A Formal Framework for Hypergraph-Based User Profiles

Hilal Tarakci and Nihan Kesim Cicekli

Abstract In this study, we propose a formal framework for user profile representation with hypergraphs. We exploit the framework to aggregate partial profiles of the individual to obtain a complete, multi-domain user model, since we aim to model the user from several perspectives. We use Freebase commons package concepts as predefined domains. The proposed user model is also capable of extracting *user domain capsules*, which models the user for the domain of interest. Moreover, using a hypergraph data structure results in solving connection-based problems easily, since the cost of local operations on a graph is low and independent of the size of the whole graph. Many problems in user modelling domain are connection-based problems, such as recommendation.

Keywords User modelling · User profile · Hypergraph user model

1 Introduction

The popularity of social networking sites has dramatically increased over the last decade. The user's activities on social websites such as his likes, comments, location declarations and friendships reveal important information about his profile. The individual's interests, goals and preferences can be exposed by mining those activities. Social networks differ in nature and are used for different purposes [1]. Therefore, mining separate social networks independently results in partial profiles of the user which merely represents user's interests for one or few domains. In this study,

This work is partially supported by The Scientific and Technical Council of Turkey Grant "TUBITAK EEEAG-112E111".

H. Tarakci (✉)
Department of Computer Engineering, Sakarya University, Sakarya, Turkey
e-mail: htarakci@sakarya.edu.tr

N.K. Cicekli
Department of Computer Engineering, Middle East Technical University, Ankara, Turkey
e-mail: nihan@ceng.metu.edu.tr

T. Czachórski et al. (eds.), *Information Sciences and Systems 2014*,
DOI 10.1007/978-3-319-09465-6_30

we present a framework to aggregate partial profiles of the individual to obtain a complete, multi-domain user model.

Representing a user profile with graph is a common strategy. The vertices usually represent the items and the users where an edge between a user and an item indicate user's interest on that item. Since the graph is only capable of representing binary relations, other approaches have been proposed for handling higher order relations in user modelling domain. There are a few studies which define user model as bipartite [2] and tripartite graphs [3]. In general, if the number of vertex types n is known in advance and the relations in the user model are binary, an n-partite graph is capable of representing the profile. However, if there are higher-order relations, a hypergraph is more appropriate to represent the user model [4, 5].

In a previous paper, we presented the initial ideas for using hypergraph in the modelling of user profiles [6, 7]. In this paper, the main contribution is a formal framework for hypergraph-based user profiles. It is claimed that aggregating profiles solves the cold-start problem and sparse user model problem [1]. Seamless aggregation of partial user profiles obtained from different knowledge sources is still an unsolved problem. We claim that the proposed hypergraph user model is effective in solving the aggregation of partial profiles.

The paper is organized as follows. Section 2 summarises the related studies. Section 3 formally defines the proposed hypergraph based user model. The application and evaluation details are presented in Sect. 4. Section 5 concludes the paper by summarizing the study.

2 Related Work

In [1], form-based and tag-based profiles are managed separately. The former is a list of attribute-value pairs whereas the latter is a set of weighted tags. The aggregation strategy for form-based profiles is unifying sets of attribute-value pairs. Heterogeneous attribute vocabularies is resolved by using an alignment function, which maps profiles to unified attribute-value space. However, this alignment function may result in duplicate entries in the final user profile. Moreover, when there are conflicts in the aggregated profiles, both values are included in the result. The aggregation of tag-based profiles is accomplished by taking a weighted accumulation of partial tag-based profiles. The authors do not consider aggregating tag-based profiles and form-based profiles with each other. In our paper, we do not make such a distinction. We seamlessly aggregate received partial user profiles by taking their weighted accumulation. We solve heterogeneous vocabulary problem by using Freebase.[1]

In [8], during aggregation the authors address the problem of recurring items and calculating a global weight for them. To achieve this, they keep track of *provenance data* which is the meta data for the user profile item such as the source of the item and the timestamps. This enables the recalculation of item weights during aggregation

[1] Freebase, https://www.freebase.com/.

of the partial profiles. We also keep track of the provenance data by storing the knowledge source, the short term profile date and the exact keyword of the item. We extend this information each time the item and user is bounded together.

3 Data Model and Problem Formulation

A hypergraph is the generalization of an ordinary graph by introducing hyperedges, which are non-empty subsets of the vertex set [9]. Vertices of a hypergraph represents the entities to be modelled such as people and concepts. Hyperedges represent the high-order relations between those entities. Besides hypergraphs, there are property graphs which contains key-value property pairs [10]. In a property graph each node and edge can have multiple key-value pairs whereas in a hypergraph, an edge can connect more than two nodes. Every hypergraph can be represented by a property graph by adding extra key-value pairs to annotate nodes, which are connected by the same hyperedge. For instance, the *Users* hyperedge is represented by assigning the value of the node's *type* as *User* for each user node in the property graph. In this paper, we actually use property graphs, since the graph database we adopted supports property graphs.

In this study, we focus on constructing a holistic user model by aggregating the short term profiles by utilizing the proposed hypergraph data structure. The notations for the proposed hypergraph is summarised in Table 1. Basically the hypergraph user model consists of set of labelled nodes and strongly typed hyperedges. Nodes representing concepts and users are assigned different labels. Similarly, hyperedges responsible for representing the user's interest on an item or indicating the semantic relations between entities belong to different types. On top of these nodes and hyperedges, there are *domains* which divide the hypergraph to overlapping regions to group nodes and hyperedges in the same domain together. In other words, every concept node belongs to one or more domain. In the implementation, we use Freebase commons package as domains and define a *domain starter node* for each domain which connects to the nodes under that domain. The projection of the user in a domain is represented by a *user domain capsule*. The proposed hypergraph facilitates profile aggregation and semantic enhancement with the help of the presented user model structure. A simplified illustration for the hypergraph data structure is presented in (Fig. 1). The figure demonstrates that a user with name *dummyUser* is interested in the item *Pride and Prejudice* which is connected to the *fictional universes* domain. During the *semantic enhancement process*, *Jane Austen* is semantically related to the item with *CreatedBy* relationship. Moreover, the items defining the genre of *Pride and Prejudice* are connected with *HasGenre* relation.

Definition 1 *Partial User Profile*: Partial user profile L_{uts} is the short term profile obtained from the knowledge source s for the user u during time period t. The input of the system are received partial profiles. A partial profile is represented as a vector of terms $[\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_n]$.

Table 1 Our hypergraph user model

Notation	Description	Type
L_{uts}	Partial (short term) user profile for user u for a time period t from knowledge source s	A vector of terms
T_s	Uniform time period for receiving short term profiles from source s	Number of days
S	Set of knowledge sources	A set of strings (Facebook, LinkedIn etc.)
U	Set of all registered users	Nodes
U_f	Set of frequent users	Nodes
U_s	Set of semi-frequent users	Nodes
U_r	Set of rare users	Nodes
$f_\alpha(u)$	Profile categorization function for user u	A function
$f_{\alpha\text{-div}}(u)$	Function to calculate the diversity of the user profile	A function
$f_{\alpha\text{-den}}(u)$	Function to calculate the density of the user profile	A function
$f_{\alpha\text{-act}}(u)$	Function to calculate the activity of the user profile	A function
Υ_f	The threshold value for being a frequent user	A double
Υ_s	The threshold value for being a semi-frequent user	A double
Υ_{div}	The threshold value for diversity	A double
Υ_{den}	The threshold value for density	A double
C	Set of entities (concepts)	Nodes
$D_{[d]}$	Domain starter node for each predefined domain d	Nodes
E_{bind}	Metadata for user-item (interest) relation	A hyperedge
E_{inner}	The semantic relation between items (entities and named entities)	A hyperedge
$\Upsilon_{\mathrm{inner}}$	The semantic relation threshold which defines the enhance limit	An integer
E_{domain}	The domain bind between domain starter node and items	A hyperedge
$\Upsilon_{\mathrm{domain}}$	Domain threshold value to decide the number of the domain connections to represent	An integer
$f_{\mathrm{ud}}(u, d)$	User domain capsule function	A function
f_{decay}	Profile decay function	A function
$f_{\mathrm{sim}}(c, u, d)$	Similarity function for concept and user domain profile	A function
$f_{\mathrm{userSim}}(u_1, u_2, d)$	Similarity function for users	A function
P_{u}	General (long term) user profile	A sub hypergraph
SemEnh	Algorithm for semantic enhancement	An algorithm
ProfAgg	Algorithm for profile aggregation	An algorithm

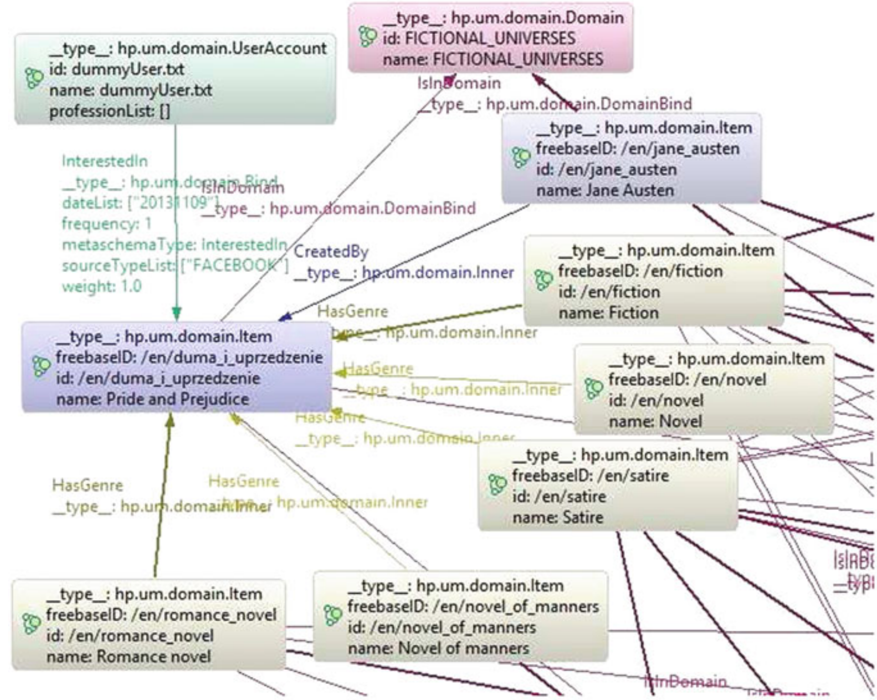
__type__: hp.um.domain.Domain
id: FICTIONAL_UNIVERSES
name: FICTIONAL_UNIVERSES
__type__: hp.um.domain.UserAccount
id: dummyUser.txt
name: dummyUser.txt
professionList: []
IsInDomain
__type__: hp.um.domain.DomainBind
__type__: hp.um.domain.Item
freebaseID: /en/jane_austen
id: /en/jane_austen
name: Jane Austen
InterestedIn
__type__: hp.um.domain.Bind
dateList: ["20131109"]
frequency: 1
metaschemaType: InterestedIn
sourceTypeList: ["FACEBOOK"]
weight: 1.0
IsInDomain
__type__: hp.um.domain.DomainBind
CreatedBy
__type__: hp.um.domain.Inner
__type__: hp.um.domain.Item
freebaseID: /en/fiction
id: /en/fiction
name: Fiction
HasGenre
__type__: hp.um.domain.Inner
__type__: hp.um.domain.Item
freebaseID: /en/duma_i_uprzedzenie
id: /en/duma_i_uprzedzenie
name: Pride and Prejudice
HasGenre
__type__: hp.um.domain.Inner
__type__: hp.um.domain.Item
freebaseID: /en/novel
id: /en/novel
name: Novel
HasGenre
__type__: hp.um.domain.Inner
__type__: hp.um.domain.Item
freebaseID: /en/satire
id: /en/satire
name: Satire
HasGenre
__type__: hp.um.domain.Inner
HasGenre
__type__: hp.um.domain.Inner
IsInDo
__typ
__type__: hp.um.domain.Item
freebaseID: /en/romance_novel
id: /en/romance_novel
name: Romance novel
__type__: hp.um.domain.Item
freebaseID: /en/novel_of_manners
id: /en/novel_of_manners
name: Novel of manners
IsInDomain

Fig. 1 Hypergraph data structure illustration

User Categorization: People have different social web usage habits. A user may be frequently active in social websites, whereas another may scarcely use his accounts. Even two frequent social web users may show differences in their usage behaviour. A user's actions may show broad interest in many domains whereas another may exhibit deep interest in few domains. Categorizing users according to their usage habits enables definition of ad hoc algorithms for each user type. Let U denote the set of all registered users. U consists of the union of frequent users U_{f}, semi-frequent users U_s and rare users U_r. Frequent users have well-defined profiles for probably many domains whereas semi-frequent users have defined profiles for few domains. Rare users consist of new users and users who barely use their social web accounts. The category to which the user belongs may change in time according to a *profile categorization function* $f_\alpha(u)$ and two threshold values. $f_\alpha(u)$ is calculated as a weighted combination of three sub-functions: $f_\alpha(u) = x.f_{\alpha\text{-div}}(u) + y.f_{\alpha\text{-den}}(u) + z.f_{\alpha\text{-act}}(u)$ where x, y and z are non-negative impact factors and their sum is equal to 1. $f_{\alpha\text{-div}}(u)$ calculates the diversity of profile amongst several domains, $f_{\alpha\text{-den}}(u)$ the density of profile under a specific domain and $f_{\alpha\text{-act}}(u)$ the activity degree on the social web accounts of the user. $f_{\alpha\text{-div}}(u)$ computes the diversification of the user's profile over domains by calculating the number of domains the user have items more than a threshold Υ_{div}. Namely, users who have items distributed in many domains have high $f_{\alpha\text{-div}}(u)$ values. $f_{\alpha\text{-den}}(u)$ computes the deepness of the user's profile in one particular domain. It is computed by calculating the number of domains user have items more than a threshold Υ_{den} where $\Upsilon_{\text{den}} > \Upsilon_{\text{div}}$. In other words, $f_{\alpha\text{-den}}(u)$ value is high for users whose profiles are defined in detail for a number of domains. $f_{\alpha\text{-act}}(u)$ computes the recent update rate of the user. It is calculated when the latest short-term profile for the user is received. The score is based on the number of modifications and extensions applied to the original user model. When the value of the profile categorization function $f_{\alpha}(u)$ is above a threshold Υ_{f}, the user is classified as a frequent user. If the score is between Υ_{f} and $\Upsilon_{s} < \Upsilon_{\text{f}}$, the user is a semi-frequent user. Otherwise, the user is categorised as a rare user.

Domains: Our proposed hypergraph aims to model the user from several perspectives. In order to achieve this, we use Freebase commons package concepts as predefined domains. In fact, Freebase also introduces these concepts as domains on its home page. Domains are represented with separate *domain starter nodes*. Let $D[d]$ denotes the *domain starter node* for the domain with name d. For instance, D_{tennis}, D_{sports}, $D_{\text{fictional-universes}}$ represents starter nodes for domains *tennis*, *sports* and *fictional universes* domains, respectively. The domains may overlap with each other. This situation does not lead to a problem, since we handle each domain as a separate projection of the user's profile.

Definition 2 *User Domain Capsule*: User domain capsule of the user u for the domain of interest d is the sub hypergraph which *maximally covers* the user under the domain d. The proposed user domain capsule resembles the *news capsule* presented in [4], which is constructed by partitioning the hypergraph into a predefined number of sub-graphs. *News capsules* are not per-user in order to enable inference on the graph for other users. In our study, we use the *capsule* notion in a different way, to obtain a compact structure to capture the user's profile for a particular domain. To obtain *user domain capsules*, the item nodes which are connected to the domain and reachable from the user are collected. When the user u is connected with an item $c \in C$, the provenance data should be kept to use the item's history in the weight calculation algorithm. The weight calculation algorithm computes the interest of the user on an item by considering the provenance data. For instance, as the time passes, the weight of the interest decays. E_{bind} hyperedge type is used to keep the provenance data. For the relations between concepts E_{inner} hyperedge type is used. The relation type between the concepts in Freebase under the domain of concern is kept in the property *freebase Relation*. We used a subset of *Freebase metaschema properties* to model the semantics between the concepts.

Weight and Similarity: The *user domain capsule* of the user u for the domain of concern d is calculated by a function $f_{\text{ud}}(u, d)$. The function returns a vector of concept-weight pairs which represents the user's projection on the domain of interest in vector space model. f_{decay} function ensures that the weight of the most recently created or updated concept is supported more than older profile items. In order to decide whether the user is interested in a concept, the similarity between the concept and user profile is calculated according to the selected similarity metric

[11, 13]. There are similarity calculation approaches including measuring semantic similarity between words using web documents [11, 12]. In this study, we define a similarity function $f_{\text{sim}}(c, u, d)$ which considers both similarity and semantic relatedness. The function moves the user profile to vector space model by obtaining the *user domain capsule* using $f_{\text{ud}}(u, d)$ and calculates the similarity score based on the cosine similarity between the concept c and the concepts in the domain user profile. In order to compute the similarity of two users under a domain, we define $f_{\text{userSim}}(u_1, u_2, d)$ which takes the user domain capsule that has fewer concepts as pivot and calculate $f_{\text{sim}}(c, u, d)$ score for each c in the pivot user domain capsule and make a weighted accumulation of the highest, lowest and average similarity scores.

Profile Aggregation: We receive short term profiles for users on a regular basis. To obtain a complete multi-domain profile of the user, short-term profiles are aggregated by using the *profile aggregation function* $f_p(u, L_{\text{uts}})$.

Definition 3 *User Profile*: The user profile P_{u} is the aggregated user model for the user u and it is the hypergraph which consists of the user u, the interest nodes of u and the hyperedges between them. $f_p(u, L_{\text{uts}})$ takes the short term profile of the user as input and outputs the general user profile denoted as P_{u}. *Profile aggregation function* aggregates the short term profile by the following algorithm:

```
foreach term t in L_uts:
 disambiguate term t from knowledge base.
 if the item is already in the hypergraph:
    if the item is already connected to the user:
    update provenance data.
    else: create a bind between the user and the item.
 else: create the item and connect it with the user.
       decide domains for the item from knowledge base.
       connect the item to the domain starter nodes of its
       domains.
       enhance the item by using the middle ontology.
       foreach enhancing item:
          create node, decide domains.
          connect the item with the enhancing item semantically.
retrieve the user and reachable item nodes, output P_u
```

4 Application and Evaluation of Formal Framework

The initial dataset is prepared by collecting short term profiles from Facebook accounts of 204 users during two months by mining page likes. 12 short term profile sets are constructed by taking the time period as 3,4 or 7 days. Since the number of users is small, user categorization is not applied and concepts and named concepts are not discriminated. During evaluation, each user is extracted from the dataset and the hypergraph is populated with the remaining user. Afterwards, during aggregation of the user to the prevously populated hypergraph, when the item is already in the graph, this is considered as a hit. For 204 users, the average of hits-to-total items ratio is calculated as *0.61*. In the baseline, the knowledge base usage and enhancement is removed and the same data is evaluated. The average hits-to-total ratio for the baseline is *0.25*. The resulting scores show that usage of a knowledge base and the enhancement procedure successfully predicts the user's future interests. The dataset is prepared by collecting only page likes; using other social activities may result in more accurate short term profiles. We are going to improve our dataset by collecting users from public feeds of social websites and analyse them for a longer period. Furthermore, we are going to accomplish more detailed and comprehensive evaluations.

5 Conclusions

In this paper, we presented a formal framework for managing a hypergraph user model. We enabled seamless aggregation of partial user profiles with the help of the semantic enhancement of short term profile items. During semantic enhancement, the short-term profile terms become semantic nodes in the graph and the item nodes are attached to their domains and other related items with specialised hyperedges. Usage of domains enable extraction of *user domain capsules*, which are domain projections of users' profiles. Moreover, a number of user modelling domain problems are connected-data problems which could be solved easily by using a graph data structure. As future work, we are going to evaluate the framework against a bigger dataset and implement and evaluate a recommendation case study, which uses the proposed system.

References

1. F. Abel, E. Herder, G.J. Houben, N. Henze, D. Krause, Cross-system user modeling and personalization on the social web. User Model. User-Adapt. Interact. **23**(2–3), 169–209 (2013)
2. A. Tiroshi, S. Berkovsky, M.A. Kaafar, T. Chen, T. Kuflik, Cross social networks interests predictions based ongraph features, in *Proceedings of the 7th ACM Conference on Recommender Systems (RecSys '13)* (ACM, New York, 2013), pp. 319–322
3. B. Chen, J. Wang, Q. Huang, T. Mei, Personalized video recommendation through tripartite graph propagation. in *Proceedings of the 20th ACM International Conference on Multimedia* (ACM, 2012), pp. 1133–1136
4. L. Li, T. Li, News recommendation via hypergraph learning: encapsulation of user behavior and news content. in *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining* (ACM, 2013), pp. 305–314
5. T. Kramr, M. Barla, M. Bielikov, Personalizing search using socially enhanced interest model, built from the stream of user's activity. J. Web Eng. **12**(1–2), 65–92 (2013)

6. H. Tarakci, N.K. Cicekli, Ubiquitous fuzzy user modeling for multi-application environments by mining socially enhanced online traces. in *User Modeling, Adaptation, and Personalization* (Springer, Berlin Heidelberg, 2012), pp. 387–390
7. H. Taraki, N.K. Cicekli, UCASFUM: A Ubiquitous context-aware semantic fuzzy user modeling system. In *KEOD* (2012), pp. 278–283
8. F. Orlandi, J. Breslin, A. Passant, Aggregated, interoperable and multi-domain user profiles for the social web. in *Proceedings of the 8th International Conference on Semantic Systems* (ACM, 2012), pp. 41–48
9. G. Gallo, G. Longo, S. Pallottino, S. Nguyen, Directed hypergraphs and applications. Discret. Appl. Math. **42**(2), 177–201 (1993)
10. I. Robinson, J. Webber, E. Eifrem, *Graph Databases* (O'Reilly Media Inc., Sebastopol, 2013)
11. S.A. Takale, S. Nandgaonkar, Measuring semantic similarity between words using web documents. Int. J. Adv. Comput. Sci. Appl. IJACSA **1**(4), 78–85 (2010)
12. L. Zhiqiang, S. Werimin, Y. Zhenhua, Measuring semantic similarity between words using wikipedia. in *Web Information Systems and Mining, 2009. WISM 2009. International Conference on* (IEEE, 2009), pp. 251–255
13. P. Ilakiya, M. Sumathi, S. Karthik, A survey on semantic similarity between words in semantic web. in *Radar, Communication and Computing (ICRCC), 2012 International Conference on* (IEEE, 2012), pp. 213–216