A Novel, Serendipitous and Dynamic User-Centric Recommender Algorithm

Tatenda Kavu[1], Kudakwashe Dube[2], Peter Raeth[3] and Gilford Hapanyengwi[4]

{tkavu@cut.ac.zw[1],k.Dube@massey.ac.nz[2], petere_raeth@ameritech.net[3]}

University of Zimbawe, 167 Mt. Pleasant. Harare,Zimbabwe[1,3,4],Massey University, Tennent Drive Palmerston North 4474. New Zealand[2]

Abstract. Information filtering for web service using machine learning has recently grown widely , since information overload has also becoming a serious problem on the World Wide Web. Recommender systems were designed to cater for this problem, but published recommender systems still fail to cope with changes of user's preferences. This paper summarizes a research that is still going on, to solve the lack of novelty, serendipity and dynamism in recommender systems. Recent research has demonstrated different methodologies to create recommender systems , unfortunately many of these which were evaluated using user-centric evaluation frameworks fall short to fulfill users' satisfaction. Therefore we propose a unique computational method to create a novel, serendipitous and dynamic recommender system. We used web content mining to gather user profiles from social media, model these profiles, and create an algorithm to suggest user preferences. The results testify that many users' social profiles for Zimbabweans dominate quite well to determine user preferences. Therefore recommender developers for developing countries, has to gather user's social profiles to predict their preferences . The main contribution is a holistic approach to model and predict dynamic user-specific preferences from categorized social media profiles namely: social, psychological, cultural, and economic profiles

Keywords: **R**ecommender system, user-centric, profile, algorithm

1 Introduction

To generate recommendations for users, recommenders use two main methods to gather user data: explicit and implicit [1],[2]. Explicit methods gather user data using ratings, reviews and votes, whilst implicit methods make use of click-streams, purchases tracking and previous recommendations. These methods cannot give detailed cognitive information about users' preferences, resulting in recommendations that cannot cope with users' dynamic preferences[1]. It is clear from recommender algorithm experiments that recommendation methods are heavily affected by dynamics of user preferences and lack of interest by users to supply information. We need to investigate ways to enhance these methods [1],[3]. Current research has demonstrated that recommender systems are static in their recommendation strategies since they wholly depend on user's explicit and implicit data (i.e. ratings, like/dislike,

ACRID 2017, June 20-21, Victoria Falls, Zimbabwe

DOI 10.4108/eai.20-6-2017.2270015

event logs). Such recommendation strategies have given birth to approaches such as collaborative filtering and content-based filtering. These approaches face serious challenges to evolve with users' profiles , thus they are not dynamic enough to keep up with users' evolving preferences[4] and they fail to offer proper serendipitous and novel recommendations[5],[1].Users choose products/items based on usability or cost, consequences of buying the item, experience with the product, feelings based on experience and social impact of the item. All these factors demand attention when making recommendations[6].

The rest of the paper is organized as follows: Related Work shows user data collection methods and illustrates the problems with current recommendation methods. Approach demonstrates the proposed methods to give solutions to existing problems. Results and Analysis section shows the data that was collected and its analysis. Evaluation section demonstrate the evaluation of the algorithm. The paper finishes with a Conclusion and a projection of Future Work

2 Related Work

Many of the challenges that occur in recommender systems are there because current methods treat a user as a virtual black box. Algorithms do not employ the decision making processes of a user for recommendations. They merely depend on users' actions to predict preferences. Yet recent research exposes the advantages of opening the black box in order to know the changes of user preferences[7], [8]. Previous researches reveal that it might be impossible to offer accurate recommendations without considering user actions. A previous investigation done by[7] implies that a holistic approach that considers user actions together with detailed information from the user can give better recommendations. In this way, recommendations should be more accurate and can cope with evolving preference changes.

[9] demonstrates how psychometric surveys can give almost the same result as social media analysis when predicting users' preferences on different brands based on their personal traits. It has became clear from contemporary research that people ignore surveys or they don't have time to answer surveys. In light of this development social media has come as a rescuer since we can gather massive unstructured data, process it and get useful insights about individuals. Chao Yang and his team gathered specific user's personal traits from both psychometric surveys and tweets from Twitter. They found that after mining the same individual's account from twitter and analysing their sentiments they were able to predict the individual's brand preferences with an 86% accuracy rate.

[10] did a research which had profound implications in recommender systems. They found that there is a strong correlation between users' social media profiles and their e-commerce behaviors. They also found that user's profile information in a social network (for example Facebook) can be leveraged to predict what categories of products the user will buy from eBay Electronics.

3 Approaches to the Design of a Novel, Serendipitous and Dynamic User-Centric Recommender Algorithm

A user's decision making process give substantial information that can be used to create recommendations. That information can be used to offer dynamic recommendations that cope up with user preference changes and also to offer serendipitous and novel recommendations, since the recommender system will be predicting the decisions of a user.

From literature and observations we have found that decision making processes are influenced by four user profiles i.e (social , cultural , psychological and economic background(profile)) . Social media platforms and many other systems that can work with recommender systems can provide developers or data scientists with such user profiles as shown on Table 1. Table 1 shows the user profile variables that can be accessed from social media like Facebook. We categorised these profile variables into the major four profile categories.

Table 1. User profile categories that can be accessed on social media

Social	Cultural	Pyschological	Economic
Relationship status	User_about_me	quotes	currency
Age_range	User_hometown	religion	User_education_history
Gender	Friends_location	Favorite_atheletes	User_work_history
Education	User_location	User_birthday	Occupation
Education Interested in Political Religion	Timezone Languages	Friends_birthday	Car_type

Similarity between user profiles is calculated using the jaccard similarity principle as illustrated below. The JACCARD similarity metric, defined as the function J, is used to calculate the similarity between people's profiles. That is, the similarity between two people's profiles p1, p2 is the Jaccard metric between their two profiles(P)

$$J(p_1,p_2)= (|P(p_1) \cap P(p_2)|/|P(p_1) \cup P(p_2)|) \quad \mathbf{(1)}$$

Using Jaccard coefficient to Calculate Similarities between $User_i$ and $User_n$ profiles,

$$J(u_i,u_n)=(S_i\cap S_n)/(S_i\cup S_n)+(C_i\cap C_n)/(C_i\cup C_n)+(P_i\cap P_n)/(P_i\cup P_n)+(E_i\cap E_n)/(E_i\cup E_n) \quad \textbf{(2)}$$

where $0 <= J_{(ui,un)} <= 1$

If $J_{(ui,un)}$ is 1 it means these profiles are similar to each other if it is zero (0) it means there is a no similarity between the profiles. S_i,, C_i, P_i, E_i represents Social, Cultural, Psychological and Economic profiles respectively . The Pseudo Codes of the algorithms are in Appendix A.

Generic Algorithm:

```
begin

  If(UserAction in [click,search])
      findSimilarUsersWhoActioned[sameItems]
UsingAssociationRuleMining.append[mostSimilarProfiles]
      For user in[ mostSimilarProfiles]
            findItemsActionedby[mostSimilarProfiles]

            RecommendTheseItemsUser.

  end
```

Domain Specific Algorithm(Rent A Space Application): The algorithm was implemented in an Android application Rent A Space , to recommend houses to tenants who are looking for houses to rent, and this is how it was implemented

```
begin
if(userEnterHousePreferences)
      FindoptimalHousesUsing(Stable Marriage Problem)
If(Optimal houses<10)
      callGenericAalgorithm
else
      recommendOptimalHouses.
end
```

4 Results and Analysis

Data that was collected from the Rent A Space application[11] over a period of two weeks , this data was analyzed to find and test the following information:

1. From the user profiles categorized into(Social, Cultural, Psychological and Economic) , which among the four dominantly determine the user's preference
2. How did the algorithm accurately predict users' preferences
3. Where the recommendations novel ?
4. Where the recommendations serendipitous?

Table 2. Sample of Results Collected :User profile and preferences recorded

user_id	gender_id	category_id	employment_class_id	gross_salary	vehicle_value	ensuite	open_fireplace
400	1	3	1	2500	10000	0	0
400	1	3	1	2500	10000	1	0
431	1	2	4	2000	50000	0	0
460	2	1	1	1890	20000	1	0
498	1	1	5	3000	0	0	0

balcony	broadband_internet	builtin_wardropes	garage	borehole	watertank	fully_fenced	alarm_system
0	0	0	0	0	0	0	0
0	1	0	0	0	1	0	0
0	1	1	1	1	0	0	0
0	1	0	1	0	1	0	0
0	1	1	1	0	0	1	0

4.1 Association Rule Mining

Association rule mining was applied to determine frequent sets, to find the frequent items which are actioned by most similar profiles to a specific user, and finally to realize the most dominant user profiles which determine house preferences. According to the principles of association rule mining, some functions were calculated which are support , confidence and lift. Support is the probability of transactions of Antecedent(user profile) with Consequent(house preferences). Confidence is the minimum number of transactions that has consequent with the same antecedent(same user profiles with same house preferences). Lift is the measure of how more likely a tenant with a profile A will prefer a house with specifications B than otherwise.

$$\text{Support}(A\text{->}B):P(A\cap B) \tag{3}$$

$$\text{Confidence}(A\text{->}B) : P(B/A)=P(A\cup B)/P(A) \tag{4}$$

$$\text{Lift}(A\text{->}B): (A\cup B)/P(B) = P(A\cup B)/ P(A)(B) \tag{5}$$

Table 3 summarizes the data that was used and some analysis that was done, we took a sample of 34 transactions from different users(tenants) with 20 variables(which comprises of user profiles and user preferences). We found 15 rules which means 15 user profiles were found to be more dominant to determine to user preferences, the support was 4 and the minimum confidence was 50%.

Table 3. Summary of analysis

# Transactions in Input Data	34
# Variables in Input Data	20
# Association Rules	15
Minimum Support	4
Minimum Confidence	50.00%

Table 4. Sample of Association Rules and analysis

Confi dence %	Antecedent(A) (Gender,categ	Consequent©(ens uite,fireplace,swi mmingpool,carpo	Support for A	Support for C	Support for A&C	Lift ratio

	ory,employment_class,salary,vehicle_value,location etc	rt,walled,balcony, internet,garage,borehole,alarm etc)				
81.25	1&2&50000	0&600	16	13	13	2.125
81.25	0&2&50000	1&600	16	13	13	2.125
100	1&1000&50000	600	13	13	13	2.615

In summary , From the Association rule mining , we managed to find out that tenants of the same gender, employment class, range of vehicle value, family setup has a chance of a minimum of 81.25% of selecting a recommended house of the same area with a variety of specifications limited to (number of rooms, price, walled, borehole). Therefore If the tenants has the same social profile (gender, employment class, family set-up) they were likely to prefer the same range of houses differing mostly in these house specifications (i.e number of rooms, price, walled, borehole).

5 Evaluation

The recommender algorithm proved to be working so well in terms of categorising user's profiles into four categories(Social, Cultural, Psychological, Economic) and determine the most dominant of these categories per tenant(user) as a method to predict the preferences of the user. We moved on to test the algorithm using conventional methods used to evaluate recommender algorithms as demonstrated below.

$$\text{Precision(Hit Rate)} = tp/tp+fp=24/(24+44) = 35\% \qquad \textbf{(6)}$$

True-Positive (*tp*) refers to the recommended houses which the user views and True-Negative (*tn*) refers to non-recommended houses which were not viewed by the user. Whilst False-Positive (*fp*) refers to the recommended houses which the user did not view, and False-Negative (*fn*) Refers to non-recommended houses which the user views. From the data collected we calculated the the precision as illustrated in equation 6 above.

Measuring Serendipity

Average number of recommendations: R
Average number of obvious recommendations: $q = (U_u=1^n)(n)$

Number of non-obvious recommendations=R-q

Serendipity : count([for i in [R-q]: Where [R-q] are useful]) **(7)**

Serendipity sometimes is difficult to measure however, it is determined by the user. The user determines from the recommendation list, how many items surprise him/her. We found that out that serendipity and diversity comes from the fact that , the algorithm create a neighborhood based on the similarity of profiles and user actioned , and that on its, bring the serendipity and novelty that we were looking for.

6 Conclusion and Summary

Social profiles of user's seems to be the most dominant when it comes to the choice of a user , as we have found out with Zimbabwean residents. However this is yet to be proven on a global scale . This can help recommender systems developers in their design methodologies such that recommender systems can be in a position to offer dynamic, serendipitous and novel recommendations that satisfy users in any platform. In our future work , we wish to venture into ranking algorithms that utilise user's (social, cultural, psychological and economic) profiles to rank items selected from these profiles. It is imperative for recommender systems to use user's decision making processes to predict user's preferences since , these are the major factors that influence decisions made by users. In our future work , we wish to venture into ranking algorithms that utilise user's (social, cultural, psychological and economic) profiles to rank items selected from these profiles.

References

[1] J. Beel, C. Breitinger, S. Langer, A. Lommatzsch, and B. Gipp, "Towards reproducibility in recommender-systems research," *User Model. User-adapt. Interact.*, vol. 26, no. 1, pp. 69–101, Mar. 2016.

[2] P. Pu, L. Chen, and R. Hu, "Evaluating recommender systems from the user's perspective: Survey of the state of the art," *User Model. User-adapt. Interact.*, vol. 22, no. 4–5, pp. 317–355, 2012.

[3] F. O. Isinkaye, Y. O. Folajimi, and B. A. Ojokoh, "Recommendation systems: Principles, methods and evaluation," *Egypt. Informatics J.*, vol. 16, no. 3, pp. 261–273, Nov. 2015.

[4] K. Kapoor, K. Subbian, J. Srivastava, and P. Schrater, “Just in Time Recommendations,” in *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining - WSDM ’15*, 2015, pp. 233–242.

[5] M. D. Ekstrand, D. Kluver, F. M. Harper, and J. A. Konstan, “Letting Users Choose Recommender Algorithms : An Experimental Study,” *Proc. 9th ACM Conf. Recomm. Syst. - RecSys ’15*, pp. 11–18, 2015.

[6] M. D. Ekstrand, J. T. Riedl, and J. A. Konstan, “Collaborative Filtering Recommender Systems,” *Found. Trends® Human–Computer Interact.*, vol. 4, no. 2, pp. 81–173, 2011.

[7] T. D. Kavu, K. Dube, P. G. Raeth, and G. T. Hapanyengwi, “A Characterisation and Framework for User-Centric Factors in Evaluation Methods for Recommender Systems,” *Int. J. ICT Res. Africa Middle East*, vol. 6, no. 1, pp. 1–16, 2017.

[8] S. Karimi, “A purchase decision-making process model of online consumers and its influential factor a cross sector analysis,” *Theses PhD*, pp. 1–326, 2013.

[9] C. Yang, S. Pan, J. Mahmud, H. Yang, and P. Srinivasan, “Using Personal Traits For Brand Preference Prediction,” *Proc. 2015 Conf. Empir. Methods Nat. Lang. Process.*, pp. 86–96, 2015.

[10] Y. Zhang and M. Pennacchiotti, “Predicting purchase behaviors from social media,” *Proc. 22nd Int. Conf. World Wide Web*, pp. 1521–1532, 2013.

[11] Edlight Simeon Maguvu, “RentaSpez - Android Apps on Google Play.” [Online]. Available: https://play.google.com/store/apps/details?id=com.rentaspace.rent_a_space&hl=en. [Accessed: 20-Feb-2018].

Appendix: A

```
#Main Algorithm
Module_2():
      Start:
            if(i in new_users[] not (actioned any item)):
                  get i s profile [S i ,C i ,P i ,E i ]
                  Module_1([S   i   ,C   i   ,P   i   ,E   i
],old_users_with_recent_events=[])
                  algorithm_2(similar_profiles[]
,actioned items[] by similar_profiles[])
                  #return     items[]     actioned     by
Most_similar_profile[]

      Module_3(Most_similar_profile[],similar_profiles[],
items_actioned[])
            else if (i in new_users[] or in old_user[]
click or search or rank items p):
```

```
                  get i s profile [S i ,C i ,P i ,E i ]
                  Module_1([S   i   ,C   i   ,P   i   ,E   i
],old_users_with_recent_events_who_actioned_p_[])
                  algorithm_2(similar_profiles[]         ,
actioned items[] by similar_profiles_excluding_p[])

      Module_3(Most_similar_profile[],similar_profiles[],
items_actioned[])
      end

#Intermediary
Module_1([S     i     ,C     i     ,P     i     ,E     i
],old_users_with_recent_events=[]):
      Start:
      #  i  is  current  user  and  x  is  old  user  with
current/recent events
      for x in old_users_with_recent_events[]:
      # Find the similarity between i and each old user
with recent events(who have clicked or actioned an item)
using Jaccard Similarity function J
            J (i,x) = (S i ∩ S x ) ⁄ (S i ∪ S x ) + (C i
∩ C x ) ⁄ (C i ∪ C x ) + (P i ∩ P x ) ⁄ (P i ∪ P x ) +
(E i ∩ E x ) ⁄(E i ∪ E x )
            while(J (i,x) >= 0.5):
                  similar_profiles.append(x)
      return similar_profiles[]
      end

#For missing profiles
Module_3(Most_similar_profile[],similar_profiles[],items_
actioned[]):
      Start:
            if (Most_similar_profile[] ==’NULL’):
                  return     items[]     actioned     by
similar_profiles[]
                  recommend items[]
            else:
                  recommend items[]
      end
# Association rule mining
Algorithm_  2(similar_profiles[]  ,actioned  items[]  by
similar_profiles[]):
      Start:
            Determine the most dominant profile category
among  [S,C,P,E]  of  users  in  similar_profiles[]  which
determine  a  product  p  in  actioned  items[]  by
similar_profiles[]
```

```
            return users with the most dominant profile
category (Most_similar_profile[])
      end

  ModuleSMP(){
  Initialize all (tenants) t ∈ T and (houses)h ∈ H to
  free
  while ∃free tenant t who still has a house h to rent{
  h = first house on t's preference list not yet
  recommended-to-t
  if(h-is-free)
  (t, h) (h is recommended to t)
else some pair (t', h) already exists // a house
recommended
if(h-fit-t-to-t')
                        t' becomes free
(t, h) occurs //This house is recommended to the tenant-t
else
  (t', h) remains    }
}
```