UDC 004.9

O. V. GERMAN[1]*, J. O. GERMAN*[1]*, S. NASR*[1]

A SELECTION MECHANISM USING MULTI-CRITERIA EVALUATION AND HIERARHICAL CLASSIFYING TREE FOR RESUME DATA PROCESSING

[1] *Belarussian State University of Indormatics and Radioelectronics, Minsk, Republic of Belarus*

The paper considers a problem of optimal feature selection for resume data processing by means of combining multi-criteria evaluation technique and hierarhical classifying trees technology what makes it possible to build a selection mechanism without necessity to collect data for the learning purposes of real applicants. Instead, the learning data are generated by means of the technique used in a full factorial experiment with quite a restricted number of samples. The suggested approach minimizes the number of the features used in selection the best candidates and does not use the quantitative ratings of candidates replacing them with multi-phases classifying procedure. These peculiarities of the suggested selection mechanism make it more flexible and form a basis for applying it in conditions characterized by vagueness and fuzziness of the applicant data.

Keywords: *multi-criteria decision making, hierarhical classifying tree, feature selection.*

Introduction

Automatic resume data processing is one of the important applications of the data mining and text mining technologies [1]. There are some world-wide known resume processing systems [2, 3]. However, they, as a rule, are restricted with a strict curiculum vitae (CV) format presentation, a fixed system of criteria priorities used to select the best candidates. In order to rise up the system flexibility, the system should allow to adapt to specifics of specialty, that is, to change criteria and their priorities accordingly to practical needs. It is also important to minimize the number of criteria in order to reduce personal information database sizes. From this point of view, the paper suggests a technique combining multicriteria decision making (MDM) [4] and hierarchical classification tree (HCT) mechanism [5] in a way, excluding the necessity to collect data for the purpose of HCT learning, and use MDM instead. It gives a formal approach realizing mathematical model of criteria evaluation and generation of the classification tree(s) on the basis of optimal feature set. The paper develops the ideas of the authors' work [6].

Problem formalization

Let the initial feature set include the following attributes: age (F_1), education (F_2), professional expierence (F_3), knowing foreign languages (F_4), participation in big projects (F_5), publications in scientific journals (F_6), participation in scientific conferences (F_7), marital status (F_8), work in other organizations (F_9). The first step to be performed is to find the integral evaluation function I in the form

$$I = \sum_i \alpha_i f_i(F_i)\,, \tag{1}$$

with α_i standing for the feature priorities (normalized non-negative numbers, total sum of which is equal to 1), and $f_i(F_i)$ representing utility functions. To define analytical form of I, one can use the T. Saati's method of hierarhies [7], the Relief procedure [8], or the other techniques used in MDM, so the details are omitted here. Now, suppose I is of the form

$$\begin{aligned} I &= 0.06f(F_1) + 0.25f(F_2) + 0.35f(F_3) + \\ &+ 0.06f(F_4) + 0.1f(F_5) + 0.1f(F_6) + \\ &+ 0.1f(F_7) + 0.04f(F_8) + 0.04f(F_9)\,, \end{aligned} \tag{2}$$

Starting from (2), one should define the hierarchical classification tree to use as a selecting means in CV processing [9]. The inputs of HCT represent the ordered sets of attributes of candidates to vacant position (further we use term Data Set (DS) for short). The HCT filters DS to two categories: (Acc_1) – accepted, (Dec_1) – declined. Clearly, the number of persons from Acc_1 may be greater than 1. In that last case, another HCT should be used to perform a more rigid selection. Again, if Acc_2 is greater than 1, the next filtering is performed accordingly to the scheme outlined later on. This iterative procedure may be finally resolved with random selection from Acc_n (n>1).

Our nearest goal is to show, how to minimize the feature set sizes and build it for selecting Acc_i.

Clearly, formula (1) may contain extra features which should be deleted. To clear which features are excessive and get a non-linear (in general) evaluating function I, one should resolve two different mathematical problems. In practice, instead of defining a non-linear function I, one build an HCT, which performs «hidden computations» replacing direct evaluation of I.

Reduction of the feature set

Introduce some basic ideas of [6] and consider table 1 with some data samples from DS (explanation is given later on).

One uses formula (2) to compute integral evaluation function I. In order to get values in columns $f_1,\ldots,f_9$, one may apply the technique of complete factorial experiments. According to this technique, each feature (utility function) $f_i(F_i)$ takes only two possible values: one is in 15 % distance from the minimum value (that is, from «0» with respect to utility function), and the other one is in the same distance from the maximum value (i.e. from «1»). Let all data objects be divided into two classes A and B, for instance, each sample in class A has value of I greater than 0.5 and, on the contrary, each sample from class B has value of I not exceeding 0.5.

Definition 1. Feature F_t discriminates between two samples $x \in A$ and $y \in B$ if and only if $F_{xt} \neq F_{yt}$.

Reformulation of this definition gives

Definition 2. Feature F_t discriminates between two samples $x \in A$ and $y \in B$ if and only if $f_t(F_{xt}) \neq f_t(F_{yt})$.

Definition 3. A set π of features F_i is discriminating with respect to a given data set DS if for each two data objects d_i and d_j from DS belonging to different classes, there is some feature $F_p \in \pi$ discriminating between d_i and d_j.

Definition 4. A set π is a minimum-size discriminating set for a given data set DS provided that it contains minimum number of features among all discriminating sets.

Lemma. With respect to integral evaluation function I from (1) and a given DS, two minimum-size discriminating sets $\pi(F)$, containing features F_1, F_2, …, F_Z, and $\pi(f)$, containing utility functions $f_k(F_k)$, k = 1, z have the same sizes, i.e. $z = Z$ and are in one-to-one correspondence to each other.

Proof. Let for simplicity there are only two different classes A and B. Let F_t discriminates between two samples d_r and d_s, but f_t – not, that is $F_t(d_r) \neq F_t(d_s)$ but $f_t(F_t(d_r)) = f_t(F_t(d_s))$. Clearly, there must be another feature F_p discriminating d_r and d_s and belonging to $\pi(F)$. Indeed, if no other features from $\pi(F)$ discriminate between d_r and d_s then they are pairwise equal to each other and therefore $I(d_r) = I(d_s)$ with respect to (1), what leads to the fact that d_r and d_s belong to the same class which is impossible. From this, there should be at least one such feature $F_q \in \pi(F)$ with $F_q(d_r) \neq F_q(d_s)$ and $f_q(F_q(d_r)) \neq f_q(F_q(d_s))$. One can include then f_q in $\pi(f)$ and exclude f_t. These considerations remain valid with respect to every two pairs of data objects d_r and d_s from different classes in DS and show the way to make two sets $\pi(F)$ and $\pi(f)$ which are in one-to-one correspondence to each other. This ends the proof.

Finding minimum-size discriminating set

The next step is to build the discriminating 0,1-matrice M, coresponding to full table 1 with elements m_{kij} = 1 if and only if feature f_k discriminates between samples i and j; otherwise m_{kij} = 0 (see Figure 1). The rows correspond to the features (utility functions), the columns are represented by pairs (i, j) with i and j specifying rows in table 1. For instance, consider row f_2 and column

Table 1. **Fragment of DS relating to the example**

	$f_1(F_1)$	$f_2(F_2)$	$f_3(F_3)$	$f_4(F_4)$	$f_5(F_5)$	$f_6(F_6)$	$f_7(F_7)$	$f_8(F_8)$	$f_9(F_9)$	I
d_1	0.15	0.85	0.85	0.85	0.15	0.15	0.15	0.15	0.15	0.627
d_2	0.85	0.85	0.15	0.85	0.15	0.15	0.15	0.15	0.15	0.424
d_3	0.85	0.85	0.85	0.85	0.15	0.15	0.15	0.85	0.15	0.697
d_4	0.85	0.15	0.85	0.15	0.85	0.85	0.15	0.85	0.15	0.62

1, 2 with «0» at the intersection. This means that feature f_2 does not discriminate between data objects d_1 and d_2.

	1, 2	2, 3	2, 4
f_1	1	0	0
f_2	0	0	1
f_3	1	1	1
f_4	0	0	1
f_5	0	0	1
f_6	0	0	1
f_7	0	0	0
f_8	0	1	1
f_9	0	0	0

Fig. 1. Discriminating 0,1-matrice for table 1

Evidently, there are no pairs corresponding to the objects from the same class; also, there should not be columns without «0». The case when no one feature discriminates between some pair of objects from different classes we do not consider (this supposes insufficiency of the criteria used in the model). Our task has reduced to finding a minimum-size cover for M.

Definition 5. One says that row k covers column (i, j) in 0,1-matrice M if and only if $m_{kij} = 1$.

Definition 6. A minimum-size covering set $\pi_{\min}(f)$ for M consists of the minimum possible number of features f_i such that each column of M is covered at least by one row from $\pi_{\min}(f)$.

The problem of finding a minimum-size feature set $\pi_{\min}(f)$ can be resolved as explained in [6]. The technique applied in [6] uses group resolution principle (grp) resembling logical resolution-based inference with more than 2 parent formulas participating in producing logical resolvent (see the details in [10]).

Return to table 1. Its form, participating in complete factorial experiment, is defined for DS with 2^9 =512 data objects. Theoretically, this table produces a discriminating matrice M with 9 rows and $256^2/2$ = 32 768 columns. However, only 512 columns remain unique with the rest 32 256 columns repeting some others. So, the maximum sizes of M are restricted by 9 rows and 512 columns for the case under consideration. This columns quantity can be theoretically obtained for approximately 33 different data objects). Evidently, such matrice M can be easily generated programmatically. The problem consists in finding a minimum-size cover of M what may be efficiently realized with the help of grp or other existing technique [11]. Then, given the features from $\pi_{\min}(f)$, it is possible to build a classification tree for instance with the help of Python analytical means.

Experimental results

The experiments showed that it is necessary to take into consideration all 9 features to build a classifying tree. However, some combinations of features may be excluded, as the candidates with such profiles get very low resulting I-estimations (e. g., 0.3 or lower). This led to reduction of the feature set to 7 features constituting the minimum-size cover set $\pi_{\min}(f)$ for discriminating matrice (*DM*) in order to correctly classify the persons to Acc_1 and Dec_1 by means of the HCT. They are: Age, Education, practical Experience, Knowing foreign Languages, Publications, Marital Status, and Work in other Organizations. It is so-called the HCT_1 of the first level as it uses only two classes A and B where class A is represented by the persons with the integral evaluation function I values greater than 0.5, and B comprising the rest candidates. The classification mechanism, used in HCT, appeals to differentiating objects by comparison of their features (not by computation of some integral evaluation criterion like I). In general, HCT may realize some kind of a complex non-linear estimation. The problem may arise, what to do when there remain more than the required number of candidates qualified as accepted. To decrease the number of candidates remained after the first selection, one can use the second classification tree HCT_2 which is created by analogy with HCT_1. However, in the case of HCT_2 one should define a higher boundary level of the integral evaluation function I, separating class A from class B. For example, if $I \geq 0.6$ then the candidate is qualified as accepted, otherwise, as declined. The corresponding changes should be made in *DS* used in factorial experiments to build HCT_2. Our program resulted in 8 features now, excluding F_5 – participating in big projects. This process should be continued to build HCT_3 (for $I \geq 0.7$), HCT_4 (for $I \geq 0.8$), HCT_5 (for $I \geq 0.9$), HCT_6 (for $I \geq 0.95$). In our case, HCT3 uses 5 features: F_2, F_3, F_5, F_7, F_8, while HCT_4 and the rest ones have only 3 features: F_2, F_4, F_6. So, a collection of the classification trees has been created to provide sequential reduction (if necessary) the number of presumably accepted candidates. If, despite the filtering, there would remain more than one candidate then the final selection is realized as a random sampling.

Conclusion

The main advantage of the outlined technique consists in saving memory expencies for there is no need to store a database with feature values. Instead, a collection of hierarhical classifying trees is used with reduced feature set(s). Each HCT processes a vector of normalized feature values (in diapason [0..1]). To build HCT, one uses a factorial experiment resulting in building the discrimination matrix which is used to find a minimum-size covering set containing an opimal feature collection. As a final step, one applies a Python procedure to build a classifying tree. Varying a boundary level of I between the sets Acc_i and Dec_i, one provides a collection of HCT to filter the accepted candidates as much as possible. The number of the features in the sequence of HCTs decreases for high levels of I. If at the end of the filtering process there remains more than one candidate, the random selection is performed. The experts are in position to test different models, represented by equation (1) in order to find the feature weights most relevant to their preferences.

REFERENCES

1. **Sneha Kumari, Punam Giri.** Automated Resume Extraction and candidate Selection System. International Journal of Research in Engineering and Technology (IJRET). 2014., vol. 3, issue 1, pp. 206–208.
2. Taleo. Applicant tracking system. [Electronic resource].– Access mode: https://www.applicanttrackingsystems.net/oracle-taleo/. Access date: 11.02.2021.
3. GreenHouse ATS: what job-seekers need to know. [Electronic resource].–Access mode: https://www.jobscan.co/blog/greenhouse-ats-what-job-seekers-need-to-know/.– Access date: 11.02.2021.
4. **Tzeng G.-H., Huang J.J.** Multipple Attribute Decision Making: Methods and Applications. Chapman and Hall. 2011. vol.166., 349 p.
5. **Sudrajat R., Irianingsih I., Krisnawan D.** Analysis of data mining classification by comparison of C4.5 and ID algorithms. IOP Conference Series: Materials and Engineering. 2017. vol. 166. pp. 12–31.
6. **German, O.V., Nasr S.** New method for optimal feature set reduction. Informatics and automation. SPIRAS Proceedings (St. Petersburg, Russia). 2020. vol. 19, № 6. pp. 1198–1221.
7. **Saaty T.L., Vargas L.G.** Decision making with the analytic Network Process. Springer. 2013.– 370 p.
8. **Urbanovicz R.J., Meeker M., Cava V.L. et al.** Relief-based feature selection: introduction and review. Journal of biomedical informatics. 2018., vol. 85, pp. 189–203.
9. **Vens C., Stryif J., Shietgat L. et al.** Decision trees for hierarchical multi-label classification. Machine Learning. 2008. vol. 73., № 2. pp. 185–214.
10. **German J.O.** One version of the group resolution principle for discrete optimization. Proc. of Intern. conf. Information Technologies and Systems (ITS) 2020. Minsk, BSUIR, 2020, pp. 165–167.
11. **Capraro A., Fischetti M.** A heuristic method for the set covering problem. Operations Research. 2000. vol. 47., № 5.

ЛИТЕРАТУРА

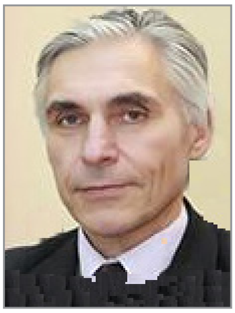
1. **Sneha, K.** Automated Resume Extraction and candidate Selection System/K. Sneha, P. Giri // International Journal of Research in Engineering and Technology (IJRET). 2014., vol. 3, issue 1, pp. 206–208.
2. Taleo. Applicant tracking system. [Electronic resource].– Access mode: https://www.applicanttrackingsystems.net/oracle-taleo/. Access date: 11.02.2021.
3. GreenHouse ATS: what job-seekers need to know. [Electronic resource].–Access mode: https://www.jobscan.co/blog/greenhouse-ats-what-job-seekers-need-to-know/.– Access date: 11.02.2021.
4. **Tzeng, G.-H.** Multipple Attribute Decision Making: Methods and Applications / G.-H. Tzeng, J.J. Huang // Chapman and Hall. 2011. vol. 166., 349 p.
5. **Sudrajat, R.** Analysis of data mining classification by comparison of C4.5 and ID algorithms / R. Sudrajat, I. Irianingsih, D. Krisnawan // IOP Conference Series: Materials and Engineering. 2017. vol. 166. pp. 12–31.
6. **German, O.V.** New method for optimal feature set reduction / O.V. German, S. Nasr // Informatics and automation. SPIRAS Proceedings (St. Petersburg, Russia). 2020. vol. 19, № 6. pp. 1198–1221.
7. **Saaty T.L.** Decision making with the analytic Network Process / T.L. Saaty, L.G. Vargas // Springer. 2013.– 370 p.
8. **Urbanovicz R.** Relief-based feature selection: introduction and review / R.J. Urbanovicz, V.L. Cava et al. // Journal of biomedical informatics. 2018., vol. 85, pp. 189–203.
9. **Vens, C.** Decision trees for hierarchical multi-label classification / C. Vens, J. Stryif, L. Shietgat et al. // Machine Learning. 2008. vol. 73., № 2. pp. 185–214.
10. **German J.O.** One version of the group resolution principle for discrete optimization. Proc. of Intern. conf. Information Technologies and Systems (ITS) 2020. Minsk, BSUIR, 2020, pp. 165–167.
11. **Capraro, A.** A heuristic method for the set covering problem / A. Caparo, M. Fischetti // Operations Research. 2000. vol. 47., № 5.

О. В. ГЕРМАН, Ю. О. ГЕРМАН, С. НАСР

МЕХАНИЗМ ОБРАБОТКИ РЕЗЮМЕ, ИСПОЛЬЗУЮЩИЙ МНОГОКРИТЕРИАЛЬНОЕ ОЦЕНИВАНИЕ И ИЕРАРХИЧЕСКИЕ КЛАССИФИЦИРУЮЩИЕ ДЕРЕВЬЯ

В статье рассматривается задача оптимального выбора атрибутов при отборе кандидатов на основании их резюме в автоматическом режиме. Описываемый подход к решению основан на объединении мультикритериального выбора (оценки), используемого в системах принятия решений, и технологии иерархических классифицирующих деревьев, что позволяет реализовать механизм селекции без необходимости собирать реальные данные кандидатов и выполнять на них обучение системы. Вместо этого данные генерируются на основе техники полнофакторного эксперимента, при этом количество генерируемых вариантов сравнительно невелико для систем машинной обработки. Сгенерированные данные используются для построения последовательности классифицирующих деревьев и определения минимального множества атрибутов заявителей, используемых для итоговой оценки о принятии на работу. Описанный в статье механизм обработки резюме является достаточно гибким и может быть использован также в условиях неполных и нечетких данных заявителей.

Ключевые слова: *многокритериальный выбор решений, иерархические классифицирующие деревья, выбор атрибутов.*



Oleg German got PhD in computer science from Belarussian state university of Informatics and RadioElectronics (BSUIR), Minsk in 1987. Now works as associate professor in BSUIR. Has published 8 monographs, 110 works, 10 patents. Scientific area includes programming, informatics.

Julia German got PhD in computer science from BSUIR, Minsk in 2018. Now works as associate professor in BSUIR. Has published 2 monographs, 40 works. Scientific area includes applied logic, informatics, decision making.

Sara Nasr PhD student of BSUIR, master of science. Has published 5 papers, participated in 2 international scientific conferences. Scientific area includes informatics, decision making.