VOL. 9, NO. 11, NOVEMBER 2014 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences
©2006-2014 Asian Research Publishing Network (ARPN). All rights reserved.

www.arpnjournals.com

WS COMPONENT SELECTION BY IMPROVISED HIGH HIT RATIO USING SIMPLE JACCARD COSINE DISTANCES WITH MODI'S COST EFFECTIVENESS

K. R. Sekar[1], S. Devasena[2], K. S. Ravichandran[1] and J Sethuraman[1]
[1]School of Computing, SASTRA University, Thanjavur, India
[2]M.Tech Advanced Computing, SASTRA University, Thanjavur, India
E-Mail: sekar_kr@cse.sastra.edu

ABSTRACT

Software component is an inevitable commodity in the field of web technology applications. Any business transaction in online has been taken care by the software component as the whole. Web service is a software component, which is all articulating highly in the market. Selection and prediction for such a type of component is a tough task for our application. Prism classifier is a statistical tool through which obtaining good classification and ontology for our semantics with number of attributes. Every web service component has its own significance and QOS. Prism classifier generates output considering only high values, resulting in the rules, which contains only the best component, skipping the next components in the priority queue. The drawback in classical prism classifier is rectified by considering the attributes of the component having tie between maximum values. The homogeneity levels amongst a class, variation between the training data sets are also analyzed. By, improving the prism classifier, the resulting rule contains the best of the best component suitable for the customer. The series of tests like Simple Matching co-efficient, Jaccard distances, cosine distance, T-test, ANOVA etc., together with modified Prism classifier is named as IH2RC [Improvised High Hit Ratio Classifier]. In this paper, IH2RC is applied on a training data set, which contains online translators with their related attributes. For cost effectiveness of the software component MODI'S method is employed in this scenario.

Keywords: improvised prism, jaccard distance, cosine distance, T-test, ANOVA, MODI'S method.

INTRODUCTION

With the advent of the World Wide Web, the desktop applications are being ousted by the web services, which include document editing, picture editing, online games etc. Web services have become a good trend in recent days. One such purpose of web service is online translator services. With the support of UTF (UCS Transformation Format) many languages have been added to the computer dictionary and hence people started writing digital content in their own native language. World Wide Web has its wide spread audience, people referring sites with language they are not familiar with, opt to translate since there has been significant improvement in online translators. Web hosting corporate find difficult to choose the right component to deliver it for the customers according to their brief technical and non-technical requirements. Choosing a component with just an employer's knowledge base leads to aftermath bugs, malfunctioning and may lead to architecture mismatch if the user opts for upgrading his website, the web service must also work for the upgraded version. Such corporate seek out for a solution to best match the client's requirement. Data Mining helps immensely to reduce the consulting cost, manual analysis time etc.

Various data mining techniques like clustering, classification, spatial mining, pattern mining, rule generation etc are existing, one from these techniques can be adopted to choose the right component. Rule based classification helps in generating rules based on the training data set. This technique is known as learning algorithms, since, it learns from the available history of data. The selection of elements is based on over-produce-and-choose strategy. Where, a large number of classifiers produce results, and NSGA-II is used along with fuzzy rule based Classifier Ensemble (CE) to select better obliging subset of items at a good accuracy with less complication [1]. Numerous rules are produced and only the rules with greater support are considered in the final rules thus generated. There are more than a few rule based classification like 1-R, fuzzy rule based, PRISM etc. We are employing improved PRISM algorithm in our project, to create IH2RC model, which checks each and every process by variance analysis, similarity measures, ANOVA.

PROPOSED METHODOLOGIES

Selection of web service (component) through mining involves data set with actual real world values for a better prediction. To facilitate correctness in data set, entry level data set is considered by Distribute measures and fastidious dataset is considered by weighted attributes. The uncertainty in supplier selection was identified by fuzzy set theory. It does not include the qualification level, instead it considers final selection. This paper presents integrated fuzzy technique to consider both non-compensatory rule for sorting in qualification stages and a compensatory rule for ranking in the final selection. Fuzzy rule based classifier is used in qualification level selection, and 3 types of defuzzification techniques are used in final level selection [2]. Results have proven lesser variance than with unprocessed data set. The purity level of the components within a range of class, variation betwixt the

©2006-2014 Asian Research Publishing Network (ARPN). All rights reserved.

data sets is also analyzed. The series of tests like Simple Matching co-efficient, Jaccard distances, Jaccard co-efficient are used to find the similarity among the same class, after classifying the raw data set. The drawback in classical prism classifier is rectified by considering the attributes of the component having tie between maximum values. By, improvising the prism classifier, resulting in the rules, this contains the best of the best component suitable for the customer. Recent inclination in electronic industries is miniaturization of components and mounting them on the Printed Circuit boards (PCB). These mounting on the PCBs, include the exact positioning of components with higher precision which classes of components have the characteristic of getting placed in a specific place. Multiple classifier combination is a technique, which considers the results of various classifiers to reduce the variance of errors in the estimation to improve the classification [3]. As various classifiers are considered in the paper to reduce variance, T-test and ANOVA are used to check if the generated rules vary with the mean value to what extent, and checks if the variance is acceptable by comparing with f-distribution Table. These series of improvisation together with modified Prism classifier is known as IH2RC [Improvised High Hit Ratio Classifier].

A) IH2RC architecture diagram

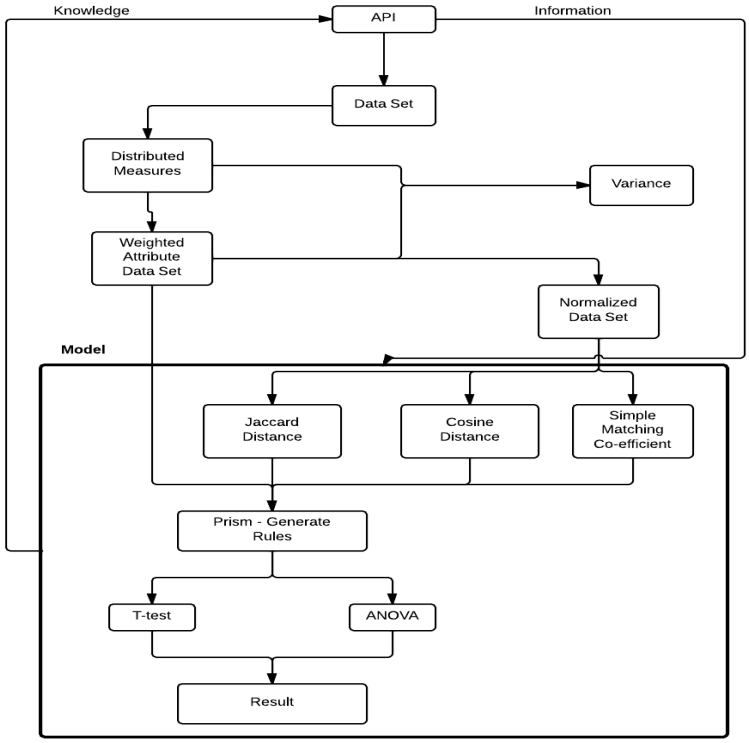

Figure-1. IH2RC (Improved High Hit Ratio Classifier) architecture.

IH2RC is developed to fasten and to create more accurate prediction rate for the web services. It has been a challenge to choose the attributes and the negative impact of some attributes on the result for a prediction. Ih2rc model consists of 2 stages, first stage is to set up the dataset required to learn knowledge (i.e.). IHRC model allows user to decide which language translator best suits for his purpose. Rule based classifier is very much helpful by generating the rules, which servers for the sole purpose of recommending the best language translator service for the user requirements from the API.

Training set (i.e.) data set is collected from Joomla.org, and its attributes are selected after a thorough investigation of reviews and comments put out by the users. The data set thus formed is not yet classified into Excellent, Poor, Average classes. Distributed Measures is applied and the translators are thus classified. Data set is reduced; a large training set produces numerous rules in a Fuzzy rule based classifier. A large training set can also affect the computation time, and thus makes less interpretability with numerous rules generated. This problem can be tackled to some extent by reducing the training set. It shows that some of these methods can considerably help to reduce the computational time of the evolutionary process and to decrease the complexity of the fuzzy rule-based models with a very limited decrease of their accuracy with respect to the models generated by using the overall training set [4]. Data set is casted to various styles based on the requirements of the process, at one stage, the data set is needed to be multiplied with weights, and at other time, all the values must be either 0s

©2006-2014 Asian Research Publishing Network (ARPN). All rights reserved.


or 1s. A variance analysis is conducted if there is a lesser variance in weighted attributes, than distributed measures.

Translators are classified as Excellent, Average and Poor classes. There must be a check on the translators which fall into the same class. Thus various similarity analyses are performed to see the coherence of properties within the same group of translators. Jaccard Distance, Simple Matching co-efficient, Cosine distance are the various similarity analysis performed. Each analysis has its own threshold and scale to see similarity among the translators of same group. Prism, a rule based classifier is used to generate rules for the data set with weights. Weighted data set is considered, since, the result of the attributes must be skewed to only important attribute. There has been enhancement in prism classifier, which proves, improved rule generation. Hence, the user's requirements are matched with the rules generated and the best suited translator is returned back as a result. Statistical evaluation methods are used to check if the result (i.e.) rules produced are coherent and do not vary much with the means. T-test and ANOVA are used to evaluate the means of distributed measures and weighted attributes. They have their own scale to measure the variance. The components can be selected from the rules thus generated by any of the methods like Apriori etc.; Spectrum in physics represents a continuous range of wavelength, till an extent comes in a band of a colour. The same way, in market, stakeholders or managers, are categorized under brand spectrum, while comparing a company with other competitor's. Apriori algorithm is used for association rules; Knowledge from the database is generated as rules, by using ontology-based data mining approach, to suggest to the improvements to the company based on the competitors [5].The selection of elements is based on over-produce-and-choose strategy. Where, a large number of classifiers produce results, and NSGA-II is used along with fuzzy rule based Classifier Ensemble (CE) to select better obliging subset of items at a good accuracy with less complication [1].

B) Dataset collection

To gauge the translators, attributes are to be found, which influences its performance more. Each attribute value is recorded from either organization manual or from user reviews. List of translator services are taken from Joomla.org and various other sites and reviews from the same.

Table-1. Raw dataset.

Translators	Usability	Number of Languages	Human Translation	Accuracy	Detect Source Language	Number of Characters	Website Translation	Document Translation
babelfish	Bad	14	Yes	Good	No	200	No	No
babylon	Good	30	Yes	good	No	1000	No	No
bing translator	Good	44	No	Good	Yes	10000	Yes	No
dictionary.com	Average	52	No	average	No	300	No	No
Google translator	Good	58	No	Good	Yes	5000	Yes	Yes
JV translator	Average	36	No	Good	No	200	No	No
MultiTrans	Good	50	No	Good	Yes	500	No	Yes
NS translator	Good	60	No	Good	No	200	No	No
Promt	Average	7	No	Good	No	3000	No	No
Reverso	Bad	5	No	bad	No	800	No	No
SDLtranslator	Good	43	Yes	average	No	4500	No	Yes
SEF translate	Average	80	No	Good	No	200	No	No
systranet	Good	15	No	good	No	10001	Yes	Yes
worldlingo	Good	32	Yes	average	Yes	3000	Yes	Yes

©2006-2014 Asian Research Publishing Network (ARPN). All rights reserved.

C) Distributed measure

Table-1 represents the raw information collected, which is not categorical yet. To classify translators to their respective ranking (i.e.) classes, we use distributed measures to present the data set to a more informative style. Components are bought from various providers by the company, which is in the middle of supply chain. Based on user's needs, company may need to provide various services, and hence, various heterogeneous components are bought from different sources and integrated. A model driven approach is developed to glue the mismatched components together, by generating automatically glue components to suit for the company's standard [6]. The steps involving Distributed measures calculation:

a) Transforming the data set with alpha numeric values to numerical values, by calculating the % value for each entry with respect to its own attributes. Assigning 100% to the maximum value by X = instance of the attribute/ maximum value of the attribute. There by, calculating number of languages for babelfish, 80 being highest value (SEF Translator) (14/80)*100 = 17.5%. In the same manner calculate the % values for the remaining attributes.
b) Calculate the sum by adding up all the attributes a translator to give a Total. Consider, for Total for babelfish translator = 219.5. On calculating similarly, totals of other translators are 340, 565, 168, 622.5, 197, 467.5, 262, 188.75, 14.25, 448.75, 252, 518.75 and 612.

Now, total% should be calculated by considering the highest value and transforming it to 100, and other transforming other values, respectively. Consider Google translator. Its total is 622.5 and so calculating total%, (622.5/622.5)*100 = 100. Likewise the total %value for other translators are 35.26 %, 54.61%, 90.76%, 26.98%, 100%, 31.64%, 75.10%, 42.08%, 30.32%, 2.28%, 72.08%, 40.48%, 83.33% and 98.31%. And on considering the total% value, Translators are categorized as Poor, Average and Excellent.

Table-2. Distributed measures.

Translators	Usability	Number of languages	Human translation	Accuracy	Detect Source language	Number of characters	Website translation	Document translation	Total	Distributed percentile	Class
Reverso	0	6.25	0	0	0	8	0	0	14.25	2.289157	Poor
dictionary.com	50	65	0	50	0	3	0	0	168	26.98795	Poor
Promt	50	8.75	0	100	0	30	0	0	188.75	30.32129	Poor
JV translator	50	45	0	100	0	2	0	0	197	31.64659	Poor
babelfish	0	17.5	100	100	0	2	0	0	219.5	35.26104	Poor
SEF translate	50	100	0	100	0	2	0	0	252	40.48193	Poor
NS translator	100	60	0	100	0	2	0	0	262	42.08835	Poor
Babylon	100	30	100	100	0	10	0	0	340	54.61847	Average
SDLtranslator	100	53.75	100	50	0	45	0	100	448.75	72.08835	Average
MultiTrans	100	62.5	0	100	100	5	0	100	467.5	75.1004	Average
systranet	100	18.75	0	100	0	100	100	100	518.75	83.33333	Average
bing translator	100	55	100	100	100	10	100	0	565	90.76305	Excellent
worldlingo	100	32	100	50	100	30	100	100	612	98.31325	Excellent
Google translator	100	72.5	0	100	100	50	100	100	622.5	100	Excellent

D) Weighted attributes

Giving equal importance to less influential attribute may affect the total percentage of the translators, which may lead to false interpretation into classes. Hence, giving importance to attributes which has better quality, leads to a total percentage of better quality, and hence the translators are categorized based on the math. Assign weights to each of the attribute, by multiplying the distributed measures values of attributes with weights.

Weights are assigned as follows:

a) Arrange the attributes based on priority of importance.
b) Give the first attribute 1.
c) Calculate the weight of next less important attribute by 1-(1/n)x, where 'n' represents number of attributes, and 'x' represents number of instance.
d) On considering the total percentage value, Translators are categorized as Poor, Average and Excellent.

©2006-2014 Asian Research Publishing Network (ARPN). All rights reserved.


Table-3. Priority of attributes and their weights.

POA	Acc	NC	DSL	NL	Us	WT	DT	HT
Weight	1	0.875	0.75	0.625	0.5	0.375	0.25	0.125

e) **Legend-1:** HT- Human Translation, DT- Document Translation, WT- Website Translation, Us- Usability, NL- Number of Languages, DSL-Detect Source Language, NC- Number of Characters, ACC-Accuracy.

f) Consider, calculating total percentage for the first translator Reverso.

g) Referring to the weights Table, Usability's weight is 0.5, multiply it with distributed measure value of usability for Reverso, viz. 0*0.5=0, likewise multiplying weights for their respective attributes.

h) Reverso=10.90625. Similarly calculating total percentage using weights, the following table is formulated.

i) To inspect the outcomes obtained from the above proposed techniques, the variance approach is used to analyze the improvement.$\sum (xi-\mu)\ 2/n$, the variance of distributed measures is 945.9805 and for Weighted Attributed are 629.8538. Since, variance is less for distributed measure it's a significant proof that considering weighted attribute dataset is a better form of classification.

Table-4. Weighted attributes.

Translators	Usability	Number of languages	Human translation	Accuracy	Detect source language	Number of characters	Website translation	Document translation	Total	Distributed percentile	Class
Reverso	0	3.90625	0	0	0	7	0	0	10.90625	2.8962656	Poor
dictionary.com	25	40.625	0	50	0	2.625	0	0	118.25	31.40249	Poor
Babelfish	0	10.9375	12.5	100	0	1.75	0	0	125.1875	33.244813	Poor
JV Translator	25	28.125	0	100	0	1.75	0	0	154.875	41.128631	Poor
Promt	25	5.46875	0	100	0	26.25	0	0	156.7188	41.618257	Poor
NS Translator	50	37.5	0	100	0	1.75	0	0	189.25	50.257261	Average
SEF Translate	25	62.5	0	100	0	1.75	0	0	189.25	50.257261	Average
Babylon	50	18.75	12.5	100	0	8.75	0	0	190	50.456432	Average
SDLTranslator	50	33.59375	12.5	50	0	39.375	0	25	210.4688	55.892116	Average
MultiTrans	50	39.0625	0	100	75	4.375	0	25	293.4375	77.925311	Average
Worldlingo	50	20	12.5	50	75	26.25	37.5	25	296.25	78.672199	Average
Systranet	50	11.71875	0	100	0	87.5	37.5	25	311.7188	82.780083	Excellent
bing translator	50	34.375	12.5	100	75	8.75	37.5	0	318.125	84.481328	Excellent
Google translator	50	45.3125	0	100	75	43.75	37.5	25	376.5625	100	Excellent

E) Variance

Table-5. Variance between distributed measures vs. weighted attributes.

TN	BF	BL	BT	DC	GT	JVT	MT	NST	PR	RV	SDLT	SEFT	SY	WL
CDM	P	A	E	P	E	P	A	P	P	P	A	P	A	E
CWA	P	A	E	P	E	P	A	A	P	P	A	A	E	A

©2006-2014 Asian Research Publishing Network (ARPN). All rights reserved.

Legend-2: Row wise: TN - Translator Name, BF- Babelfish, BL- Babylon, BT- Bing Translator, DC- Dictionary.com, GT- Google Translator, JVT- JV Translator, MT- Multi Trans, NST-NS Translator, PR- Promt, RV- Reverso, SDLT- SDL Translator, SEFT- SEF Translator, SY- Systranet, WL- Worldingo.

Legend-3: Column wise: CDM- Classified by Distributed Measured, CWA- Classified by Weighted Attributes,

Legend-4: Instance-P-Poor, A- Average, E- Excellent.

F) Similarity analysis

To authenticate the translators which are classified under Excellent have similar characteristics, various similar tests are carried out among entities under same class. The similarity can be applied via cohesion and coupling, this makes some sort of similarity between the components cited in the paper components can be developed [BUILD] or can be bought [BUY] depending on the budget. Cohesion and coupling are the two vital aspects in component selection, how far the components react to each other after deployment. Cohesion is the internal interaction and coupling is the inter-component interaction. Low coupling and high cohesion is essential for software. Cohesion and coupling are measured by Intra-modular coupling density (ICD) [7]. We are pre-processing the dataset to 0s and 1s as follows:

Table-6. Normalized values to 0s and 1s.

Translators	Usability	Number of languages	Human translation	Accuracy	Detect source language	Number of characters	Website translation	Document translation	Class
Babylon	1	0	1	1	0	0	0	0	Average
MultiTrans	1	1	0	1	1	0	0	1	Average
SDLTranslator	1	1	1	0	0	1	0	1	Average
Systranet	1	0	0	1	0	1	1	1	Average
bing translator	1	1	1	1	1	0	1	0	Excellent
Google translator	1	1	0	1	1	1	1	1	Excellent
worldlingo	1	0	1	0	1	1	1	1	Excellent
Babelfish	0	0	1	1	0	0	0	0	Poor
dictionary.com	0	1	0	0	0	0	0	0	Poor
JV translator	0	1	0	1	0	0	0	0	Poor
NS translator	1	1	0	1	0	0	0	0	Poor
Promt	0	0	0	1	0	1	0	0	Poor
Reverso	0	0	0	0	0	0	0	0	Poor
SEF Translate	0	1	0	1	0	0	0	0	Poor

Below are various similarity analyses:

G) Simple matching co-efficient

The Simple Matching Coefficient (SMC) is a statistical process for finding out presence/absence of distributions on a set and simply counts the number of samples which have presence or absence in both distributions. It is calculated within a set of class to see similarities among the members of a class. Data set is pre-processed to fit in for the formula. Column average is taken and the values which are greater than average are considered 1 and lesser than average is considered 0. Clustering reduces the time for searching components, rather than from the whole pool of components. A new technique of hybrid XOR, similar to simple matching co-efficient is used to cluster the components to find the similarity between the components by constructing similarity matrices if the order n by n. The output is highly cohesive groups [8].

Formula

SMC= m11+m00/m11+m10+m01+m00

where, m11- both positive, m10- 1st variable positive and 2nd variable negative, m01 - 1st variable negative and 2nd variable positive, m00- both negative.

Consider excellent class and their respective dataset after pre-processing is,

©2006-2014 Asian Research Publishing Network (ARPN). All rights reserved.


Table-7. Dataset containing translators of only excellent class.

Bing	**1**	**1**	**1**	**1**	**1**	**0**	**1**	**0**	**Excellent**
Google translator	1	1	0	1	1	1	1	1	Excellent
Worldlingo	1	0	1	0	1	1	1	1	Excellent

SMC value for Bing and Google: (5+0)/ (1+2+5+0) = 0.625, similarly calculating SMC between each pair in excellent class, following results were obtained: 0.5625, 0.5833333. The variation between minimum and maximum among Excellent (0.625-0.5625) = 0.0625. This proves more similarity between the members. Similarly calculating for Average and Poor, following observations were made, Average = 0.5, 0.5, 0.5, 0.5, 0.5. The variation between minimum and maximum among Average (0.5-0.5) = 0.0. This proves more similarity between the members.

Poor = 0.625, 0.6875, 0.6666667, 0.6875, 0.7, 0.7083333, 0.73214287, 0.734375, 0.7222222, 0.7375, 0.75, 0.7604167, 0.7596156, 0.7589286, 0.775, 0.765625, 0.75735295, 0.7638889, 0.7631579, 0.7625. The variation between minimum and maximum among Poor (0.775-0.625) = 0.14999998. This proves more similarity between the members, since the value is low on a scale of 0-1.

H) Jaccard distance

This is useful when both positive and negative values don't carry equal information (Asymmetric). Jaccard co-efficient is used only to find the similarity between the entities, while Jaccard distance is (1-JACCARD DISTANCE), which calculates dissimilarity between the entities. The intra class distance must be equal, and the inter class distance must vary since they are from dissimilar class, Jaccard = m11/m11+m10+m01, where, m11- both positive, m10- 1st variable positive and 2nd variable negative, m01- 1st variable negative and 2nd variable positive, Consider the Excellent class again, calculating Jaccard distance between Bing and Google: (5)/ (1+2+5) = 0.375,Similarly calculating Jaccard distance between each pair in excellent class, following results were obtained: 0.4375, 0.41666666, the Variation between minimum and maximum among Excellent (0.4375-0.375) = 0.0625, Similarly calculating for Average and Poor, following observations were made, Average = 0.6666667, 0.6666667, 0.6666667, 0.64, 0.625, 0.61538464. The Variation between minimum and maximum among Average (0.6666667-0.61538464) = 0.05128205. Poor = 1.0, 0.8333333, 0.8, 0.7692308, 0.8, 0.7777778, 0.75, 0.73913044, 0.7692308, 0.7777778, 0.7586207, 0.71875, 0.71428573, 0.7297297, 0.6923077, 0.6976744, 0.7173913, 0.6938776, 0.7058824, 0.7037037, 0.71428573, the Variation between minimum and maximum among Poor (1.0-0.6923077) = 0.3076923. This proves more similarity between the members, since the value is low on a scale of 0-1 for all the 3 classes.

I) Cosine distance

Cosine distance analysis is used to measure the cohesion within the formed class in the field of Data Mining. The Cos 0o is 1, and if the outcome is nearing to 1, there is more similarity between the entities. Cos 90o will have similarity value 0 and so there is no similarity between the entities. Hence if the value is nearing to 1 then there exists more the similarity between entities.

Cos (θ) = x*y/|x|*|y|, x stands for classes in Distributed Measure, y stands for classes in Distributed measures with weight. Considering the dataset with 0s and 1s, Consider the Excellent class again, calculating cosine distance between Bing and Google: 0.22848324. Similarly calculating cosine distance between each pair in excellent class, following results were obtained: 0.2794233, 0.2631579, the Variation between minimum and maximum among Excellent (0.2794233-0.22848324) = 0.05094005. Similarly calculating for Average and Poor, following observations were made, Average = 0.48360223, 0.48360223, 0.48360223, 0.46214712, 0.44940224, 0.440983. The Variation between minimum and maximum among Average (0.48360223-0.440983) = 0.04261923. Poor = 1.0, 0.7113249, 0.6666667, 0.625, 0.6645898, 0.63485163, 0.5996796, 0.5859607, 0.62426543, 0.63619655, 0.6105096, 0.5597956, 0.5545646, 0.57437176, 0.52932125, 0.53571427, 0.5587512, 0.5303318, 0.54356456, 0.5411685, 0.5548681. The Variation between minimum and maximum among Poor (1.0-0.52932125) = 0.47067875. This proves more similarity between the members, since the value is nearing to 0 for all the 3 classes.

J) Improvised prism

A rule based classifier is a technique for classifying records using "if-then" rules, to a given set of class. The "if-then" rules are used to classify and produce rules by learning from the dataset. These rules can be used to predict in future, software components are prone to be defective after their deployment and evolution. Predicting those software components is essential, to improve the software constantly. Machine learning classification models are implemented to learn the defective components. Relational association rules are generated exploring the numerical ordering between the attributes of the data set that occurs frequently [9]. The PRISM algorithm uses a depth-first search to construct the next rule for a given class C. Since the consequent of the rule must be the given class C, only the antecedent needs to be constructed; this is done by starting with an empty antecedent and iteratively adding the most promising attribute value constraint next. The classification accuracy

©2006-2014 Asian Research Publishing Network (ARPN). All rights reserved.


is used to rate candidate rules. This depth-first search continues until the resulting rule is specific enough that it makes no classification errors over the available data instances. Prism classifier generates output considering only high values, resulting in the rules, which contains only the best component, skipping the next components with same high value in the priority queue. Improving it, by considering all the high values, without considering only the first instance of high value, there by breaking the tie. Same rules are generated more than once, when resolving the tie condition, it is over come by, and associative classifiers are modeled to generate rules which are interpretable. But, these associative classifiers often consists of large rules generated, which makes decision making tough, and duplicate rules being generated. Hence, a tree model of associative classifier is developed, and it is clear that these classifiers are restricted in that at least one child node of a non-leaf node is never split. It is known as condition based tree, which we apply on data set, known as condition based classifier. CBC with feature selection has even a smaller number of rules [10].

Algorithm-1: Improvised PRISM

```
for each class C
prism(instance set)
        initialise E to the complete instance set
        while E contains instances with class C
                create empty rule R
until R is perfect (or no more attributes)
                for each attribute A not in R, and each value
        v,
                consider adding A=max(v) to R
                add A=v to R
                remove instances covered by R from E
                if(tie happens for max(coverage))
                prism(instance set)
end prism
```

Consider generating a rule for Average class; hence first consider the members who belong only to Average class:
Rule: If….AND…. then AVERAGE

Table-8. Dataset containing translator of only average class.

Translators	Usability	Number of languages	Human translation	Accuracy	Detect source language	characters	Website translation	Document translation	Class
babylon	good	21-40	Yes	good	No	501-1000	No	No	Average
MultiTrans	good	41-60	No	Good	Yes	0-500	No	Yes	Average
SDLTranslator	good	41-60	Yes	average	No	1000-5000	No	Yes	Average
systranet	good	0-20	No	good	No	>5000	Yes	Yes	Average

Calculating the coverage values for all the distinct values in the attributes.
Usability-good = 4/4 =1

Accuracy-good =3/4=0.75 etc, calculate for all the distinct values.

Since usability has highest coverage, it is pushed into the Rule and the table is reduced where members belong to only usability=good.

Rule-1: If Usability=GOOD AND …. Then AVERAGE

Table-9. Table reduced after Rule-1: If usability=GOOD AND …. Then AVERAGE.

Translators	Number of language	Human translation	Accuracy	Detect source language	Characters	Website translation	Document translation	Class
Babylon	21-40	Yes	Good	No	501-1000	No	no	Average
MultiTrans	41-60	No	Good	Yes	0-500	No	yes	Average
SDLTranslator	41-60	Yes	Average	No	1000-5000	No	yes	Average
Systranet	0-20	No	Good	No	>5000	Yes	yes	Average

©2006-2014 Asian Research Publishing Network (ARPN). All rights reserved.

We can see Usability attribute is removed. Calculating the coverage values for all the distinct values in the attributes, Accuracy-good = ¾ = 0.75, Detect source language – no =3/4 = 0.75, Website Tran – no = ¾ = 0.75, Document Translation - yes = ¾ = 0.75 Etc, calculate for all the distinct values. Since, there are 4 equal highest coverage values, this situation is known as TIE. Considering all of them, and calculating further, leads to a minimum of 4 different rules. Hence, the 4 rules are:

Rule-1.1: If Usability=GOOD AND Accuracy=good AND…. Then AVERAGE
Rule-1.2: If Usability=GOOD AND Detect source language = no AND…. Then AVERAGE
Rule-1.3: If Usability=GOOD AND Website Tran = no AND…. Then AVERAGE
Rule-1.4: If Usability=GOOD AND Document Translation = AND…. Then AVERAGE

Table-10. Table reduced after Rule-1.1: If Usability=GOOD AND Accuracy=good AND…. Then AVERAGE.

Translators	Number of languages	Human translation	Detect source language	Characters	Website translation	Document translation	Class
Babylon	21-40	Yes	No	501-1000	no	No	Average
MultiTrans	41-60	No	Yes	0-500	no	Yes	Average
Systranet	0-20	No	No	>5000	yes	Yes	Average

This Table contains only members abiding Rule 1.1: If Usability= GOOD AND Accuracy= good AND…. Then AVERAGE. Hence, iteratively calculating rules are generated until there is no more attribute left.

K) Statistical evaluation test

To assess the power of evaluation the statistical test that has been performed are T-test and ANOVA and they are as follows.

L) T-Test

T test is a statistical way which is used to determine how for the two samples are different from each other. It incorporates the mean value, and how far it varies from the mean values of both the samples. T-test here, determines, the outcomes of distributed measures and weighted attributes.

T-test= xbar-ybar /√var1/n+var2/n

where, Xbar- mean of training set 1, Ybar- mean of training set 2, var1- variance of training set 1, var2- variance of training set 2, n- Number of classes.

Table-11. Frequencies of translators based on class.

	X	Y
Excellent	3	3
Average	4	6
Poor	7	5

X- Distributed Measures Data set, Y- Weighted Attribute Data Set
Calculating weighted average: weighted average = 12355.55.
Calculating the Linearity Mapping between Distributed Measures and Weighted Attributes
T-test = 0.35729466420809775. Since 0.35729466420809775 is <1, the mean between both the dataset is not varying vast and the value is accepted.

M) Anova 1 way calculation

Anova evaluation is conducted to test the variation of mean among various groups, not just 2 groups. It evaluates the results considering every detail in data set, by considering classes in the dataset. The calculated values of ANOVA are shown in Table-10.

Table-12. Calculated values of ANOVA.

Source	SS	DF	MS	F-Ratio
Between	16000	2	8000	0.478
Within	13066.66	3	16711.108	

Legend-5: SS - Sum of Square, DF - Degrees of Freedom and MS - Mean Square

The level of significance taken is as 0.05 i.e., 95% level of significance. Then F-distribution value is found. The Table value is 19.16. Since 0.478< 19.16 the hypothesis H0 is accepted.

©2006-2014 Asian Research Publishing Network (ARPN). All rights reserved.

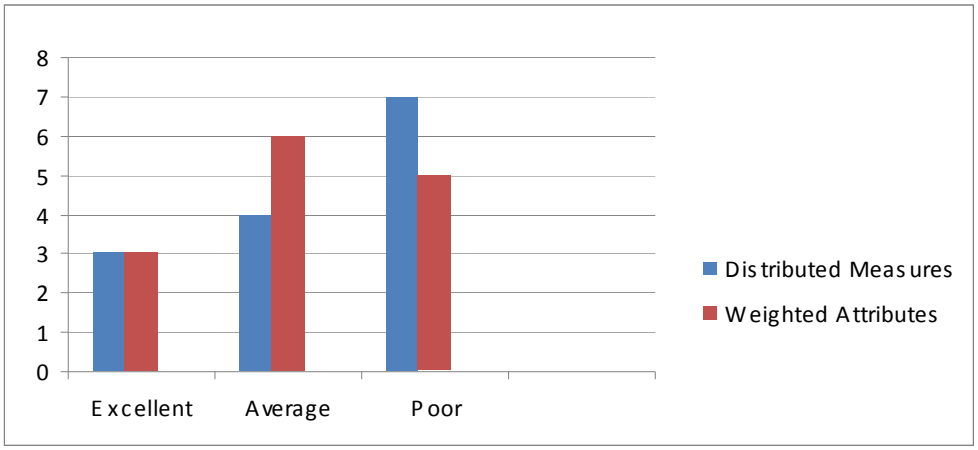

Figure-2. Distributed measure classification vs. Weighted attributes classification.

MODI'S METHOD

Table-13. VOGEL'S table with cost.

	HT	DT	WT	Us	NL	DSL	NC	ACC	SUPPLY
Reverso	303	606	908	1211	1514	1817	2119	2422	10900
Dictionary.com	3278	6556	9833	13111	16389	19667	22944	26222	118000
Babelfish	3472	6944	10417	13889	17361	20833	24306	27778	125000
Tv translator	4278	8556	12833	17111	21389	25667	29944	34222	154000
Promt	4333	8667	13000	17333	21667	26000	30333	34667	156000
NS translator	5250	10500	15750	21000	26250	31500	36750	42000	189000
SEF translator	5250	10500	15750	21000	26250	31500	36750	42000	189000
Babylon	5278	10556	15833	21111	26389	31667	36944	42222	190000
SDL translator	5833	11667	17500	23333	29167	35000	40833	46667	210000
Multi trans	8139	16278	24417	32556	40694	48833	56972	65111	293000
Worldingo	8222	16444	24667	32889	41111	49333	57556	65778	296000
Systranet	8639	17278	25917	34556	43194	51833	60472	69111	311000
Bing translator	8833	17667	26500	35333	44167	53000	61833	70667	318000
Google translator	10444	20889	31333	41778	52222	62667	73111	83556	376000
Demand	81552	163108	244658	326211	407764	489317	570867	652423	2935900

Legend-6: HT- Human Translation, DT- Document Translation, WT- Website Translation, Us- Usability, NL- Number of Languages, DSL-Detect Source Language, NC- Number of Characters, ACC- Accuracy.

For the calculation of MODI'S method total cost is taken into account from the Table-4. And the corresponding cost is multiplied with thousand. Thus the above Table is obtained and for the purpose of easier calculation, since the numbers are large we are dividing it by hundred. Rounding off is done and VOGEL'S method is applied in order to calculate the total cost. To obtain optimal cost MODI'S method is applied and the total cost is 10462668. In MODI'S u + v calculation there was no evidence of minus sign and hence the above said result is achieved.

©2006-2014 Asian Research Publishing Network (ARPN). All rights reserved.


Table-14. MODI'S unassigned rather watery cell table.

	HT	DT	WT	Us	NL	DSL	NC	ACC	SUPPLY
Reverso	-550	-445	-341	-253	-167	-85	-27	*	-550
	3	6	9	12	15	18	21	25	
Dictionary.com	-313	-208	-104	-16	70	152	210	*	-313
	33	66	98	131	164	197	229	262	
Babelfish	-297	-192	-88	0	86	168	226	*	-297
	35	69	104	139	174	208	243	278	
Tv Translator	-233	-128	-24	64	150	232	290	*	-233
	43	86	128	171	214	257	299	342	
Promt	-228	-123	-19	69	155	237	295	*	-228
	43	87	130	173	217	260	303	347	
NS translator	-155	-50	54	142	228	310	*	*	-155
	53	105	157	210	262	315	368	420	
SEF translator	-155	-50	54	142	228	310	*	420	-155
	53	105	157	210	262	315	368	420	
Babylon	-154	-49	55	143	229	311	*	421	-154
	53	106	158	211	264	317	369	422	
SDL translator	-115	-10	94	182	268	*	*	460	-115
	58	117	175	233	292	350	408	467	
Multi trans	23	128	232	320	406	*	546	598	23
	81	163	244	326	407	488	570	651	
Worldingo	28	133	237	325	*	*	551	603	28
	82	164	247	329	411	493	576	658	
Systranet	49	154	258	*	*	514	572	624	49
	86	173	259	346	432	518	605	691	
Bing Translator	56	161	*	*	439	521	579	631	56
	88	177	265	353	442	530	618	707	
Google Translator	*	*	*	401	481	569	627	679	104
	104	209	313	418	522	627	731	836	
Demand	0	105	209	297	383	465	523	575	

The main objective in the Transportation Problem is to reduce the transportation costs. By using Dual simple method, two phase method and Big M method problem is solved with the help of Tora software and this solution is compared with solution obtained from Vogel's Approximation Method [11]. Vogel's Approximation Method which incorporates Total Opportunity Cost concept were thoroughly analyzed and while experiments being carried out it was found that VAM with TOC yields a better solution [12]. VAM provides good solution for the transportation problems. VAM is analyzed through iterations and again it provides a good solution in combination with Total Opportunity Cost [13]. Modified Vogel's Approximation Method is proposed for solving Fuzzy Transportation Problems and it provides a great solution than previously existing methods [14]. A variant of VAM was proposed by using Total Opportunity Cost and alternative allocation costs. While carrying out an experiment it was found that improved version of Vogel's Approximation Method provides an efficient solution than VAM [15].

Legend-7: HT- Human Translation, DT- Document Translation, WT- Website Translation, Us- Usability, NL- Number of Languages, DSL-Detect Source Language, NC- Number of Characters, ACC- Accuracy.

©2006-2014 Asian Research Publishing Network (ARPN). All rights reserved.


For calculation purpose the above said Table is normalized by diving all the numbers by hundred (100). Through this the outcome of the result should be multiplied by the number hundred in order to get the real optimized cost.

RESULTS AND DISCUSSIONS

Improvised High Hit Ratio Classifier is applied for the prediction of the translator web service which is one of the improvised PRISM proposed in this paper. T-Test for variance is found to be 0.357. Since it is less than 1, the value is accepted. 0.478 is obtained as the result of one way ANOVA and the hypothesis H0 is accepted. Jaccard distance is calculated to find the relevancy available between excellent, average and poor components and the value is found to be 0.0625, 0.0512 and 0.3076. Cosine distance is calculated for excellent, average and poor the corresponding value is 0.0509, 0.0426 and 0.4706. The variance is calculated and is found to be negligible. Through this we can conclude that expected entry level prediction is more coincide with the methodological outcomes. For cost effectiveness MODI'S optimized method is employed and appropriate cost value is arrived after the application of VOGEL'S principle the above said was obtained and the cost is calculated as 10462668. The optimized cost for purchasing excellent component is less than or equal to rupees 1032917 is the worthwhile price for purchasing an above said component. In similar manner prices for Average and Poor are 939421 and 345477.

CONCLUSIONS

The optimal component selection with cost effectiveness is achieved via IH2RC [Improvised High Hit Ratio Classifier] rather improvised PRISM and MODI'S method is engaged to know the optimized cost for the components like Excellent, Average and Poor. For every selection of the component the comparison between the category of component with cost is very much be appreciated.

ACKNOWLEDGEMENT

We appreciate Mr. S. Sudharsun of M.Tech, Advanced Computing Discipline, School of Computing, SASTRA University for his un-payable and tireless work in bringing up MODI'S optimization method.

REFERENCES

[1] Krzysztof Trawiński, Oscar Cordón, Arnaud Quirin and Luciano Sánchez. 2013. Multi objective genetic classifier selection for random oracles fuzzy rule-based classifier ensembles: How beneficial is the additional diversity? Knowledge-Based Systems. 54: 3-21.

[2] Francisco Rodrigues Lima Junior, Lauro Osiro and Luiz Cesar R. Carpinetti. 2013. A fuzzy inference and categorization approach for supplier selection using compensatory and non-compensatory decision rules. Applied Soft Computing. 13(10): 4133-4147.

[3] Stefanos K. Goumas, Ioannis N. Dimou and Michalis E. Zervakis. 2010. Combination of multiple classifiers for post-placement quality inspection of components: A comparative study. Information Fusion. 11(2): 149-162.

[4] Michela Fazzolari, Bruno Giglio, Rafael Alcalá, Francesco Marcelloni and Francisco Herrera. 2013. A study on the application of instance selection techniques in genetic fuzzy rule-based classification systems: Accuracy-complexity trade-off. Knowledge-Based Systems. 54: 32-41.

[5] Shu-hsien Liao, Hsu-hui Ho and Feng-chich Yang. 2009. Ontology-based data mining approach implemented on exploring product and brand spectrum. Expert Systems with Applications. 36(9): 11730-11744.

[6] Herman Hartmann, Mila Keren, Aart Matsinger, Julia Rubin, Tim Trew and Tali Yatzkar-Haham. 2013. Using MDA for integration of heterogeneous components in software supply chains. Science of Computer Programming. 78(12): 2313-2330.

[7] P.C. Jhaa, Vikram Balib, Sonam Narulaa and Mala Kalrac. 2014. Optimal component selection based on cohesion and coupling for component based software system under build-or-buy scheme. Journal of Computational Science. 5(2): 233-242.

[8] Chintakindi Srinivas, Vangipuram Radhakrishna and C.V. Guru Rao. 2013. Clustering Software Components for Program Restructuring and Component Reuse Using Hybrid XOR Similarity Function. AASRI Procedia. 4: 319-328.

[9] Gabriela Czibula, Zsuzsanna Marian and Istvan Gergely Czibula. 2014. Software defect prediction using relational association rule mining. Information Sciences. 264: 260-278.

[10] Houtao Deng, George Runger, Eugene Tuv and Wade Bannister. 2014. CBC: An associative classifier with a small number of rules. Decision Support Systems. 59: 163-170.

[11] Gaurav Sharma and S.H. Abbas. 2012. Vijay Kumar Gupta. Solving Transportation Problem with the various method of Linear Programming Problem. Asian Journal of Current Engineering and Maths. 1(3): 81-83.

[12] M. A. Hakim. 2012. An Alternative Method to Find Initial Basic Feasible Solution of a Transportation

©2006-2014 Asian Research Publishing Network (ARPN). All rights reserved.


Problem. Annals of Pure and Applied Mathematics. 1(2): 203-209.

[13] Shweta Singh, G.C. Dubey and Rajesh Shrivastava. 2012. Optimization and analysis of some variants through Vogel's approximation method. IOSR Journal of Engineering. 2(9): 20-30.

[14] A. Edward Samuel and M. Venkatachalapathy. 2011. Modified Vogel's Approximation method for Fuzzy Transportation Problems. Applied Mathematics. 5(28): 1367-1372.

[15] Serdar Korukoglu and Serkan Balli. 2011. An improved Vogel's Approximation Method for the Transportation Problem. Mathematical and Computational Applications. 16(2): 370-381.