Modelling User Preferences with Multi-Instance Genetic Programming

Amelia Zafra, Sebastián Ventura
Departament of Computer Science and Numerical Analysis
azafra@uco.es, sventura@uco.es

Abstract

In this paper we introduce a novel model for providing users with recommendations about web index pages of their interests. The approach proposed developes user profiles based on evolutionary multi-instance learning which determines what users find interesting and uninteresting by means of rules which add comprehensibility and clarity to user models and increase the quality of the recommendations. Experimental results show that our methodology achieves competitive results, providing high-quality user models which improve the accuracy of recommendations.

Keywords: User Modelling, Recommender Systems, Multi-Instance Learning, Multi-Objective Algorithms, Genetic Programming

1 Introduction

During the last decade, the quantity of potentially interesting products or information services available online has been growing rapidly until it now exceeds human processing capabilities [12]. Moreover, there are many information search situations where the users would like to choose among a set of alternative items or services, but do not have sufficient knowledge, capabilities or time to make such decisions. As such, there is a pressing need for intelligent systems that advise users taking into account their personal needs and interests. Such systems, referred to the literature as recommendation systems [13], can deliver tailored service in the most appropriate and valuable way to the users.

The current recommendation systems can be classified by attending to the processes and the sources of information that are used to achieve the recommendations. According to these criterions, we can find three main classes of recommendation systems [3]: *collaborative recommendation systems*, use explicit and implicit preferences from many users to filter and recommend objects to a given user, ignoring the representation of the objects; *content-based recommendation systems*, filter and recommend the items by matching user query terms with the index term used in the representation of the items, ignoring data from other users, and *hybrid recommendation systems*, which combine based content-based and collaborative methods.

We will focus on content-based recommendation systems, which has its roots in information retrieval [2, 16] and information filtering [4]. Its main improvement over the traditional information retrieval approaches comes from the use of user profiles that contain information about users' tastes, preferences and needs. This information is typically referred to in the literature as the User Model (UM) [11, 14].

The quality of the recommendations provided to the user depends largely on the characteristics of the UM, e.g., how accurate it is, what amount of information it stores, and whether

L. Magdalena, M. Ojeda-Aciego, J.L. Verdegay (eds): Proceedings of IPMU'08, pp. 1113–1120
Torremolinos (Málaga), June 22–27, 2008

this information is up to date. For this reason, the construction of accurate profiles is a key task that the system's success will depend on to a large extent. However, the modeling of user preferences is hard work; it is difficult to obtain enough user modeling data to deliver high quality recommendations, mainly at the initial stages of the interaction with the user, when little information about him/her is available.

In this paper, we propose using the MOG3P-MI algorithm [20] to develop user profiles. The approach proposed is content-based, as it discovers rules that explain if a user is interested in the content of a given item or not. Experiments made with benchmarks in a web index recommendation problem show that our approach achieves competitive results and obtains classifiers which contain simple rules that add comprehensibility and simplicity to the knowledge discovery process, obtaining high-quality user models which improve the accuracy of recommendations.

The rest of this paper is organized as follows. In Section 2 we describe the web index recommendation problem. In Section 3 we describe the proposed system and its components. Next, in section 4 the experimental results are presented and analyzed. Finally, we comment briefly on the conclusions, and propose future work in Section 5.

2 Background

Multiple Instance Learning (MIL) introduced by Dietterich et al. [8] appears in problems where knowledge about training examples is incomplete. In this problem, the teacher labels examples that are sets of instances (called bags in multi-instance literature). The teacher does not label whether an individual instance in a bag is positive or negative so the learning algorithm needs to generate a classifier that will correctly classify unseen examples (i.e. bags of instances). This learning framework is receiving growing attention in the machine learning community and since it was introduced, a wide range of tasks have been formulated as multi-instance problems. Among these tasks, we can cite text categorization [1], content-based image retrieval [5], drug activity prediction [8], image annotation [15] and web index page recommendation [22] (the problem we have faced in this paper).

Web Index Pages are pages that provide titles or brief summaries of other pages. These pages contain plentiful information by means of references, leaving the detailed presentation to their linked pages. A example of a web index pages is http://health.yahoo.com.

There are many web index pages on Internet. Some of these pages may contain issues interesting to the web user while some may not. It would be interesting to analyze automatically these pages and to show to the user only the pages which contain issues interesting for them. To do that, it is necessary to identify the users' interests through analyzing the web index pages that the user has browsed and decide on if a new web index page will interest the user or not. This problem, called web index recommendation, is a specific web usage mining task. Its main difficulty is that the user only specifies whether he or she is interested in an web index page, instead of specifying the concrete links that he or she is really interested in.

This problem could be viewed as a multi-instance problem, where the goal is to label unseen web index pages as positive or negative. A positive web index page is such a page that the user is interested in at least one of its linked pages. A negative web index page is such a page that none of its linked pages interested the user. Thus, each web index page could be regarded as a bag while its linked pages could be regarded as the instances in the bag, and each instance could be represented by means of any of the representations used habitually in text categorization[17]. We use a bag of the most frequent terms appearing on the page along with its frencuency.

3 Multi-objective Grammar Guided Genetic Programming

MOG3P-MI, a Multi-Objective Grammar Guided Genetic Programming for Multi-

```
<cond_I> = <cmp> | "OR" <cmp> <cond_I> | "AND" <cmp> <cond_I>
<cmp> = <op> <term_name>
<op> = "Contains" | "NotContains"
<term_name> = Any valid term name
```

(a) Boolean representation of pages

```
<cond_I> = <cmp> | "OR" <cmp> <cond_I> | "AND" <cmp> <cond_I>
<cmp> = <op> <term_name> <freq_value>
<op> = "<" | ">="
<term_name> = Any valid term name
<term_freq> = Any valid freq value for term
```

(b) Numeric representation of pages

Figure 1: Grammars used in the web index recommendation problem

Instance Learning algorithm has been presented as an improvement of the G3P-MI algorithm [21] that introduces multi-objective strategies to optimize several conflicting learner quality measures at the same time. The algorithm allow us to obtain a set of optimal solutions called non-dominated solutions, that represent a trade-off between the different measurements considered, where no one can be considered to be better than any other with respect to all objective functions. Then, we could introduce preference information to select the solution which offers the best classification guarantee with respect to new data sets.

In this section we specify different aspects which have been taken into account in the design of the MOG3P-MI algorithm.

3.1 Individual Representation

In the MOG3P-MI, as in G3P-MI, individuals represent rules that determine if a bag should be considered positive (that is, is an instance of the concept we want to represent) or negative (if it is not).

if $cond_B(bag)$ **then**
  bag is an instance of the concept;
else
  bag is not an instance of the concept;
end

where $cond_B$ is a condition that is applied over the bag. Considering the multi-instance perspective, $cond_B$ can be expressed as:

$$cond_B(bag) = \bigvee_{\forall instance \in bag} cond_I(instance)$$

Where $\vee$ is the disjunction operator, and $cond_I$ is a condition that is applied over every instance contained in a given bag[1].

Given that the only variable part in the last expressions is the condition that is applied to instances (that is, $cond_I$), the individuals genotype represents this part, while phenotype represents the whole rule that is applied over the bags.

Figure 1 shows the two grammars used to represent individual genotypes for the web index recommendation problem. The first one is applied when we use a boolean representation for web pages, and generate expressions that inform about the presence/absence of a term in the web pages (instances). The second grammar is applied in the case of the web pages representation uses the term frequency, and informs about if a term is present with a frequency more or less than a value.

3.2 Genetic Operators

The elements of the following population are generated by means of two operators: selective mutation and selective crossover[7].

3.2.1 Mutation

The selective mutation operator randomly selects a node in the tree and the grammar is used to derive a new subtree which replaces the subtree in this node. If the new offspring is too large, it will be eliminated to avoid having invalid individuals.

3.2.2 Crossover

The selective crossover is performed by swapping the sub-trees of two parents which have

[1]This expression is equivalent to the one used to define the concept of multi-instance rule coverage in the RIPPER-MI algorithm [6].

the same root symbol according to the defined grammar. If either of the two offspring is too large, they will be replaced by one of their parents.

3.3 Fitness

The problem of developing good metrics to measure the effectiveness of recommendations has been extensively addressed in recommender systems literature [9, 10, 19]. We will use two very commmon measures, sensitivity and specificity. Sensitivity is the proportion of cases correctly identified as meeting a certain condition

$$sensitivity = \frac{tp}{tp + fp} \tag{1}$$

and specificity is the proportion of cases correctly identified as not meeting a certain condition,

$$specificity = \frac{tn}{tn + fn} \tag{2}$$

where *true positive (tp)* represents the cases where the rule predicts that the bag has a given class and the bag does have that class. *True negative (tn)* are cases where the rule predicts that the bag does not have a given class, and indeed the bag does not have it. *False negative (fn)* represents cases where the rule predicts that the bag does not have a given class but the bag does have it. *False positive (fp)* represents cases where the rule predicts that the bag has a given class but the bag does not have it.

The evaluation involves a simultaneous optimization of these two conflicting objectives where a value of 1 in both measurements represents perfect classification. Normally, any increase in sensitivity will be accompanied by a decrease in specificity. Thus, there is no single optimal solution, and the interaction among different objectives gives rise to a set of compromised solutions, largely known as Pareto-optimal solutions. Since none of these Pareto-optimal solutions can be identified as being better than any others without further consideration, our goal is to find as many Pareto-optimal solutions as possible and include preference information to choose one of them as the final classifier. With this aim, we use a multiobjective algorithm which is specified in the next section.

3.4 Evolutionary Algorithm

The main steps of our algorithm are based on the well-known SPEA2. This algorithm was designed by Zitzler, Laumanns and Thiele[23]. It is a Pareto Front based multi-objective evolutionary algorithm that introduces some interesting concepts, such as an external elitist set of non-dominated solutions, a fitness assignment schema which takes into account how many individuals each individual dominates and is dominated by, a nearest neighbour density estimation technique and a truncation method that guarantees the preservation of boundary solutions. The general outline of our algorithm is the following:

BEGIN
Generate random initial population of rules, P_0
and empty archive (external set) A_0.
Set t = 0.
DO
Calculate fitness values of individuals in
P_t and A_t.
A_{t+1} = nondominated individuals in
P_t and A_t.
IF (size of A_{t+1} > N)
Reduce A_{t+1}.
ELSE IF (size of A_{t+1} < N)
Fill A_{t+1} with dominated individuals in
P_t and A_t.
END-IF
Fill mating pool with binary tournament
selection with replacement on A_{t+1}.
Apply recombination and mutation
operators to the mating pool and
set P_{t+1} to the resulting population.
Set t = t + 1
UNTIL an acceptable classification rule is
found or the specified maximum number of
generations has been reached.
END

4 Experiments and Results

To evaluate the suitability of MOG3P-MI in solving the web index recommendation problem, we have compare its results with G3P-MI [21] (a previous version of the algorithm with uniobjective fitness) and with the results reported by Zhou et al. [22] that analysed several variants of the kNN algorithm over these data sets. This section introduces employed data sets, explains some configuration aspects of the algorithms tested and analyzes the results obtained.

4.1 Dataset and Running Parameters

Experiments have been done in nine data sets, in each one of which one different volunteer labelled 113 web index pages according to his/her interests. For each data set, 75 web index pages are randomly selected as training bags while the remaining 38 index pages are used as test bags. We follow exactly the same setup as [22].

These data sets can be categorized into three categories. The first one comprises datasets 1 to 3, and corresponds to users that ignore a high percentage of pages (*selective users*); the second category (datasets 4 to 6) contains users that accept a high percentage of received pages (*permissive users*). Finally, the third category, which we have called *balanced users*, is made up of users who accept and reject pages to the same degree. Table 1 shows a description of data sets evaluated. Given that a recommendation system must manage all kind of users, a good profile learner should be able to generate reliable user models regardless of the type of information available. Due to this, in the next section we will examine the results obtained with each considered category.

Table 1: Experimental data sets

	Training		Test	
Dataset	Pos	Neg	Pos	Neg
1	17	58	4	34
2	18	57	3	35
3	14	61	7	31
4	56	19	33	5
5	62	13	27	11
6	60	15	29	9
7	39	36	16	22
8	35	40	20	18
9	37	38	18	20

Table 2: Global Experimental Results.

	Acc	Se	Sp
Fretcit-kNN	0.8103	0.7007	0.7803
Txt-KNN	0.7233	0.7380	0.4847
Citation-KNN	0.7577	0.6073	0.7407
G3P-MI	0.7810	0.7723	0.7297
MOG3P-MI	0.8480	0.7793	0.7567
Fretcit-kNN[1]	0.8043	0.7117	0.7420
Citation-KNN[1]	0.7357	0.7020	0.5283
Txt-KNN[1]	0.7630	0.6130	0.7207
G3P-MI[1]	0.7313	0.9403	0.4013
MOG3P-MI[1]	0.8420	0.8727	0.7037

[1] Using frecuency of words

Both MOG3P-MI and G3P-MI algorithms have been implemented in the JCLEC framework [18]. The parameters used in all GP runs were: population size: 1000, generations: 100, crossover probability: 95%, mutation probability: 15%, selection method for both parents: tournament selection) and maximum tree depth 15. All experiments are repeated five times with different seeds, and average values were used in report performed in the next section.

4.2 Experimental Results

Table 2 shows results obtained over all available datasets. This table is splitted in two section. The first one corresponds to the results obtained with a boolean page representation (see Section 3.1 and Figure 1a) while the lower section corresponds to a numerical, frequency-based, representation of pages (see Figure 1b).

As we can see, MOG3P-MI achieves the most accurate, selective and specific results, obtaining the most accurate user models both

Table 3: Summary results

	Selective users			Permisive users			Balanced users		
Algorithm	Acc	Se	Sp	Acc	Se	Sp	Acc	Se	Sp
Txt-KNN	0.795	0.636	0.822	0.805	0.863	0.194	0.570	0.715	0.438
Citation-KNN	0.803	0.397	0.868	0.796	0.863	0.577	0.674	0.562	0.777
Fretcit-kNN	0.879	0.579	0.919	0.854	0.924	0.634	0.698	0.599	0.788
G3P-MI	0.807	0.690	0.919	0.825	0.877	0.628	0.711	0.750	0.642
MOG3P-MI	0.904	0.579	0.950	0.868	0.975	0.557	0.772	0.784	0.763
Txt-KNN[1]	0.795	0.519	0.843	0.812	0.851	0.264	0.600	0.736	0.478
Citation-KNN[1]	0.833	0.402	0.907	0.782	0.851	0.498	0.674	0.586	0.757
Fretcit-kNN[1]	0.870	0.615	0.904	0.811	0.916	0.470	0.732	0.604	0.852
G3P-MI[1]	0.845	0.821	0.904	0.823	1.000	0.201	0.526	1.000	0.099
MOG3P-MI[1]	0.895	0.774	0.919	0.860	1.000	0.466	0.771	0.844	0.726

[1] Using frecuency of words

with boolean and numerical representations. G3P-MI algorithm gets worse results with a boolean representation. In the case of a numeric representation although it obtains slightly higher sensitivity values, gives up too much the specificity values. This means that its models do not identify correctly what does not interests users and therefore they are not so dependable. With respect to the rest of techniques (kNN variants), all of them show worse results in all metrics studied. Therefore, we conclude that our algorithm is more reliable (that is, it achieves better balanced results both user interests and does not interest) getting in all cases the best results in global accuracy.

With regard to the study over different kinds of data sets, Table 3 shows the results grouped by the different type of users. As can be seen in the first column, MOG3P-MI gets competitive results in the case of selective users, with very accurate and specific profiles (better accuracy and specificity values) without an important losing of sensitivity values. This result is specially important, because in this case there is not enough information about the interests of users and learning the correct profile is a specially difficult task. The second column shows the results in the case of permissive users. As can be seen, MOG3P-MI obtains better results than other techniques with respect to the sensitivity measure and similar results for the specificity measure. This case, working in the MIL framework, have a greater difficulty because, although we have enough information about the interests of the user, we do not know which specific links are of interest; we only know that the page contains at least one link that interests to the user. Even so, our new algorithm obtains competitive results, improving the accuracy obtained with respect to the other algorithms. Finally, the last column shows the results for balanced users. In this case, our algorithm remains reliable, providing the best results in both specificity and sensitivity and predicting everyone's tastes very well.

Another advantage of our system is the ability to generate comprehensive rules that are easy to understand and provide user profiles with representative information about the user's interest. This comprehensibility of rules is greater when we use a boolean representation than we use a numerical representation, because the use of term frequencies is less friendly than a list of user preferences. This fact can be shown by means of the following examples of rules obtained with our system using both representations:

Firstly, we show a rule obtained for the first user/dataset using boolean representation.

IF (*(no_contain* ***financial****)* ∨
(contain ***violence*** ∧ *no_contain* ***science****)* ∨
(no_contain ***services*** ∧ *no_contain* ***web****)*)
THEN *Recommend page to V1 user.*
ELSE *No recommend page to V1 user.*

We can learn by mean of this rule what topics can be recommended to the user. Thus, user 1 is interested in such topics as violence and is not interested in financial or services or web.

Secondly, we show a rule obtained for the first user/datset using numerical representacion.

IF (*(****frech*** $>$ ***16****)* ∨ *(****house*** $>$ *11)* ∨
*(****science*** $>$ *2* ∧ ***edt*** $>$ *20)* ∨
*(****aol*** $>$ *7)* ∨ *(* ***online*** $>$ *6)*)
THEN *Recommend page to V1 user.*
ELSE *No recommend page to V1 user.*

We can see that this rule is more complex because the words are limited by their frequency and it is more difficult to identify the user preferences. For this, although both representations obtain similar results, after this study we can conclude that numerical representation are less interesting because they obtain less comprehensive rules.

5 Conclusions and Future Work

This paper describes the use of MOG3P-MI for the generation of content-based user profiles, and compares its results with other techniques applied over a Web Index Recommendation problem. As have been proved, MOG3P-MI obtains significantly better results than other techniques in terms of accuracy, sensitivity and specificity and generates interpretable hypotheses with few terms. Also, this representation allows us to export easily acquired knwoledge to new examples.

Although the results are interesting, there are still quite a few considerations that will surely increase the model results. Thus, it would be interesting to employ feature selection techniques that allow us to reduce the number of attributes considered. Our proposal has problems with search space that are too large, a reduction in space would enhance the results. Another interesting aspect is the choice of a concrete solution to be selected from the Pareto optimal set. This set of solutions can not determinate if one is better than another without some information about specific preferences. Thus, we are studying various measures that identify, within the set of user models obtained, which of them can be expected to be better at identifying new topics of interest for the user.

Acknowledgements

This work has been subsidised in part by the TIN2005-08386-C05-02 project of the Spanish Inter-Ministerial Commission of Science and Technology (CICYT) and FEDER funds.

References

[1] S. Andrews, T. Hofmann, and I. Tsochantaridis. Multiple instance learning with generalized support vector machines. In *18th National Conference on Artificial Intelligence (AAAI-02),*, pages 943–944, Edmonton, Alta., 2002.

[2] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley, 1999.

[3] M. Balabanovic and Y. Shoham. Fab: Content-based, collaborative recommendation. *Commum. ACM*, 40(3):66–72, 1997.

[4] N. Belkin and B. Croft. Information filtering and information retrieval. *Commum. ACM*, 35(12):29–37, 1992.

[5] Y. Chen, J. Bi, and J. Wang. Miles: Multiple-instance learning via embedded instance selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(12):1931–1947, 2006.

[6] Y. Chevaleyre and J.-D. Zucker. A framework for learning rules from multiple instance data. In L. de Raedt and P. Flach, editors, *ECML 2001*, volume 2167 of *Lecture Notes in Artificial Intelligence*, pages 49–60, 2001.

[7] J. Couchet, D. Manrique, J. Ríos, and A. Rodríguez-Patón. Crossover and mutation operators for grammar-guided genetic programming. *Soft Computing*, 11(10):943–955, 2007.

[8] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Perez. Solving the multiple instance problem with axis-parallel rectangles. *Artifical Intelligence*, 89(1-2):31–71, 1997.

[9] J. Herlocker, J. Konstan, A. Borchers, and J. Riedl. An algorithmic framework for performing collaborative filtering. In *SIGIR'99: Proceedings of the 22nd Annual International ACM Conference on Research and Development in Information Retrieval*, pages 230–237, Berkeley, California, United States, 1999.

[10] J. Herlocker, J. Konstan, L. Terveen, and J. Riedl. Evaluating collaborative filtering recommender systems. *ACM Transaction Information Systems*, 22(1):5–53, 2004.

[11] A. Kobsa. Generic user modeling systems. user model. *User-Adapt. Interact.*, 11(1–2):49–63, 2001.

[12] P. Maes. Agents that reduce work and information overload. *Commumications of the ACM*, 37(7):31–40, 1994.

[13] A. B. M.D. Mulvenna, S.S. Anand. Personalization on the net using web mining. *Commum. ACM*, 43(8):123–125, 2000.

[14] M. Montaner, B. Lopez, and J. de la Rosa. A taxonomy of recommender agents on the internet. *Artificial Intelligence Review*, 19(4):285–330, 2003.

[15] X. Qi and Y. Han. Incorporating multiple svms for automatic image annotation. *Pattern Recogn.*, 40(2):728–741, 2007.

[16] G. Salton. *Automatic Text Processing*. Addison-Wesley, 1989.

[17] F. Sebastiani. Machine learning in automated text categorization. *ACM Comput. Surv*, 34(1):1–47, 2002.

[18] S. Ventura, C. Romero, A. Zafra, J. A. Delgado, and C. Hervás. JCLEC: A java framework for evolutionary computation soft computing. *Soft Computing*, 12(4):381–392., 2007.

[19] Y. Yang and B. Padmanabhan. Evaluation of online personalization systems: A survey of evaluation schemes and a knowledge-based approach. *Journal of Electronic Commerce Research*, 6(2):112–122., 2005.

[20] A. Zafra and S. Ventura. Multi-objective genetic programming for multiple instance learning. In *EMCL'07: Proceedings of the 18th European Conference on Machine Learning*, LNAI 4701, pages 790–797, Warsaw, Poland, 2007.

[21] A. Zafra, S. Ventura, E. Herrera-Viedma, and C. Romero. Multiple instance learning with genetic programming for web mining. In *IWANN'07: Proceedings of the 9th International Work-Conference on Artificial Neural Networks*, LNCS 4507, pages 919–927, San Sebastian, Spain, 2007.

[22] Z.-H. Zhou, K. Jiang, and M. Li. Multi-instance learning based web mining. *Applied Intelligence*, 22(2):135–147, 2005.

[23] E. Zitzler, M. Laumanns, and L. Thiele. SPEA2: Improving the Strength Pareto Evolutionary Algorithm. Technical Report 103, Gloriastrasse 35, CH-8092 Zurich, Switzerland, 2001.