Use of topical and temporal profiles and their hybridisation for content-based recommendation

Luis M. de Campos[1] · Juan M. Fernández-Luna[1] · Juan F. Huete[1]

Abstract

In the context of content-based recommender systems, the aim of this paper is to determine how better profiles can be built and how these affect the recommendation process based on the incorporation of temporality, i.e. the inclusion of time in the recommendation process, and topicality, i.e. the representation of texts associated with users and items using topics and their combination. To that end, we build both topically and temporally homogeneous subprofiles to represent items. The main contribution of the paper is to present two different ways of hybridising these two dimensions and to evaluate and compare them with other alternatives. Our proposals and experiments are carried out in the specific context of publication venue recommendation.

Keywords Content-based recommendation · Topical profiles · Temporal profiles · Publication venue recommendation

✉ Juan M. Fernández-Luna
jmfluna@decsai.ugr.es

Luis M. de Campos
lci@decsai.ugr.es

Juan F. Huete
jhg@decsai.ugr.es

[1] Departamento de Ciencias de la Computación e Inteligencia Artificial, Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación, CITIC-UGR, Universidad de Granada, C/ Periodista Daniel Saucedo Aranda, S/N, 18013 Granada, Spain

1 Introduction

In today's world, recommender systems (RSs) (Bobadilla et al., 2013) play an extremely important role in our digital life (Lu et al. 2015). Many e-commerce platforms (such as Amazon or eBay, for example), entertainment services (e.g. Netflix, Spotify), or social media (Instagram, Facebook, Twitter, etc.) incorporate recommendation into their functionalities. These systems analyse how users interact with their products and suggest ones that the users might be interested in.

A suitable context where an RS could help, which is the research problem where our proposal is framed in, is the so-called publication venue recommendation (Wang et al. 2018). The general problem is stated as follows: A scientist, who just recently wrote a scientific paper (target article), wishes to select the most appropriate venue (journal, conference or scientific event[1]) where this article could be submitted for its evaluation trying to maximise the acceptance possibilities. The author could consider different criteria for selecting a journal, among others the impact factor or position in a ranking, but one of the most important is that the topics of the target paper fit with those from the journal scope (if the target article is about RS, the journal scope should contain RS). Once this fact is confirmed, the next step would be to verify that a number of papers dealing with the common topics are found published in the journal, i.e. a topical compatibility between some published papers and the target article exists (if the target article is about personalisation, tagging and book recommendation, there should appear articles published in the journal dealing with these same topics). A final condition to verify would be that these related articles had recently been published, which would mean that, for that journal, their topics are hot ones. In this case, an RS might help the author to perform this task. Given the target article, the RS would suggests some relevant journals where it could be submitted. The use of these RSs is widely spreading, and most of the big scientific publishers are incorporating them in their web sites for supporting the general premise of this paper. Then, the recommendation process continues starting from a target article as input, and matching it with the venues profiles. As a result, a ranking of suitable journals, conferences or general scientific events is generated and presented to the writer. As we have already mentioned, a large proportion of the profile consists of (weighted) terms or keywords, i.e. the words included in the textual descriptions of items associated to a number that reflects the importance of each term in the item and/or in the entire collection of terms. Other alternatives have, however, been explored in the literature in terms of profile construction such as the inclusion of topical and temporal dimensions. Regarding such topic-based profiles and assuming that the topics covered by the documents in the collection are available, textual subprofiles could be created for each topic, incorporating the text of all the associated documents about this topic. Let us imagine a simple situation in the context of publication venue recommendation. Let us suppose that a journal has published articles about RS, IR, personalisation and applications. The journal's profile would consequently consist of four subprofiles, each containing the texts of the articles in that research area (four different subprofiles

---

[1] From now on, and in order to simplify the language used in this paper, when journal is used, it could be interchanged by conference, seminar, scientific event or, in general, any publication venue.

containing articles dealing with RS, IR, personalisation and applications separately). Each journal, therefore, would be represented by several subprofiles that are more election of the best possible venue, although with some important limitations. An example is Springer Nature Journal Suggester,[2] among many others.

There are two main types of recommender systems (Adomavicius and Tuzhilin 2005): those based on collaborative filtering (CF) and those based on content-based recommendation (CBR) (Lops et al. 2019). While the first type generates suggestions based on user ratings for items they have consumed, the second offers recommendations based on the features (frequently textual descriptions) of these consumed items. In both cases, ratings or contents are stored in special structures called profiles (Gauch et al. 2007) and these represent the users' interests. RSs exploit these profiles to generate useful suggestions for users.

Returning to our venue recommendation context, considering that the main source of information of an article is its own content, and assuming that it is difficult to find user ratings assigned to journals (beyond the obvious binary rating of having published or not), CBR seems to be the most plausible approach to tackle this problem.

As mentioned below, in CBR, as long as there is a textual description of the items, profiles are usually represented by a bag of words which is obtained by combining the texts for any item the user has shown interest in. As CBR is primarily based on information retrieval (IR), the recommendation process consists in carrying out a matching between the active user's profile (in terms of IR, a representation of an information need, i.e. a query) and the textual representation of all the items in the collection (documents in the IR context). The degree of relevance of each item with respect to the corresponding profile is therefore computed, and a ranking of relevant items to be recommended is subsequently generated.

This approach is used for most recommendation contexts (books, films, music, etc.). There are, however, other situations where items are associated with a group of documents rather than a single text, such as expert finding (Lin et al. 2017), whereby experts on a certain subject are recommended according to a set of documents that define their expertise to any user who requires them, or our problem at hand, publication venue recommendation. In this case, the venues (items) comprise a series of articles published there (documents). In these two cases, item profiles might be built to define their specific informational context and subsequently used in the recommendation stage. The context in which the venues are also described by means of profiles compiled from the associated articles published in these venues represents the general premise of this paper. Then, the recommendation process continues starting from a target article as input, and matching it with the venues profiles. As a result, a ranking of suitable journals, conferences or general scientific events is generated and presented to the writer.

As we have already mentioned, a large proportion of the profile consists of (weighted) terms or keywords, i.e. the words included in the textual descriptions of items associated to a number that reflects the importance of each term in the item and/or in the entire collection of terms. Other alternatives have, however, been explored in the

---

[2] https://journalsuggester.springer.com/.

literature in terms of profile construction such as the inclusion of topical and temporal dimensions.

Regarding such topic-based profiles and assuming that the topics covered by the documents in the collection are available, textual subprofiles could be created for each topic, incorporating the text of all the associated documents about this topic. Let us imagine a simple situation in the context of publication venue recommendation. Let us suppose that a journal has published articles about RS, IR, personalisation and applications. The journal's profile would consequently consist of four subprofiles, each containing the texts of the articles in that research area (four different subprofiles containing articles dealing with RS, IR, personalisation and applications separately). Each journal, therefore, would be represented by several subprofiles that are more topically homogeneous, as articles are not mixed in one profile (heterogeneous) but grouped according to topics of interest. This method of organisation would clearly increase the interpretability of these (sub)profiles. This topical dimension can be automatically incorporated into the profiles by mining the texts in situations in which these categories are not clearly available, such as, for example, learning topic models using Latent Dirichlet Allocation (LDA) (de Campos et al. 2021; Jelodar et al. 2019) or by applying clustering algorithms (de Campos et al. 2020).

In situations where time is a feature included in those documents that define items (such as any type of timestamp), this dimension could be incorporated in different ways into the recommendation (Campos et al. 2014). Firstly, temporal subprofiles may be built by dividing the temporal line of documents into periods, grouping them into each one of these and thereby building the corresponding subprofiles. For example, in a journal with 10 years of existence, 10 different temporal subprofiles could be constructed starting from the articles published in the same year.

The homogeneity of the profiles (in this case temporal) is also present in the item profiling process. This approach could be considered as a generalisation of the well-known profiles based on long and short-term preferences. Secondly, but unrelated to profiles, another option is to include time in the recommendation process, by means of the application of a decay factor that penalises older items.

From now on focusing in the publication venue recommendation problem, and bearing in mind these topical and temporal dimensions for building homogeneous subprofiles, a question immediately arises: would it make sense to combine both aspects in order to improve the quality of the journal recommendation in terms of system effectiveness? Since the state of the art of general CBR shows that it is suitable, two innovative methods for carrying out such an integration are presented and evaluated in this paper and these represent the main contribution of this article. Being more concrete, two combination methods are introduced: while the first one mines the topics of the whole article collection and creates journal subprofiles according to a temporal division, the second approach initially carries out a temporal division of the articles and later, for each temporal split, extracts the topics inside and builds the corresponding journal subprofiles. Our goal is, therefore, to determine whether this general mixture of time and topics is valuable in comparison with other non-hybrid alternatives and which the best option is. For this purpose, this study will address the following research questions:

- RQ1: Does a temporal division of the scientific articles published in journals provide a reliable source for constructing high quality profiles?
- RQ2: Can decay-based techniques which penalise older scientific articles be successfully incorporated?
- RQ3: To what extent is building journal profiles based on latent topics in the article collection an added value for the venue recommendation problem?
- RQ4: Is hybridisation, i.e. the combination of topical and temporal aspects for creating journal profiles, a good alternative for the problem at hand?
- RQ5: Which method is the best form of hybridisation for the venue recommendation problem?

In order to find an answer to these questions, we have designed a detailed experimentation where the performance of the different models is tested. Although the answers drawn after analysing the obtained results are specific to the publication venue recommendation problem, and therefore not totally generalisable to other content-based problems, the methodology described in this paper, i.e. the profiling proposals, may be applied to many types of item collections represented by text in the context of CBR.

The remainder of this paper is organised as follows: Sect. 2 discusses other related work; Sect. 3 introduces the different methods for creating (sub)profiles based on terms, topics and time, and for combining them; Sect. 4 focuses on the experimental part of the work, including results and discussion; and, finally, Sect. 5 details our conclusions and outlines our future lines of research.

2 Related work

Starting with the simplest form of representing items or user profiles, the method usually adopted is to compile a list of weighted terms which are automatically extracted from the document associated with them (Gauch et al. 2007). These terms are supposed to correctly represent the document subjects and the weights are responsible for measuring their importance in terms of the entire document collection and within each document. Some examples of the use of term-based profiles for recommendation reflect how widely they are used in CBR, and these include TV programme recommendation (Wartena et al. 2011), expert finding (de Campos et al. 2020), treatments for patients in a health RS (Bateja et al. 2018), tweet recommendation (Benzarti and Faiz 2016), and image recommendation (Karlsen et al. 2018).

An alternative way of representing profiles to keywords is through the use of tags, concepts, categories or topics. We could say that these are higher-level features, named in a different way, are synonym terms and symbolise topics, trying to capture the underlying semantics of the items. Since the profiles comprise more general concepts rather than just words, certain authors believe that this is beneficial for the quality of the recommendation (Firan et al. 2007), at least in the case of tags. There are a number of examples of experiences that build and recommend based on tag profiles (Bogers 2018; Aliannejadi and Crestani 2018; Stakhiyevich and Huang 2019; Yan et al. 2020; Becerra et al. 2017). Tags are assigned to user profiles either manually or on the basis of a machine learning-based approach. The use of concepts is discussed

in a number of papers (Ren et al. 2015; Narducci et al. 2016; Sharma et al. 2017; Simsek and Karagoz 2020). In most cases, the concepts are extracted from ontologies or concept graphs giving the information associated to items or users. With respect to topic-based profiles (Saraswatm et al. 2020), these comprise latent topics mined directly from document collections, typically using the LDA algorithm or extensions of it. Starting with user-associated texts, the profile is fed with the most probable topics associated to the words contained in them. Certain papers illustrate the use of topic-based profiles to CBR in a wide variety of problems (Chen et al. 2017; Huang and Wu 2019; de Campos et al. 2021; Fu et al. 2021; Khan et al. 2021). Although the underlying representation based on topics is very appropriate for representing profiles, it does, in fact, lack the interpretability offered by terms, tags or concepts. A translation from topics to human-understandable labels is needed and this requires an additional effort.

All these previous item or user profiles are monolithic in the sense that all the possible facets of interests are combined into a single profile. Another alternative is to consider profiles as comprising different subprofiles, each associated to a possible facet, thereby capturing the various underlying, non-explicit topics which are usually extracted by machine learning algorithms from the associated texts of users and items. These multi-faceted profiles are no longer flat although they may have different shapes: trees, representing personal data, expertise and interests (Pavan and Luca 2015); graphs of clusters capturing different facets from different sources (Zeng et al. 2002); two subprofiles to capture user interests and friends' interests (Gulla et al. 2014); subprofiles comprising subsets of items rated by the user and which are used to improve the diversity of the recommendations (Kaya and Bridge 2019); different types of subprofiles, each containing keywords, concepts and tags (Narducci et al. 2013), or hierarchies of weighted topics (Kook 2005). Clustering is the usual technique for creating such multi-faceted profiles. This unsupervised learning is applied to the document collection resulting in clusters of documents or keywords, which will integrate the profiles as subprofiles. Each cluster would represent a concept in the entire collection. There are a number of papers which cover this methodology (Somlo et al. 2001; McGowan et al. 2002; Yeung et al. 2009; Amini et al. 2014; de Campos et al. 2020; de Campos et al. 2021).

In the research presented in this paper, topic-based profiles will comprise subprofiles which represent different concepts but rather than containing a list of topics, they contain terms, i.e. those from the documents associated with the topics.

Much has been published on temporal dynamics, i.e. the inclusion of time in recommendation, and this has mainly focused on CF (Campos et al. 2014). One of the most common approaches is to use decay functions to penalise old items and reward new ones (Ding and Li 2005; Yeniterzi and Callan 2015). A second alternative is to include time in the computation of item weights (Linda and Bharadwaj 2019). Another possibility is to integrate time into the rating matrix in CF and use it to find trusted relationships between users (Ngaffo et al. 2021). Another research line is to consider time frames: in the article (Ramos and Paraboni 2014), the authors propose a CBR system for tweets, where a specific time frame is learned for each user and only tweets within this personalised frame are recommended. The same idea has been used by

other authors (Si et al. 2017) but for points-of-interest recommendation. One generalisation is the use of long and short-term profiles as another option to include time and take into account the users' most recent interests in contrast to those which were acquired by interacting with the system some time ago (Li et al. 2011; Xiang et al. 2010). The time domain is included in our models simply by splitting the documents into time periods of equal size rather than using long and short-term profiles. Within each time period, the topic subprofiles are learnt.

This review of related work will also examine the combination of topicality and temporality in profiles by taking advantage of both dimensions in order to improve recommendation. This combination is performed by following a wide range of methods which are outlined in the papers mentioned, but the most usual way is to apply a latent topic discovery algorithm to the available text collections, obtain the topics associated to each document and incorporate time by means of weights associated to topics. Other authors use decay functions (Wangwatcharakul and Wongthanavasu 2021) to mitigate the impact of old ratings. They also use item reviews in order to obtain the underlying topics in the collection and associate the rated items to the corresponding topics in the reviews in order to track how the topics evolve with time. Finally, they propose an optimisation method to make predictions. In the article (Li et al. 2014) about news recommendation, the authors build long and short-term profiles. While long-term profiles comprise latent topics extracted from LDA from a collection of news and weighted by considering a time decay function to capture how they evolve in time, short-term profiles comprise topics occurring in documents in the most recent period of time. In the paper (Yin et al. 2015), their authors describe a method for the context of social media to combine interests and temporal context. It is based on mixing a latent class statistical mixture model to represent topic distributions not only from users' interests (user-oriented topics) but also from a temporal context (time-oriented topics). They also compute the distribution of topics for items. With all of this information, they are able to model different users' interests in different time periods. In Liu (2015), it is considered the interaction of each user u with each item i in a given time. LDA is applied to extract topics from the set of textual representations of all these interactions, which are represented by a topic distribution. For a given user, once all of their interactions have been sorted chronologically, the assigned topics are modelled as a time series. Recommendation takes place when a Gaussian process predicts the value of each topic at a given time and similarities are computed between the predicted topic distribution and the distribution associated with each user. In another article (Neshati et al. 2017) about community question answering, the authors introduce a method for future expert finding which suggests the most suitable experts for the future. In order to do so, they first apply LDA in order to extract topics from documents associated to experts, and their corresponding timestamps, and calculate the probability of a future expert candidate for a given query. In the context of social media (Nishioka and Scherp 2016), users' interests are extracted from social media streams. Profiles are then built using weighted topics, which are those obtained by applying LDA to the collection. Items are also indexed using concepts and matched to user profiles. Recommendation is carried out by computing a similarity between user profiles and item profiles. Publication times are also taken into account by means of decay functions, which penalise the older topics and are included in the topic weights.

In Zeng et al. (2018), it is built temporal user profiles by directly incorporating time into the LDA algorithm, thereby obtaining topic distributions for words and times, as in the case of timeSVD++ Koren (2010). The last two papers on expert finding use concepts rather than topics from LDA. In Rybak et al. (2014), the profiles consist of weighted concepts, where the weights represent the degree of expertise in each concept. The temporal expertise profile is a set of single profiles which are computed at different periods of time, and a decay function is incorporated into the calculation of the concept weights. In Ziaimatin et al. (2012), while short-term profiles are built by extracting and weighting concepts from an ontology over given time periods, long-term profiles are built by detecting the concepts which are uniformly distributed in the short-term profiles.

The way in which temporality and topicality are combined in this paper is a contribution to the state of the art. In most cases where LDA is used, it has been applied globally to the entire document collection. In this research, LDA is applied locally only to those documents belonging to the same period of time when both dimensions are combined.

Finally, and concerning the specific application field where we are focused, namely content-based publication venue recommendation, although the information within the profiles may vary (terms, noun-phrases, n-grams or topics), always each venue has either a single profile (Medvet et al. 2014; Silva et al. 2015; Yang and Davison 2012) or as many subprofiles as published articles (Kang et al. 2015; Rollins et al. 2017; Pradhan and Pal 2020; Errami et al. 2007). Different topically homogeneous subprofiles are only considered in de Campos et al. (2022). Moreover, there are hardly any works that explicitly use temporal information: in Alhoori and Furuta (2017), in the context of a collaborative filtering algorithm to recommend venues, a personal venue rating which considers the years when the articles published in a venue were added to a researcher personal collection is proposed. In Pradhan and Pal (2020), a similarity between venues which penalises older articles using a decay function (inverse log-weighting) is computed; this similarity in turn is used by a random walk with restart algorithm in a graph of venues. We have not found any work about publication venue recommendation combining topical and temporal information.

3 Alternatives to profile construction based on terms, topics and time

Considering a researcher who would be interested in knowing possible journals where they could publish a recently written paper, and the fact that journals are going to be modelled by profiles built from the articles published in them, as the building blocks of the recommendation process, in this section the different alternatives for building such profiles, based on terms, topics and time, are introduced. Although always with the publication venue recommendation problem in mind, these methods will be presented in a more formal way in order to understand how they could be generalised to other CBR contexts (as for example Expert finding).
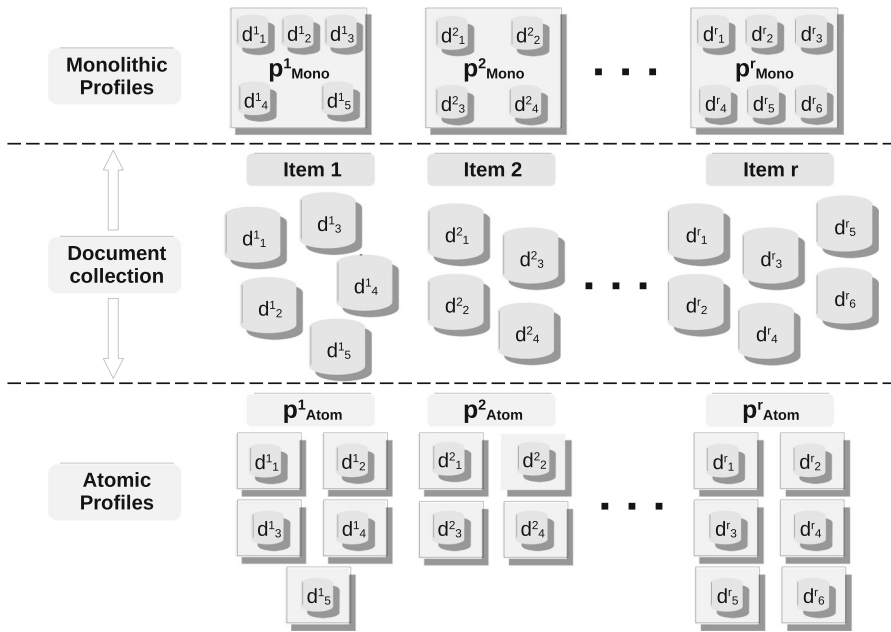
Fig. 1 Building monolithic and atomic (sub)profiles

3.1 Term-based profiles

Formally, let $I = \{i_1, \ldots, i_r\}$ be the set of items to be recommended. Linked with each of these, $i \in I$, is a set of n_i text documents $D^i = \{d_1^i, d_2^i, \ldots, d_{n_i}^i\}$. In the specific problem of publication venue recommendation, I would be the set of available venues and D^i the corresponding set of articles published in each of them.

Each item will also be represented by a profile that contains in one way or another the content of its related documents (the terms appearing in them). These profiles can basically be organised in one of two ways:

- Monolithic profiles: where all the documents linked to each item i are concatenated to create a single document, $d^i = \cup_{j=1}^{n_i} d_j^i$. This macro document will act as a unique profile p_{Mono}^i for item i, i.e. $p_{Mono}^i = \{d^i\}$.
- Atomic subprofiles: where for item i, its profile will comprise as many subprofiles as documents attached to it but in an isolated, unconcatenated way, i.e. $p_{Atom}^i = \{d_1^i, d_2^i, \ldots, d_{n_i}^i\}$ (each document is treated as a subprofile in itself).

The collection of items is then represented by a set of profiles, $\mathcal{P} = \{p^{i_1}, \ldots, p^{i_r}\}$, which will serve as retrieval units in this context of CBR systems. A graphical representation of this profile construction process is shown in Fig. 1.

These two basic ways of classifying profiles as either monolithic or atomic correspond in the expert finding literature with the so-called profile-based methods and document-based methods, respectively (Balog et al. 2012). There is no general agreement about which method is preferable, and although document-based methods tend

to be considered better than profile-based methods, profile-based methods perform better in certain cases (de Campos et al. 2015; Liu et al. 2005).

As a main conceptual drawback of both monolithic and atomic profiles is that they do not consider any valuable information provided by the articles in terms of the different general concepts that they deal with. In that sense, the monolithic profiles present an extreme compaction that makes them be totally heterogeneous (they integrate all the concepts in one single structure). On the other hand, the atomic subprofiles are in the complete opposite place, showing a radical but homogeneous decomposition.

We believe that there is room for improvement between these two extreme ways of building (sub)profiles and that we can create less extreme ways of organising the information related to each item. We will therefore use both approaches as baselines in our experiments, as they are the most basic ways of representing (sub)profiles and, as shown, are frequently found in the literature.

3.2 Topical profiles

One first alternative for organising an item's subprofiles (out of the two basic organisational schemes presented in the previous section) would be to build more homogeneous subprofiles around the different concepts or topics which can be identified in the entire collection of text documents associated to the items. The construction of subprofiles from a topical perspective can be based on a partition of the document collection by means of a clustering algorithm which uses the documents' terms as features. This would identify the different clusters of documents according to their subjects, placing all the conceptually-related documents in the same cluster. For an item i, each subprofile will correspond to the concatenation of the documents associated to i which are assigned to the same cluster, and this results in a set of topically homogeneous subprofiles. It is apparent that while this clustering process is global in that it is carried out with the entire document collection (not with the documents associated to an item), the subprofile construction is local, as the subprofiles only contain the text of documents associated to the item i.

Although there are many ways of performing this clustering process, in this paper we will use LDA (Latent Dirichlet Allocation) (Blei et al. 2003) for this purpose. LDA finds k latent topics, $x_1, x_2, \ldots, x_k$, in a document collection,[3] where each topic x_l is characterised by a conditional probability distribution of terms, $p(t \mid x_l)$, and determines for each document d a probability distribution of topics, $p(x_l \mid d)$. For example, in the context of recommending Machine Learning journals, we could highlight the fact that the articles published in all the journals deal with five different topics (to clarify the example, these will be called clustering, classification, regression, association, and feature selection). One article might deal only with classification (100%), a second one might mainly be about feature selection (70%) but may also discuss classification as a secondary topic (30%), a third one might mostly cover regression (90%) but also briefly touch on feature selection (10%), and a fourth one might cover all the topics equally (20% each), as it could be an introduction to Machine Learning.

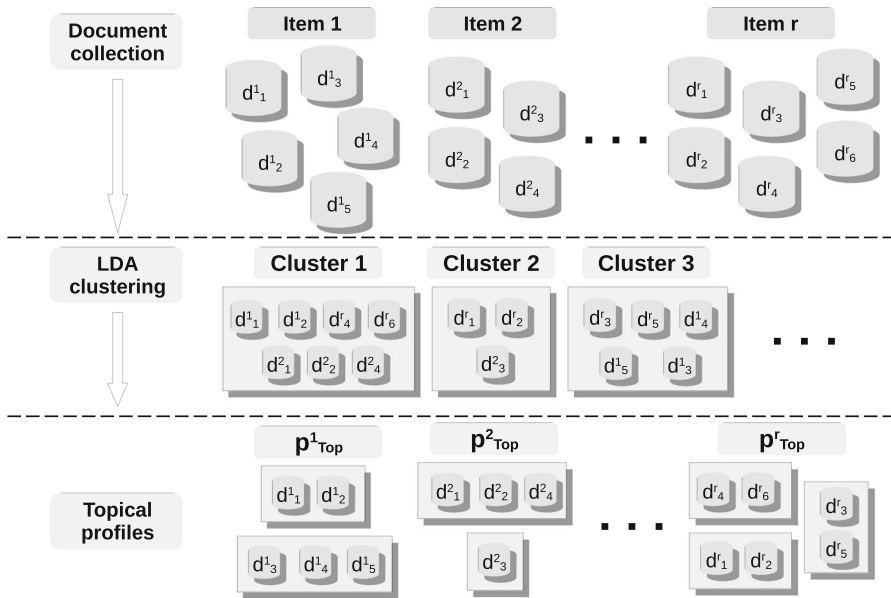[3] The number of topics, k, is an input parameter of LDA.

Fig. 2 Building topical profiles

The clustering generated by LDA obtains k clusters, one for each topic x_l, where each document is assigned to its most probable topic. For each item i, there are therefore at most[4] k subprofiles, each containing the concatenated text of those documents associated to i on the main topic of x_l. Figure 2 illustrates the entire process for generating the topical subprofiles.

In terms of a more formal description, given the set of all the documents $\mathcal{D} = \cup_{i=1}^{r} D^i$, each cluster, $\mathcal{D}_l$, $l = 1, \ldots, k$, consists of the documents of the items which are associated to the l-th topic, x_l (documents where the most probable topic is x_l), that is to say:

$$\mathcal{D}_l = \{d_m^{i_j} \mid x_l = \arg \max_{s=1,\ldots,k} p(x_s \mid d_m^{i_j}), \ j = 1, \ldots, r, \ m = 1, \ldots, n_{i_j}\} \quad (1)$$

From these sets of documents relating to each of the k topics, the subprofiles of each item i must be constructed by grouping the documents in each global cluster that are associated to this item i, thereby obtaining a local cluster, $D_l^i = \mathcal{D}_l \cap D^i$. Each item will therefore have assigned as many subprofiles as local clusters have been generated for it. The subprofiles of item i are then documents, $d^{i,l}$, built by concatenating the documents within each local cluster D_l^i, $d^{i,l} = \cup_{d_j^i \in D_l^i} d_j^i$. In this case, the topical profile for an item i is $p_{Top}^i = \{d^{i,1}, d^{i,2}, \ldots, d^{i,k}\}$.

Following with the example of recommending Machine Learning journals introduced some paragraphs above, all the articles published in all the journals would serve

---

[4] If an item i has no associated documents about the main topic of x_l, then the corresponding subprofile does not exist for that item.

as input for LDA, as well as a value of k (let us say 5 for the sake of continuity with the example). The output would be the assignation of the most probable topic to each article. Let us assume that one hypothetical journal entitled "Machine Learning Trends", denoted as journal i_3, contains six articles with the following assignations to their most probable topic: $d_1^3 \rightarrow x_3$, $d_2^3 \rightarrow x_2$, $d_3^3 \rightarrow x_5$, $d_4^3 \rightarrow x_2$, $d_5^3 \rightarrow x_3$, $d_6^3 \rightarrow x_3$. Then, 3 different subprofiles would be built for this journal according to the distribution of its articles in 3 topics: $D_2^3 = \{d_2^3, d_4^3\}$, $D_3^3 = \{d_1^3, d_5^3, d_6^3\}$ and $D_5^3 = \{d_3^3\}$, conforming in this way the profile p_{Top}^3 of journal i_3.

With this method of creating venue subprofiles based on the underlying topics mined from the collection, the subprofiles are better structured, distinguishing between the intra-homogeneity and extra-heterogeneity properties. Each single subprofile presents a very homogeneous content as the documents included in it are all related among them because they deal with the same topic. So in the previous example, as all articles d_1^3, d_5^3 and d_6^3 discuss, one way or another, about the topic x_3, they conform a homogeneous subprofile $d^{3,3}$. The other two, concerned to different topics, are the base for the other two subprofiles. The three of them deal with different topics and are heterogeneous among them. This division originates a journal may be recommended because a single matching between one of its subprofiles and the target article, avoiding the potential noise of being all of the articles mixed in the same profile, as happening in the monolithic profiles.

3.3 Temporal profiles

Another alternative for organising item subprofiles assumes that all the documents relating to any item have an assigned date (e.g. a publication date). We could then sort them accordingly and establish certain temporal divisions to exploit the temporal dimension. From each period, homogeneous subprofiles (in a temporal sense) could be built to represent an item by grouping together all the documents associated to that item which belong to the same period.

Continuing with the previous example of scientific venue recommendation of Sect. 3.2, imagine the situation of a journal about machine learning with an extensive record in the publication of articles. As expected, there are times when there are a large number of papers on the same topic. After a while, new research topics appear and as these are published, they displace and possibly replace some of the existing topics. Initially, the majority of work published in journals of this kind dealt with neural networks but after a few years, researchers became more interested in support vector machines, and nowadays the focus has shifted towards deep learning and this is currently one of the most published topics. Splitting the time line into different intervals and building subprofiles for the corresponding journal in each period is an alternative and simple way of endorsing a temporal perspective to subprofiles and reflects how the focus of the papers has changed over time in the journal.

In order to formalise this idea, let us consider the $h+1$ time points $t_0 < t_1 < \ldots < t_h$ and the h temporal intervals $[t_{u-1}, t_u)$, $u = 1, \ldots, h$. If $date(d)$ is a function that returns the date of document d, then the h global temporal clusters $\mathcal{T}_u$ are defined as

follows:

$$\mathcal{T}_u = \{d_m^{i_j} \mid t_{u-1} \le date(d_m^{i_j}) < t_u, \ j = 1, \ldots, r, \ m = 1, \ldots, n_{i_j}\}. \quad (2)$$

The local temporal clusters for each item i are built as in the case of topical clusters by grouping the documents associated to i that belong to each global cluster $\mathcal{T}_u$, i.e. $T_u^i = \mathcal{T}_u \cap D^i$. Each subprofile for item i concatenates the documents in T_u^i to form a single document $d^{i;u} = \cup_{d_j^i \in T_u^i} d_j^i$. Each item, therefore, is now represented by at most h temporal subprofiles.[5] The temporal profile for item i in this case is $p_{Temp}^i = \{d^{i;1}, d^{i;2}, \ldots, d^{i;h}\}$.

Let us assume that our hypothetical journal entitled "Machine Learning Trends", i_3, contains six articles with the following publication dates: $d_1^3 \to 2015$, $d_2^3 \to 2016$, $d_3^3 \to 2017$, $d_4^3 \to 2020$, $d_5^3 \to 2022$, $d_6^3 \to 2022$. Then four temporal subprofiles, corresponding to the four biannual intervals [2015, 2017), [2017, 2019), [2019, 2021), [2021, 2023), could be constructed grouping documents by publication year: $T_1^3 = \{d_1^3, d_2^3\}$, $T_2^3 = \{d_3^3\}$, $T_3^3 = \{d_4^3\}$, $T_4^3 = \{d_5^3, d_6^3\}$, respectively, conforming in this way the four subprofiles of the temporal profile, p_{Temp}^3 of journal i_3.

One advantage of this approach is that, in certain cases, articles dealing with the same underlying topics would tend to appear in the same temporal subprofiles, reflecting the natural evolution of topics along the time. Another important advantage is that these subprofiles could be weighted in order to give more weight to those closer to the current time, thus giving more importance to journals which recently published about the topic of the target article.

Following the argument of this second benefit, a temporal decay might also be considered to reduce the influence of older articles or subprofiles. This is a complementary way of introducing the temporal dimension into this journal recommendation problem. The underlying idea is that a user would be more interested in journals where their last published articles are closer to the paper to be published rather than "older" articles. In order to carry out this idea, once a query has been submitted (the text of the article) and the score or Relevance Status Value (RSV) for each article or subprofile has been computed, a decay function (Larrain et al. 2015) is applied in order to modify the corresponding RSV according to the temporal distance to the year of the "newest" papers in the article collection.

3.4 Hybrid profiles: combining topical and temporal profiles

The next natural step is to combine both previous ways of constructing profiles by simultaneously exploiting the topical and temporal dimensions in order to obtain homogeneous subprofiles in terms of these two properties. This homogeneity could be reached in two different ways:

- By first discovering the underlying global topics within the whole collection and creating their corresponding clusters (topical division), as explained in Sect. 3.2, and secondly by splitting them into temporal units (temporal division), from which

---

[5] This is because it is possible for certain items not to have any associated documents in a given period.

the final subprofiles will be constructed. This approach is then a topical-temporal one.

- By first splitting the collection into temporal units (temporal division), secondly by discovering the underlying topics within each single temporal partition (locally), and finally by building the final subprofiles from them. In this case, this is a temporal-topical approach. It should be noted that in each temporal division, the discovered topics would be different.

More specifically, for the topical–temporal approach, let $\mathcal{D}_l$ be defined as in Eq. (1), $l = 1, \ldots, k$, and $\mathcal{T}_u$ be defined as in Eq. (2), $u = 1, \ldots, h$. The global topical-temporal clusters $\mathcal{DT}_{lu}$ are then defined as

$$\mathcal{DT}_{lu} = \mathcal{D}_l \cap \mathcal{T}_u, \ l = 1, \ldots, k, \ u = 1, \ldots, h. \tag{3}$$

Given an item i, its local clusters are obtained by joining the documents found in each global topical-temporal cluster which are associated to i: $DT^i_{lu} = \mathcal{DT}_{lu} \cap D^i$. As in the previous cases, the documents within each local cluster are concatenated to form the corresponding subprofiles, $d^{i,l;u} = \cup_{d^i_j \in DT^i_{lu}} d^i_j$. The topical-temporal profile for item i is $p^i_{TopTemp} = \{d^{i,1;1}, \ldots, d^{i,k;h}\}$.

Following with the example presented in the previous sections about the journal recommendation problem, all the articles of all the journals would serve as input to an LDA algorithm, which would assign the articles up to 5 topics. Then inside of each cluster, temporal partitions would be applied in order to create subprofiles considering periods of time. For the journal i_3, whose articles were grouped in only three topics (x_2, x_3 and x_5) and four biannual intervals, we would obtain only five (out of twelve possible) topical-temporal subprofiles, namely $DT^3_{21} = \{d^3_2\}$, $DT^3_{23} = \{d^3_4\}$, $DT^3_{31} = \{d^3_1\}$, $DT^3_{34} = \{d^3_5, d^3_6\}$ and $DT^3_{52} = \{d^3_3\}$. Note that there could be combinations of temporal periods and topics which might be empty, as in this example.

The main feature of this approach is that LDA is globally applied to the whole collection of articles, reflecting the global topics present at the journals belonging to the collection. The division of subprofiles in time periods will cause to place articles in their corresponding publication slot. This means that a topic could be placed in specific time divisions, being a more fine-grained approach. If a topic is just mentioned within a journal in, let us say, only three articles, all from the same period, they will be allocated in the same subprofile, thus giving a more precise information about when this topic has been treated in the journal.

In the case of the temporal-topical approach, starting from each $\mathcal{T}_u$ as defined in Eq. (2), $u = 1, \ldots, h$, as the document collection, we use LDA to obtain the k topics $x_{u1}, \ldots, x_{uk}$ corresponding to this collection.[6] We then proceed in the same way as with the topical profiles, i.e. we obtain the clusters $\mathcal{TD}_{ul}$ as the subset of documents of $\mathcal{T}_u$ where the most probable topic is x_{ul}:

---

[6] It should be noted that we must apply LDA to every h temporal subcollections of documents, thereby obtaining specific topics for each time period.

$$\mathcal{TD}_{ul} = \{d_m^{i_j} \in \mathcal{T}_u \mid x_{ul} = \arg \max_{s=1,\dots,k} p(x_{us} \mid d_m^{i_j}), \ j = 1, \dots, r, \ m = 1, \dots, n_{i_j}\},$$
$$l = 1, \dots, k, \ u = 1, \dots, h. \quad (4)$$

We then obtain the clusters associated to each item i as $TD_{ul}^i = \mathcal{TD}_{ul} \cap D^i$. Finally, the documents in TD_{ul}^i are concatenated to build the subprofiles, $d^{i;u,l} = \cup_{d_j^i \in TD_{ul}^i} d_j^i$. The temporal-topical profile for item i is $p_{TempTop}^i = \{d^{i;1,1}, \dots, d^{i;h,k}\}$.

The difference with the previous topical-temporal method is that the clustering is local to the articles published in each specific period of time, reflecting better topic distributions of the articles in that period. This could be a more precise approach, but at the same time, each LDA is run with a lower number of articles and this might affect the quality of the mined topics.

It is worth noting that in both cases in this process, each item i will have associated at most $k * h$ subprofiles because there might not be any document associated to i which deals with a specific topic at a given time period. Also we would like to remark that the temporal decay is suitable to be applied in both topical-temporal and temporal-topical profiling methods.

3.5 Using the subprofiles for recommendation

Once we have presented the different types of (sub)profiles[7] relating to each journal, then a mechanism has to be designed in order to carry out the journal recommendations. For that purpose, we consider that IR-based techniques are the most appropriate ones to perform this content-based recommendation task.

The first step is to index the (sub)profiles,[8] so all of them are correctly arranged with the aim of an efficient posterior access. This task will be performed by an IR system (IRS).

The second step is the recommendation itself. Then, given an active user, the objective is to obtain a ranking of relevant journals for her target article. This might be achieved by means of an IR model implemented in an IRS, which will be responsible for retrieving relevant journals in respect of the user's query. More specifically, the textual content of a paper to be published by the user represents a query submitted to the IRS. Then by means of the similarity measure in which the IR is based on, a score of similarity between the target article and each subprofile is computed and a ranking of journal (sub)profiles is finally generated (sorted decreasingly according to their relevance to the query).

As mentioned in the previous paragraph, the ranking consists of subprofiles. However, since the active user requires journal recommendations, the subprofile ranking should then be transformed into a journal ranking. For this purpose, a final fusion process must combine the scores of the subprofiles for each journal and generate a final

---

[7] All of the methods presented (with the exception of the one based on monolithic profiles) generate several subprofiles per journal. The monolithic approach builds exactly one profile per journal and this is why we write (sub)profiles.

[8] It should be remembered that each subprofile is represented as a text document.

journal ranking. This is performed by means of the application of fusions methods (de Campos et al. 2017), which aggregate the scores of all of the journal subprofiles, computing a single relevance score for each journal. This fusion is not necessary for monolithic profiles since in this case there is only one profile per journal.

4 Evaluation and results

In this section, we shall detail everything relating to the evaluation of the previously explained alternatives for item profiling and also the results obtained.

Reminding the problem at hand, publication venue recommendation, given a target article (or at least an abstract, title and keywords), the problem is to recommend to the active user the most suitable venue for publishing such a paper on account of the suitability of the scope of the journal.

The following sections will present all the details of the experimental design and also the results of the experiments.

4.1 Test collection

The test collection used in the experimentation is called PMSC-UGR (Albusac et al. 2018) and has been created by the authors from PubMed and Scopus. It originally contained 762,508 articles from 12,396 journals in the biomedical domain, with a title, abstract, keywords, citations and authors for each paper. Out of all the authors from this selection of papers, those who were unequivocally represented by their corresponding ORCID codes were finally selected, leaving a total of 20, 406 authors.

4.2 Implementation details of the recommendation model

As mentioned in Sect. 3.5, an IRS is in charge of indexing and retrieving journal (sub)profiles. Therefore, for the first task, we have developed an indexing programme based on the Lucene library.[9] Previously, this piece of software removes stop words and performs stemming, indexing only the resulting stems. With respect to the similarity measure required in the second task, the Lucene implementation of the Language Model with Jelinek-Mercer smoothing has been used.

The task of fusing the rankings of subprofiles of a same journal has been implemented by means of the *CombLgDCS* fusion method (de Campos et al. 2017), which aggregates the scores of all of the journal subprofiles, decreasing them proportionally to the logarithm of their positions in the ranking.

In order to discover the latent topics required for topical, temporal-topical and topical-temporal models, and obtain topically homogeneous subprofiles, it is required to implement an LDA algorithm. In our case, the chosen Python implementation of LDA is the given by the Gensim library, with the default values for hyper-parameters.

During this process, terms that appear in fewer than 750 documents are ignored and also those that appear in more than 90%. This decision was made in order to

---

[9] https://lucene.apache.org/.

remove very common (almost considered stopwords) and very rare terms, which, if used as input for LDA, could distort the learned topics and also increase the complexity. Moreover, only a maximum of 5000 of the remaining most frequent terms in the corpus have been considered as LDA input. These figures correspond to previous experiments to set the most suitable size of the vocabulary. Nevertheless, the subprofiles built with this technique contain all the terms from the original documents.[10]

4.3 Experimental design

For evaluation purposes, we have restricted the PMSC-UGR collection to those papers published between 2007 and 2016 which appear in journals with more than 100 papers in this period,[11] leaving the dataset with a total of 1002 journals. The collection has then been split into two partitions: the first comprises articles dating from 2007 to 2015 (a total of 276,679 papers), and this will be reserved for building the (sub)profiles and serve as the training set; and the second only contains articles from 2016 (32,864 articles) and this will be used as the test set. This holdout method is suitable for this evaluation and does not require cross-validation to obtain reliable results given the large number of articles in the test set. Each article from the test partition will be considered as a query to be submitted to the underlying IRS and this query will comprise the combined text of its title, abstract and keywords.

Starting from the training partition, the following types of (sub)profiles will be considered for our experiments, pointing out which are the baselines:

- Monolithic profiles ($Mono$) (Baseline)
- Atomic subprofiles ($Atomic$) (Baseline)
- Topical subprofiles (Top)
- Temporal subprofiles ($Temp$): These are built from four temporal partitions of two years each (2007–2008, 2009–2010, 2011–2012 and 2013–2014) and one of only one year (2015), i.e. $h = 5$.
- Topical + Temporal ($TopTemp$): After applying a global LDA to the entire article collection, subprofiles are built in the five temporal partitions.
- Temporal + Topical ($TempTop$): Subprofiles are built from the five temporal partitions after applying local LDA to the article collection in each partition.

In addition, 10 instances of randomised subprofiles have been created ($Random1, \ldots, Random10$). Each comprises 5 random partitions of the articles from the training set, thereby replacing the 5 temporal subprofiles with 5 random subprofiles. The retrieval effectiveness of these data sets will also be measured in order to test whether the temporal divisions differ from randomness.

For the LDA algorithm on which the Top, $TopTemp$ and $TempTop$ approaches are based, it is necessary to set up the k parameter, i.e. the number of latent topics to be discovered. This is not an easy task because the quality of the results could be

---

[10] i.e. the mentioned reduction of the vocabulary is temporal, only for applying LDA and obtaining the clusters of articles.

[11] We have removed the journal PlosOne from this set because it has a much greater number of papers than the others.

very highly depend on this value. In this research, we have considered three different values, two of which are related to medical categories or specialities:

- number of comprehensive medical specialities extracted from the Medical School blog at St George's University,[12] $k = 20$
- number of second-level categories of the MESH thesaurus,[13] $k = 110$
- $k = 400$, in order to test a very large number of topics

It is apparent that these three values attempt to cover a wide range of topics, from a low to a relatively large number of them, in order to evaluate the performance of the different types of profiles according to the number of topics discovered by the LDA algorithm. The underlying idea is also to choose meaningful values of k relating to medical categories. In this case, and in order to not increase the complexity of the study, we have made the decision of using the same k values for all the different subprofile approaches.

Regarding the decay functions introduced in Sect. 3.3 that might be applied for the truly temporal approaches, i.e. $Temp$, $TopTemp$ and $TempTop$, and for $Atomic$, which also supports a temporal treatment, and among the wide range of decay functions found in the literature, we have implemented two of them to be tested:

- linear: $RSV_{LinearD} = \frac{NRSV}{1+Penalty}$
- doubled squared root[14]: $RSV_{2SqrtD} = \frac{NRSV}{\sqrt{\sqrt{1+Penalty}}}$

where $NRSV$ is the normalised RSV, computed by dividing the corresponding RSV by the maximum of the ranking and $Penalty = 2015\text{-}Publication\ year$, where 2015 is the year of the "newest" papers in the article collection. For individual articles in the atomic subprofiles, their publication year is used directly to compute the new score, whereas for the remaining temporal subprofiles, their average value is incorporated into the decay formulas since they contain articles over two years. A third value of the decay parameter would be "None", which means that no temporal penalisation is considered in the RSV.

These three options try to study the behaviour of decay functions in three extreme situations: a rather smooth penalisation of older articles (2Sqrt), a strong penalisation (Linear) and no penalisation at all (None). We have not tried to find the best decay functions, although some other functions were tested (the results were more or less similar and we will not include them).

4.4 Evaluation measures

In order to evaluate the recommendation, and before presenting the evaluation measures, it is important to determine the ground truth: in this case, only one journal is relevant for each query (test target article) and this is the journal where the paper has actually been published. This is a very objective criterion, although it is also very

---

[12] https://www.sgu.edu/blog/medical/ultimate-list-of-medical-specialities.

[13] https://meshb.nlm.nih.gov/treeView.

[14] A kind of power decay using the function x^{-4}.

conservative, as probably other recommended journals could also be appropriate for publishing the test target article.

The following evaluation measures found in the literature to determine the quality of the results obtained by venue recommendation methods are the most common:

- Recall@X (R@X): This measures the ability of recommending the relevant journal where the test target article has been published in the first X journals in the ranking (recommended venues). In other words, we compute the average number of times where the actual venue where a test paper was published is among the first X recommended venues. In previous work on venue recommendation (e.g. Luong et al. 2012; Medvet et al. 2014; Wang et al. 2018), this measure is also called accuracy@X. Two thresholds X are considered to compute the values of this measure: 1 and 5. $X = 1$ is considered because the number of relevant journals is 1 and so we would like to know how successful it would be to recommend only one journal. Since the user does not usually obtain only one recommendation but a number of alternative journals, $X = 5$ is also used as the threshold.
- Mean Reciprocal Rank (MRR@Y): In this case, the idea is to reflect how high in the ranking the only relevant journal is recommended. It therefore computes the average of the inverse of the positions in the ranking for the journal where each test paper was published. The total number of results (journals) in the ranking is limited to Y.[15] We have only considered the top 40 positions in the ranking, i.e. $Y = 40$. In our specific case where only an item (a journal) is relevant, this measure coincides with mean Average Precision, MAP@Y.

4.5 Results

The results of our experiments are displayed in Table 1 . Although the rankings of methods obtained for the different performance metrics are not identical, the trends are the same. In fact, if we calculate the Pearson correlation between these rankings, we always obtain correlation coefficients which are greater than 0.86 (if we exclude the results obtained by the random subprofiles and also those which use linear decay, which are both quite poor, then all the correlation coefficients are greater than 0.98). We can, therefore, comment on our results without referring to any specific metric.

If we first focus on RQ1, we can conclude that the use of the temporal dimension alone (*Temp* subprofiles, without decay) only very slightly improves the results obtained by the *Mono* baseline. This implies that the impact of dividing the monolithic profiles into several parts which are only based on temporal criteria is limited. In addition, the *Atomic* approach performs almost the same as *Temp*. Splitting the article collection into years and building the journal subprofiles upon them does not, therefore, offer any clear advantage. The random subprofiles, Random*i* (with the same number of subprofiles as *Temp*), perform almost identically to *Mono* and worse than *Temp*. This implies that the temporal dimension has a slightly positive effect on the results which is not attributable merely to the fact of creating several subprofiles.

When a decay factor is also used, very poor results are obtained for *Temp* (even worse than the baselines) with the linear version and much better results with the

---

[15] If the relevant journal is not within the first Y journals in the ranking, then the value is 0.

Table 1 Results of the experiments (the best results for each column are shown in bold)

Subprofiles	#Topics	Decay	R@1	R@5	MRR@40
TempTop	110	2Sqrt	**0.2501**	**0.5655**	**0.3935**
TempTop	20	2Sqrt	0.2490	0.5619	0.3923
TempTop	20	None	0.2466	0.5615	0.3903
TempTop	400	2Sqrt	0.2455	0.5596	0.3883
TempTop	110	None	0.2451	0.5638	0.3898
TempTop	400	None	0.2434	0.5575	0.3860
TopTemp	400	2Sqrt	0.2373	0.5462	0.3783
TopTemp	110	2Sqrt	0.2368	0.5463	0.3781
Top	20	None	0.2348	0.5400	0.3760
Top	110	None	0.2346	0.5448	0.3763
Top	400	None	0.2345	0.5462	0.3771
Temp		2Sqrt	0.2341	0.5431	0.3758
TempTop	110	Linear	0.2335	0.5307	0.3692
Atomic		2Sqrt	0.2331	0.5403	0.3731
TopTemp	110	None	0.2331	0.5431	0.3747
TopTemp	20	2Sqrt	0.2329	0.5419	0.3746
TempTop	400	None	0.2328	0.5430	0.3746
TempTop	400	Linear	0.2308	0.5266	0.3651
TopTemp	400	Linear	0.2298	0.5179	0.3615
TempTop	20	Linear	0.2294	0.5267	0.3657
TopTemp	110	Linear	0.2290	0.5207	0.3618
TopTemp	20	Linear	0.2286	0.5230	0.3629
TopTemp	20	None	0.2286	0.5361	0.3699
Atomic		None	0.2282	0.5370	0.3696
Temp		None	0.2258	0.5330	0.3671
Atomic		Linear	0.2255	0.5118	0.3558
Random2		None	0.2245	0.5340	0.3656
Random1		None	0.2243	0.5326	0.3658
Random6		None	0.2242	0.5330	0.3660
Random9		None	0.2241	0.5321	0.3653
Random3		None	0.2240	0.5319	0.3657
Random5		None	0.2239	0.5330	0.3656
Random4		None	0.2239	0.5333	0.3655
Random8		None	0.2238	0.5332	0.3655
Random0		None	0.2236	0.5335	0.3653
Monolithic		None	0.2236	0.5278	0.3653
Random7		None	0.2227	0.5314	0.3648
Temp		Linear	0.2147	0.5159	0.3543

doubled squared root version (2Sqrt) (which is smoother than linear). Consequently, the performance of $Temp$ clearly depends on the aid that decay functions can provide. This behaviour of the three decay methods (Linear $<$ None $<$ 2Sqrt) also persists when the temporal dimension is combined with the topical dimension[16] or when it is applied to the other baseline, $Atomic$, so that the preferred version of decay is always 2Sqrt. $Temp$ is also better than $Atomic$ (using 2Sqrt decay). By way of conclusion, a well-designed decay function incorporated into the journal recommendation process can boost performance, and this therefore answers RQ2.

In terms of RQ3, the use of topical subprofiles (Top) clearly improves the results of the baselines $Mono$ and $Atomic$, regardless of the number of topics selected. Since there are no very important differences between the results of Top with a different number of topics, this parameter does not seem critical for good behaviour. Top is also better than $Temp$ without decay and has a similar performance to $Temp$ with 2Sqrt. However, the contributions of Top and $Temp$ to improved performance seem to be based on different premises, so that their combination could generate a kind of synergy. This is indeed the case but it depends on the way the topical and temporal subprofiles are combined. When $TopTemp$ subprofiles are used, i.e. first a topical division and then the temporal division (and using the same topics in every time period), we do not observe any clear improvement in the results in terms of using only either Top or $Temp$ (and only a minuscule improvement is obtained). However, when we use the other proposed combination of $TempTop$ subprofiles, i.e. first a temporal division and then the topical division (with the topics being specific for each time period), a clear improvement is apparent. Moreover, the $TempTop$ results are always the best ones for all the metrics, regardless of the number of topics being considered. We think that this is due to the fact that topic identification, and the subsequent construction of subprofiles, is tailored to the set of articles included in each temporal partition, which is more precise and totally adapted to the content of such articles. Additionally, the $TopTemp$ approach is more general and not so well fitted to the texts in each temporal partition. Hybridising topical and temporal profiles is, therefore, a very interesting approach but only if temporal divisions are made and journal subprofiles built based on topic discovery in each time-based partition, which answers RQ4. The absolutely best results for the problem of journal recommendation are obtained using $TempTop$ subprofiles with 110 topics and 2Sqrt decay (RQ5).

Finally, and in order to try to verify these conclusions, a statistical significance test has been applied for the measure $R@1$. More specifically, the McNemar test (McNemar 1947) was selected, which is a non-parametric test for paired data, as recommended in Dieterich (1998) for comparing machine learning algorithms. The significance threshold, α, has been set to 0.1. It has been run for families of subprofiles (for the three values of k and the best decay method, when applicable): $TempTop$ with 2Sqrt, $TopTemp$ with 2Sqrt and Top without decay. The idea is to first determine if there are significant differences in each family. The results of these tests fulfil the same pattern for $TempTop$ and $TopTemp$ families: there are no differences between the two top k values and there is with the worst. In the case of $TempTop$, there are no differences between 110 and 20, but there are with 400. For $TopTemp$, there are no differences

---

[16] Although in this case, the difference between None and 2Sqrt is smaller.

Table 2 *p* values of the McNemar's tests

Model	*TempTop* 110 2Sqrt	*TopTemp* 400 2Sqrt	Top 20 None	Temp 2Sqrt	Atomic 2Sqrt
TopTemp 400 2Sqrt	2.5e−16*	–	–	–	–
Top 20 None	8.3e−16*	0.1986	–	–	–
Temp 2Sqrt	1.4e−15*	0.1439	0.7117	–	–
Atomic 2Sqrt	1.1e−25*	0.0001*	0.4047	0.6245	–
Mono None	2.2e−30*	1.4e−08*	8.4e−09*	2.4e−10*	0.0001*

between 400 and 110, but there are for 20. In terms of Top, there are no differences between the three values of k. A final series of tests is run between the best values of each family, including in this case the baselines (*p* values are shown in Table 2, where the * means that there are significant differences): $TempTop$, 110, 2Sqrt; $TopTemp$, 400, 2Sqrt; Top, 20, None; $Temp$, 2Sqrt; $Atomic$, 2Sqrt; and $Mono$, None. The results show that there are significant differences between $TempTop$ and the others and also between $Mono$ and the others, and there are no differences between $TopTemp$, Top and $Temp$. And there are differences between $TopTemp$ and $Atomic$, but not between $Atomic$, Top and $Temp$. By way of summary, $TempTop$ is clearly the best option for combining temporal and topical dimensions and $Mono$ is the worst alternative.

5 Conclusions and further research

In this paper, we have focused on testing how useful temporality, topicality and the combination of these are for the problem of building and using profiles in the context of journal recommendation. We have proposed five research questions and tried to give answers to them by means of a detailed experimental design. We have proposed two different ways of hybridising temporal and topical dimensions with the aim of improving the capacity of organising the available information and the performance of this specific type of recommender. A biomedical journal collection on publication venue recommendation was used to test our proposals, in conjunction with several state-of-the-art models, and these revealed that the combination of these two types of approaches is a good alternative although it is important to note that order matters in terms of performance. From our experiments, we can conclude that the best option for our problem at hand is to create temporal partitions and discover the latent topics starting from the papers in each partition using LDA and then to construct the profiles. It is important to mention that the number of topics for building the profiles is not a critical parameter. The application of a decay factor might be a valuable aid but it clearly depends on the quality of the penalising function. In our context, 2Sqrt helps to improve the performance of the recommendation with hybrid subprofiles.

The findings revealed in this paper might not be directly extrapolated to a generic content-based recommender system, although we think that they could be useful, and a good starting point, to improve it.

In terms of future lines of research, we plan to explore other methods of combining temporal and topical dimensions to obtain better subprofiles. One alternative is the use of temporal topic models (Blei and Lafferty 2006; Dieng et al. 2019). Another option is to use methods based on the aggregation or fusion (Wu 2012) of topical and temporal rankings individually obtained by Top and $Temp$, respectively. Another research line is to design high quality decay functions that boost the performance of hybridisation. Finding optimal values for the number of topics for each topic-based profiling model and how they could impact in the performance of the recommendation model is another interesting area of investigation. Finally, we also plan to study the most suitable ways of explaining the recommendations (Tintarev and Masthoff 2012) offered by our models.

Declarations

Conflict of interest The authors have no relevant financial or non-financial interests to disclose.

References

Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. IEEE Trans. Knowl. Data Eng. **17**(6), 734–749 (2005)

Albusac, C., de Campos, L.M., Fernández-Luna, J.M., Huete, J.F.: PMSC-UGR: a test collection for expert recommendation based on PubMed and Scopus. In: Lecture Notes in Artificial Intelligence, vol. 11160. Advances in Artificial Intelligence, CAEPIA (2018)

Alhoori, H., Furuta, R.: Recommendation of scholarly venues based on dynamic user interests. J. Informetr. **11**, 553–563 (2017)

Aliannejadi, M., Crestani, F.: Personalized context-aware point of interest recommendation. ACM Trans. Inf. Syst. **36**(4O), 1–28 (2018)

Amini, B., Ibrahim, R., Othman, M.S., Selamat, A.: Capturing scholar's knowledge from heterogeneous resources for profiling in recommender systems. Expert Syst. Appl. **41**(17), 7945–7957 (2014)

Balog, K., Fang, Y., de Rijke, M., Serdyukov, P., Si, L.: Expertise retrieval. Found. Trends Inf. Retr. **6**, 127–256 (2012)

Bateja, R., Dubey, S., Bhatt, A.: A patient-centric healthcare model based on health recommender systems. In: Recent Findings in Intelligent Computing Techniques, pp. 269–276. Springer, Berlin (2018)

Becerra, C., Gonzalez, F., Gelbukh, A.: LinearTag models: recommendations using linear user profiles based on tags. Computación y Sistemas **21**(1), 115–127 (2017)

Benzarti, S., Faiz, R.: EgoTR: personalized tweets recommendation approach. In: Intelligent Systems in Cybernetics and Automation Theory, pp. 227–238, Springer, Berlin (2016)

Blei, D.M., Lafferty, J.D.: Dynamic topic models. In: Proceedings of the 23rd International Conference on Machine Learning, pp. 113–120 (2006)

Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. J. Mach. Learn. Res. **3**, 993–1022 (2003)

Bobadilla, J., Ortega, F., Hernando, A., Gutiérrez, A.: Recommender systems survey. Knowl.-Based Syst. **46**, 109–132 (2013)

Bogers, T.: Tag-based recommendation. Lect. Notes Comput. Sci. **10100**, 441–479 (2018)

Campos, P., Díez, F., Cantador, I.: Time-aware recommender systems: a comprehensive survey and analysis of existing evaluation protocols. User Model. User-Adap. Inter. **24**, 67–119 (2014)

Chen, C., Meng, X., Xu, X., Lukasiewicz, T.: Location-aware personalized news recommendation with deep semantic analysis. IEEE Access **5**, 1624–1638 (2017)

de Campos, L.M., Fernández-Luna, J.M., Huete, J.F.: A lazy approach for filtering parliamentary documents. Lect. Notes Comput. Sci. **9265**, 364–378 (2015)

de Campos, L.M., Fernández-Luna, J.M., Huete, J.F.: Committee-based profiles for politician finding. Int.. J. Uncertain. Fuzziness Knowl. Based Syst. **25**(Suppl. 2), 21–36 (2017)

de Campos, L.M., Fernández-Luna, J.M., Huete, J.F., Redondo-Expósito, L.: Automatic construction of multi-faceted user profiles using text clustering and its application to expert recommendation and filtering problems. Knowl. Based Syst. **190**, article number 105337 (2020)

de Campos, L.M., Fernández-Luna, J.M., Huete, J.F., Redondo-Expósito, L.: LDA-based term profiles for expert finding in a political setting. J. Intell. Inf. Syst. **56**(3), 529–559 (2021)

de Campos, L.M., Fernández-Luna, J.M., Huete, J.F.: Publication venue recommendation using profiles based on clustering. **10**, 106886–106896 (2022)

Dieng, A., Ruiz, F., Blei, D.: The dynamic embedded topic model, arXiv:1907.05545v2 (2019)

Dietterich, T.G.: Approximate statistical tests for comparing supervised classification learning algorithms. Neural Comput. **10**(7), 1895–1923 (1998)

Ding, Y., Li, X.: Time weight collaborative filtering. In: Proceedings of the 14th ACM International Conference on Information and Knowledge Management, pp. 485–492 (2005)

Errami, M., Wren, J.D., Hicks, J.M., Garner, H.R.: eTBLAST: a web server to identify expert reviewers, appropriate journals and similar publications. Nucleic Acids Res. **35**, W12–W15 (2007)

Firan, C.S., Nejdl W., Paiu, R.: The benefit of using tag-based profiles. In: Proceedings of the LA-WEB conference, pp. 32–41 (2007)

Fu, Y., Shen, B., Chen, Y., Huang, L.: TDMatcher: A topic-based approach to task-developer matching with predictive intelligence for recommendation. Appl. Soft Comput. **110**, 107720 (2021)

Gauch, S., Chandramouli, A., Micarelli, A.: User profiles for personalized information access. In: The Adaptive Web: Methods and Strategies of Web Personalization, pp. 54–89. Springer, Berlin (2007)

Gulla, J.A., Fidjestøl, A.D., Su, X., Castejon, H.: Implicit user profiling in news recommender systems. In: Proceedings of the 10th International Conference on Web Information Systems and Technologies, vol. 1, pp. 185-192 (2014)

Huang, X., Wu, F.: A novel topic-based framework for recommending long tail products. Comput. Ind. Eng. **137**, 1–15 (2019)

Jelodar, H., Wang, Y., Yuan, C., Feng, X., Jiang, X., Li, Y., Zhao, L.: Latent Dirichlet allocation (LDA) and topic modeling: models, applications, a survey. Multimedia Tools Appl. **78**, 15169–15211 (2019)

Kang, N., Doornenbal, M., Schijvenaars, B.: Elsevier journal finder: recommending journals for your paper. RecSys **15**, 261–264 (2015)

Karlsen, R., Elahi, N., Andersen, A.: Personalized recommendation of socially relevant images. In: Proceedings of the 8th International Conference on Web Intelligence, Mining and Semantics, Article number 41 (2018)

Kaufman, L., Rousseeuw, P.J.: Finding groups in data: an introduction to cluster analysis. Wiley, New York (1990)

Kaya, M., Bridge, D.: Subprofile-aware diversification of recommendations. User Model. User-Adap. Inter. **29**, 661–700 (2019)

Khan, M., Smyth, B., Coyle, D.: Addressing the complexity of personalized, context-aware and health-aware food recommendations: an ensemble topic modelling based approach. J. Intell. Inf. Syst. **57**, 229–269 (2021)

Kook, H.J.: Profiling multiple domains of user interests and using them for personalized web support. In: Proceedings of the International Conference on Intelligent Computing, pp. 512–520 (2005)

Koren, Y.: Collaborative filtering with temporal dynamics. Commun. ACM **53**, 89–97 (2010)

Larrain, S., Trattner, C., Parra, D., Graells-Garridoe, E., Nørvåg, K.: Good times bad times: a study on recency effects in collaborative filtering for social ragging. In: Proceedings of the 9th ACM Conference on Recommender Systems, pp. 269–272 (2015)

Li, L., Zheng, L., Li, T., LOGO: a long-short user interest integration in personalized news recommendation. In: Proceedings of the 5th ACM Conference on Recommender Systems, pp. 317–320 (2011)

Li, L., Zheng, L., Yang, F., Li, T.: Modeling and broadening temporal interest in personalized news recommendation. Expert Syst. Appl. **41**, 3168–3177 (2014)

Lin, S., Hong, W., Wang, D., Li, T.: A survey on expert finding techniques. J. Intell. Inf. Syst. **49**, 255–279 (2017)

Linda, S., Bharadwaj, K.: A genetic algorithm approach to context-aware recommendations based on spatio-temporal aspects. In: Integrated Intelligent Computing. Communication and Security, pp. 59–70. Springer, Berlin (2019)

Liu, X., Croft, W.B., Koll, M.: Finding experts in community-based question-answering services. In: Proceedings of the 14th ACM International Conference on Information and Knowledge Management, pp. 315–316 (2005)

Liu, X.: Modeling users' dynamic preference for personalized recommendation. In: Proceedings of the 24th International Joint Conference on Artificial Intelligence, pp. 1785–1179 (2015)

Lops, P., Jannach, D., Musto, C., Bogers, T., Koolen, M.: Trends in content-based recommendation. User Model. User-Adap. Inter. **29**, 239–249 (2019)

Lu, J., Wu, D., Mao, M., Wang, W., Zhang, G.: Recommender system application developments: a survey. Decis. Support Syst. **4**, 12–32 (2015)

Luong, H., Huynh, T., Gauch, S., Do, L., Hoang, K.: Publication venue recommendation using author networks publication history. Lect. Notes Comput. Sci. **7198**, 426–435 (2012)

McGowan, J.P., Kushmerick, N., Smyth, B.: Who do you want to be today? Web personae for personalised information access. In: Web-Based Systems: Second International Conference, pp. 514–517 (2002)

McNemar, Q.: Note on the sampling error of the difference between correlated proportions or percentages. Psychometrika **12**(2), 153–157 (1947)

Medvet, E., Bartoli, A., Piccinin, G.: Publication venue recommendation based on paper abstract. In: Proceedings of the 26th IEEE International Conference on Tools with Artificial Intelligence, pp. 1004–1010 (2014)

Narducci, F., Musto, C., Semeraro, G., Lops, P., de Gemmis, M.: Exploiting big data for enhanced representations in content-based recommender systems. In: Proceedings of the 14th International Conference on Electronic Commerce and Web Technologies, pp. 182–193 (2013)

Narducci, F., Basile, P., Musto, C., Lops, P., Caputo, A., de Gemmis, M., Iaquinta, L., Semeraro, G.: Concept-based item representations for a cross-lingual content-based recommendation process. Inf. Sci. **374**, 15–31 (2016)

Neshati, M., Fallahnejad, Z., Beigy, H.: On dynamicity of expert finding in community question answering. Inf. Process. Manag. **53**(5), 1026–1042 (2017)

Ngaffo, A., El Ayeb, W., Choukair, Z.: A time-aware service recommendation based on implicit trust relationships and enhanced user similarities. J. Ambient. Intell. Humaniz. Comput. **12**, 3017–3035 (2021)

Nishioka, C., Scherp, A.: Profiling vs. time vs. content: what does matter for top-k publication recommendation based on Twitter profiles? In: Proceeding of the Joint Conference on Digital Libraries pp. 171–180 (2016)

Pavan, M., Luca, E.W.D.: Semantic-based expert search in textbook research archives. In: Proceedings of the 5th International Workshop on Semantic Digital Archives, CEUR Workshop Proceedings, vol. 1529, pp. 18–29 (2015)

Pradhan, T., Pal, S.: CNAVER: a content and network-based academic venue recommender system. Knowl. Based Syst. **189**, 105092 (2020)

Ramos, C., Paraboni, I.: Temporal aspects of content recommendation on a microblog corpus. Lect. Notes Comput. Sci. **8775**, 189–194 (2014)

Ren, R., Zhang, L., Cui, L., Deng, B., Shil, Y.: Personalized financial news recommendation algorithm based on ontology. In: Proceedings of 3rd International Conference on Information Technology and Quantitative Management, Procedia Computer Science vol. 55, pp. 843–851 (2015)

Rollins, J., McCusker, M., Carlson, J., Stroll, J.: Manuscript Matcher: a content and bibliometric-based scholarly journal recommendation system. In: 5th International Workshop on Bibliometric-Enhanced Information Retrieval (BIR 2017), CEUR Workshop Proceedings, vol. 1823 (2017)

Rybak, J., Balog, K., Nørvåg, K.: Temporal expertise profiling. Lect. Notes Comput. Sci. **8416**, 540–546 (2014)

Saraswatm, M., Chakraverty, S., Sharma, A.: Review-based topic distribution profile for recommender systems. In: Advances in Data Sciences. Security and Applications, pp. 433–443. Springer, Berlin (2020)

Sharma, R., Gopalani, D., Meena, Y.: Concept-based approach for research paper recommendation. In: Proceedings of the Pattern Recognition and Machine Intelligence Conference, pp. 687–692 (2017)

Si, Y., Zhang, F., Liu, W.: CTF-ARA: an adaptive method for POI recommendation based on check-in and temporal features. Knowl.-Based Syst. **128**, 59–70 (2017)

Silva, T., Ma, J., Yang, C., Liang, H.: A profile-boosted research analytics framework to recommend journals for manuscripts. J. Am. Soc. Inf. Sci. **66**, 180–200 (2015)

Simsek, A., Karagoz, P.: Wikipedia enriched advertisement recommendation for microblogs by using sentiment enhanced user profiles. J. Intell. Inf. Syst. **54**, 245–269 (2020)

Somlo, G.L., Howe, A.E.: Incremental clustering for profile maintenance in information gathering web agents. In: Proceedings of the 5th International Conference on Autonomous Agents, pp. 262–269 (2001)

Stakhiyevich, P., Huang, Z.: Building user profiles based on user interests and preferences for recommender systems. In: Proceedings of the IEEE International Conferences on Ubiquitous Computing and Communications (IUCC) and Data Science and Computational Intelligence (DSCI) and Smart Computing, Networking and Services (SmartCNS), pp. 450–455 (2019)

Tintarev, N., Masthoff, J.: Evaluating the effectiveness of explanations for recommender systems. User Model. User-Adap. Inter. **22**, 399–439 (2012)

Wang, D., Liang, Y., Xu, D., Feng, X., Guan, R.: A content-based recommender system for computer science publications. Knowl.-Based Syst. **157**, 1–9 (2018)

Wangwatcharakul, C., Wongthanavasu, S.: A novel temporal recommender system based on multiple transitions in user preference drift and topic review evolution. Expert Syst. Appl. **185**, 115626 (2021)

Wartena, C., Slakhorst, W., Wibbels, W., Gantner, Z., Freudenthaler, C., Newell, C., Schmidt-Thieme, L.: Keyword-based TV program recommendation. In: Proceedings of the 9th Workshop on Intelligent Techniques for Web Personalization and Recommender Systems. Ceurs Workshop Proceedings, pp. 756–762 (2011)

Wu, S.: Data fusion in information retrieval. In: Adaptation, Learning, and Optimization, vol. 13, Springer, Berlin (2012)

Xiang, L., Yuan, Q., Zhao, S., Chen, L., Zhang, X., Yang, Q., Sun, J.: Temporal recommendation on graphs via long- and short-term preference fusion. In: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 723–732 (2010)

Yan, S., Chen, X., Huo, R., Zhang, X., Lin, L.: Learning to build user-tag profile in recommendation system. In: Proceedings of the 29th ACM International Conference on Information and Knowledge Management, pp. 2877–2884 (2020)

Yang, Z., Davison, B.D.: Venue recommendation: submitting your paper with style. In: Proceedings of the 11th International Conference on Machine Learning and Applications, vol. 1, pp. 681–686 (2012)

Yeniterzi, R., Callan, J.: Moving from static to dynamic modelling of expertise for question routing in CQA sites. In: Proceedings of the 9th International AAAI Conference on Web and Social Media, pp. 702–705 (2015)

Yeung, C., Gibbins, N. Shadbolt, N.: Multiple interests of users in collaborative tagging systems. In: Weaving Services and People on the World Wide Web, Springer, pp. 255–274 (2009)

Yin, H., Cui, B., Chen, L., Hu, Z., Zhou, X.: Dynamic user modeling in social media system. ACM Trans. Inf. Syst. 33(3), Article 10 (2015)

Zeng, H.Z., Chen, Z., Ma, W.: A unified framework for clustering heterogeneous web objects. In: Proceedings of the 3rd International Conference on Web Information Systems Engineering, pp. 161–172 (2002)

Zeng, W., Du, Y., Zhang, D., Ye, Z., Dou, Z.: TUP-RS: temporal user profile based recommender system. Lecture Notes Artif. Intell. **10842**, 463–474 (2018)

Ziaimatin, H., Groza, T., Bordea, G., Buitelaar, P., Hunter, J.: Expertise profiling in evolving knowledge-curation platforms. J. Comput. **2**(3), 118–126 (2012)