cloud. In Proceedings of the 10th ACM Conference on Recommender Systems (pp. 151-154). ACM.

[8] Musto, C., Narducci, F., Lops, P., de Gemmis, M., & Semeraro, G. (2018). Linked open data-based explanations for transparent recommender systems. International Journal of Human-Computer Studies.

[9] Billsus, D., & Pazzani, M. J. (1999, April). A personal news agent that talks, learns and explains. In Proceedings of the third annual conference on Autonomous Agents (pp. 268-275). ACM.

[10] Bilgic, M., & Mooney, R. J. (2005, January). Explaining recommendations: Satisfaction vs. promotion. In Beyond Personalization Workshop, IUI (Vol. 5, p. 153).

[11] Blanco, R., Ceccarelli, D., Lucchese, C., Perego, R., & Silvestri, F. (2012, October). You should read this! let me explain you why: explaining news recommendations to users. In Proceedings of the 21st ACM international conference on Information and knowledge management (pp. 1995-1999). ACM.

[12] Usbeck, R., Röder, M., Ngonga Ngomo, A. C., Baron, C., Both, A., Brümmer, M., ... & Ferragina, P. (2015, May). GERBIL: general entity annotator benchmarking framework. In Proceedings of the 24th International Conference on World Wide Web (pp. 1133-1143). International World Wide Web Conferences Steering Committee.

[13] Lu, C., Stankovic, M., Radulovic, F., & Laublet, P. (2017, May). Crowdsourced Affinity: A Matter of Fact or Experience. In European Semantic Web Conference (pp. 554-570). Springer, Cham.

[14] Vig, J., Sen, S., & Riedl, J. (2009, February). Tagsplanations: explaining recommendations using tags. In Proceedings of the 14th international conference on Intelligent user interfaces (pp. 47-56). ACM.

[15] Gedikli, F., Ge, M., & Jannach, D. (2011, August). Understanding recommendations by reading the clouds. In International Conference on Electronic Commerce and Web Technologies (pp. 196-208). Springer Berlin Heidelberg.

[16] Chang, S., Harper, F. M., & Terveen, L. (2016, September). Crowd-Based Personalized Natural Language Explanations for Recommendations. In Proceedings of the 10th ACM Conference on Recommender Systems (pp. 175-182). ACM.

[17] Borràs, J., Moreno, A., & Valls, A. (2014). Intelligent tourism recommender systems: A survey. Expert Systems with Applications, 41(16), 7370-7389.

[18] Neidhardt, J., Fesenmaier, D. R., Kuflik, T., & Wörndl, W. (2017, August). RecTour 2017: Workshop on Recommenders in Tourism. In Proceedings of the Eleventh ACM Conference on Recommender Systems (pp. 386-387). ACM.

[19] Ardissono, L., Goy, A., Petrone, G., Segnan, M., & Torasso, P. (2003). Intrigue: personalized recommendation of tourist attractions for desktop and hand held devices. Applied artificial intelligence, 17(8-9), 687-714.

[20] Zanker, M., & Ninaus, D. (2010, August). Knowledgeable explanations for recommender systems. In Web Intelligence and Intelligent Agent Technology (WI-IAT), 2010 IEEE/WIC/ACM International Conference on (Vol. 1, pp. 657-660). IEEE.

[21] Kapanipathi, P., Jain, P., Venkataramani, C., & Sheth, A. (2014, May). User interests identification on twitter using a hierarchical knowledge base. In European Semantic Web Conference (pp. 99- 113). Springer, Cham'

[22] Tonon, A., Catasta, M., Demartini, G., Cudré-Mauroux, P., & Aberer, K. (2013, October). Trank: Ranking entity types using the web of data. In International Semantic Web Conference (pp. 640-656). Springer Berlin Heidelberg.

[23] Suchanek, F. M., Kasneci, G., & Weikum, G. (2007, May). Yago: a core of semantic knowledge. In Proceedings of the 16th international conference on World Wide Web (pp. 697-706). ACM.