Building a Financial Case-Based Reasoning Prototype from Scratch with Respect to Credit Lending and Association Models Driven by Knowledge Discovery

Jürgen Hönigl
Institute for Application-Oriented Knowledge Processing
Johannes Kepler University
Linz, Austria
juergen.hoenigl@jku.at

Yuliya Nebylovych
Department for Applied Linguistics
Lviv Polytechnic National University, 11th group
Lviv, Ukraine
julianebylovych@gmail.com

Abstract in English—**Credit lending can be seen as a challenging task due to many available procedures such as the cash flow analysis and scoring methods. Using Case-Based Reasoning (CBR) and knowledge discovery to support and gain comparison and risk assessment of loan cases are demonstrated in this paper. The knowledge discovery of a credit data set was made with open source algorithms using Waikato Environment for Knowledge Analysis (WEKA). Building a prototype from scratch can be seen as an interesting task when considering the full CBR methodology especially with heterogeneous input data schemes.**

Keywords- Case-Based Reasoning; Credit Lending; Knowledge Discovery; WEKA;

Abstract in Russian—**Выдачу кредитов можно рассматривать как сложную задачу из-за множества доступных процедур, таких как анализ денежных потоков и методы оценки. В этой статье показано использывание системы рассуждений на основе прецедентов (CBR) и выявление знаний для поддержки и получения сравнения и оценки случаев риска кредита. Виявление знаний о кредитном наборе данных было сделано с помощью открытых исходных алгоритмов, используя алгоритм среды Вайкато для анализа знаний (WEKA). Создание прототипа с нуля можно рассматривать как интересную задачу при изучении вопроса о полной методологии CBR, особенно с гетерогенной схемой ввода данных.**

Ключевые слова - система рассуждений на основе прецедентов; выдача кредитов; выявление знаний; среда Вайкато для анализа знаний (WEKA).

I. INTRODUCTION

Financial reasoning to gain a quality factor concerning bank lending in comparison with static credit investigation company scores will be a challenge, especially when the alpha and beta error should be decreased. The alpha error occurs when a credit was incorrectly granted whereby the beta error occurs when a credit was incorrectly rejected. Minimizing the alpha error will avoid loan loss for the credit grantor which can be seen as a really good motivation. The problem statement can be defined as reducing the Alpha error regarding credit lending when using Case-Based Reasoning which is rather an interesting field of research especially if loan loss within different countries will be remembered such as Japan at the end of the eighties or the subprime crisis 2008 in the United States of America. Previous experience was enumerated with the related research section. Key components of this approach will be demonstrated within the components and prototype section.
Results of the knowledge discovery are documented in the data analysis section. This paper describes the current state of work in progress.

II. RELATED RESEARCH

This paper is a work in progress paper concerning the proof of concept for a doctoral thesis. Therefore, references of the related research will be provided for further reading. Previous development regarding decision support systems started within the research of cognitive processes and problem solving. Miller described within his work 'The Magical Number Seven, Plus or Minus Two Some Limits on Our Capacity for Processing Information' in 1955 his research concerning the field of cognitive psychology. He described the information measurement and the short-term capacity of human beings which was an early step towards researching problem solving. [1] Many general facts and systems are demonstrated by Klein within his work ‘Knowledge-based Decision Support Systems’. A noteworthy expert system arises in 1981 which was called BANKER. It will be explained within a few lines to demonstrate the evolution of early systems to the present. The inference structure of BANKER was divided into four parts namely input, calculations, reasoning and conclusions. Pro forma income/balance can be defined as input, cash flow model will be used for calculations, financial analysis was described as reasoning and in each case the final credit rating was the conclusion. [2]
The knowledge base of BANKER was defined by rules which would be nowadays a part of the algorithm part. The whole system was more a static system than a modern reasoning system which clearly shows that the term 'reasoning' has changed over the years.

Case-Based Reasoning was documented within a paper by Aamodt and Plaza which will be described in the section 'CBR in a Nutshell'. Many CBR applications within different domains were developed within the INRECA (INduction and REasoning from CAses) project. [3] An extended overview about CBR systems can be seen within ‘On Reasoning within different Domains in the Past, Present and Future’. [4]
CreditCBR was an approach developed within INRECA. The approach of their work was using weights for attributes in combination with k-Nearest Neighbor querying to improve reasoning results. [5]
Popular approaches and procedures for loan decision making such as cash flow analysis methods and multivariate ratio analysis were published within ‘Credit Engineering for Bankers’ beside other issues like risk rating and risk analysis. Time series will be a rather good method for risk assessment. It distinguishes between different time series method which can detect a trend and take care about seasonality. For instance Single Moving Average detects no trend and no seasonality but Hold-Winter’s Additive works with both trend and seasonality. [6]

TABLE I. ALPHA AND BETA ERROR

Solvency Assumed	**Solvency Actually**	
	Good	***Bad***
Good	Type I accuracy	Alpha Error
Bad	Beta Error	Type II accuracy

The alpha and beta errors are a classification of customers as mentioned with the introduction which is displayed within table I. Conventional techniques for credit lending are including scoring procedures like Altman’s Z-score. [7] This score distinguishes between different kinds of customers such as private firms and non-manufacturer industrials which results in different weights within his formula. However, a part

of the discrimination is called ‘zone of ignorance’ which can be seen as a rather high possibility for misclassification. In these grey zones an additional technique like reasoning can help to reduce the alpha errors for the loaner.

III. CBR IN A NUTSHELL

A case consists of a problem, a solution and some annotations if necessary.
The R4 model published by Aamodt and Plaza describes the workflow which starts with a new given problem and results into a new case. [8]

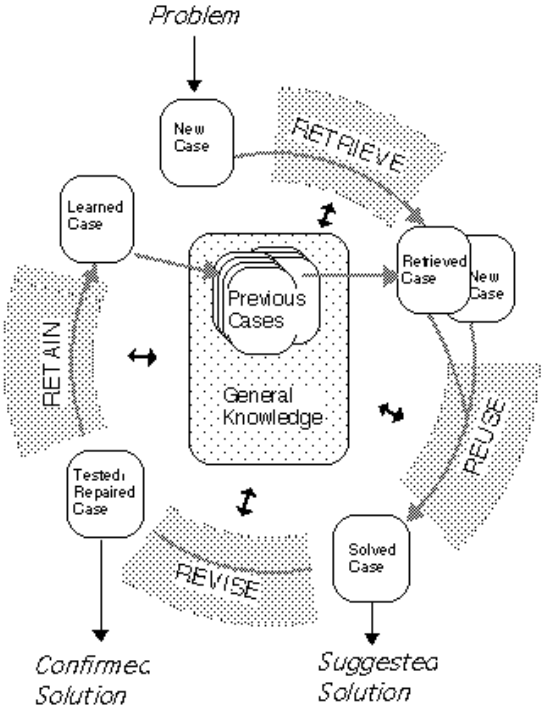

Figure : CBR cycle by Aamodt

Firstly, a new given problem will be handed over to a CBR system. Similarity measures are used within the retrieve step to get the nearest cases from the case base which can be described as the knowledge of the system. A solution will be used or adapted, if necessary, within the reuse step. Afterwards a solved case can occur which will be revised. If the solution has to be changed after an evaluation, then a user can use the graphical user interface to modify it manual which leads into a newer version of the case. This case will be retained if it will be additional information for the case base.

IV. SIMILARITY MEASURES

Zezula explains within ‘Similarity Search - The Metric Space Approach’ many issues. However, distance measures (e.g. Jaccard's Coefficient) are not currently suitable for querying the nearest cases but a similarity query, such as the nearest neighbor query, will be adequate. A range query will be suitable to gain the cases within a given range but the count of these cases cannot be defined which will be an advantage when using a k-Nearest Neighbor query. [9]

k-Nearest Neighbor (k-NN) will be used to retrieve similar cases for the first version of the prototype[1]. However, Statlog, a former project within the European Union which evaluated machine learning methods, demonstrated that k-NN provides rather good classification results for the German credit data set in comparison to other methods such as NaiveBayes or the C 4.5 decision tree. However, the C 4.5 algorithm can have a slightly better classification but with over proportional costs. Statlog presented a result that three of the top six algorithms were decision trees (Cal5, C4.5 and IndCART). However, the algorithm in second place (DIPOL92) is similar with a neural network. [10]
Big data which is an upcoming issue would be handled by a database management system such as Oracle Database or SQL Server. However, the CBR retrieve step will be used in combination with an adequate similarity measure to perform the random access memory operations when dealing with the nearest cases.

Different procedures are used for single persons or small and medium-sized enterprises. Therefore, a case base will include only one type of customer which affects a similarity measure. The kNN query will operate on only one case base for the current type of customer.

V. DATA ANALYSIS

The term data mining is used everywhere – even within WEKA documentations. However, we do not have to grab for data because we mostly have it already. Therefore, our goal is searching and obtaining knowledge within data and databases which refers to the term ‘Knowledge Discovery’.

Two Statlog data sets were compared. [11] The German Credit Data Set provided by Statlog provides categorical and numerical attributes. The credit amount and the desired duration of the payments are numerical values. Attributes such as the purpose of the credit and the credit history of the customer are described as qualitative attributes. For instance the credit history contains different possibilities like no credits taken/all credits paid back duly, all credits at this bank paid back duly, existing credits paid back duly till now, delay in paying off in the past, critical account/other credits existing but not at this bank.

The Japan Credit Data Set of Statlog provided an interesting List Processing (LISP) code which contains rules for banking. A LISP method was discovered about unmarried women who will never get a loan without checking their income but this was a data set of the early nineties.

1 Prototype means in this context the Proof of Concept. However, the term ‘prototype’ exists within the domain of Case-Based Reasoning as a special kind of representative for cases within a case base.

Waikato Environment for Knowledge Analysis (WEKA) was used to gain knowledge of the German credit data set. All WEKA algorithms are available as open source. [12]

The German Data Set described the alpha and beta errors with other words and defined them within a cost matrix. It was defined as cost intensive (cost 5) if a customer will be classified as good when they are bad concerning their solvency in comparison to class a customer as bad when they are good (cost 1). The cost factor 5 was assigned to the alpha error, the beta error was associated with 1. However, the beta error is not really mostly measurable.

The HotSpot association algorithm was used to gain rules towards a target variable of interest. Different arguments can be adjusted to obtain different models. These arguments are the target index which defines the target attribute, the value of the target, the segment size and a numeric value which can be used to determine the maximum branching factor to define the resulting model as specific or generic.

These rules were provided with a minimum segment size of 330 instances and a branching factor of two. We can see that the attribute which describes the credit amount occurs in both branches – once in the first level and once in the second level. A14 means that a checking account does not exist, A211 describes a good cost factor regarding the risk of payback .

```
Cost_Factor=A211 (70% [700/1000])
  Status_Checking_Account = A14 (88.32%
[348/394])
  |   Credit_Amount <= 7824 (89.67%
[330/368])
  Duration_Months <= 15 (79.35%
[342/431])
  |   Credit_Amount <= 3973 (80.88%
[330/408])
```

Using 650 instances will lead to a simpler model.

```
Cost_Factor=A211 (70% [700/1000])
  Duration_Months <= 39 (72.11%
[662/918])
  |   Credit_Amount <= 8613 (73.36%
[650/886])
  Credit_Amount <= 7476 (72.07%
[658/913])
```

Using 290 instances will lead not to a very specific model with a branching factor of 3 because the maximum support segment size was limited to 300 instances when using the category A212 (suboptimal risk concerning payback). A201 describes only the code for ‘yes’.

```
Cost_Factor=A212 (30% [300/1000])
Duration_Months > 8 (32.01% [290/906])
Foreign_Worker = A201 (30.74% [296/963])
  |   Credit_Amount > 601 (31.17%
[293/940])
Age <= 61 (30.46% [293/962])
  |   Credit_Amount > 601 (30.88%
[290/939])
```

This HotSpot association algorithm clearly shows that increasing of the segment size (argument S) will mostly lead to a simpler association model. With a small supported segment size such as 30 percent (300 instances when using Cost_Factor=A212) the count of lines within an association model can increase with a minor change of the argument segment size of the algorithm. For instance, the branching factor 2 will be used with a maximum of 30% regarding the supported segment size, then 21 lines with a depth of 5 will reached when using 17% as argument S but only 2 lines will reached when using 22% as argument S.

Certain attributes, such as the employment history, which can be used to describe a customer, can be detected by associations. In the following result a segment size of 22 percent, A75 (code for >= 7 years) and two branches were selected as arguments which results in this association.

```
Present_Employment_Since=A75 (25.3%
[253/1000])
  Age > 29 (35.93% [226/629])
  |   Foreign_Worker = A201 (36.53%
[221/605])
  Present_Residence_since > 1 (27.93%
[243/870])
  |   Age > 28 (36.91% [220/596])
  |
Installment_Rate_in_Percentage_of_Disposa
ble_Income > 1 (29.02% [220/758])
```

The top level branches are Age>29 and Present Residence > 1. We can see that the second branch is using the age attribute within a sub branch.

If we are using a maximum branching factor of seven, then it will generate a very specific and detailed model which contains almost similar rules which contains Age, Duration_Months and sometimes Credit_Amount. However, these are rather good attributes to clarify a decision and build a association model but redundant rules should be avoided which isn’t the case when using a maximum branching factor like seven.

```
Present_Employment_Since=A75 (25.3%
[253/1000])
  Age > 29 (35.93% [226/629])
  |   Foreign_Worker = A201 (36.53%
[221/605])
  Present_Residence_since > 1 (27.93%
[243/870])
  |   Age > 28 (36.91% [220/596])
```

```
  |
Installment_Rate_in_Percentage_of_Disposa
ble_Income > 1 (29.02% [220/758])
  |   Other_Debtors_Guarantors = A101
(28.64% [226/789])
  |   |   Age > 23 (31.08% [221/711])
  |   |   Foreign_Worker = A201 (29.15%
[223/765])
  |   |   |   Age > 22 (30.36% [221/728])
  |   Foreign_Worker = A201 (28.43%
[238/837])
  |   |   Age > 27 (36.18% [220/608])
  |   |   Duration_Months <= 36 (28.96%
[221/763])
  |   |   Credit_Amount <= 7763 (28.83%
[222/770])
  |   Duration_Months <= 36 (28.43%
[226/795])
  |   |   Age > 23 (30.9% [220/712])
  |   Credit_Amount <= 8335 (28.22%
[230/815])
  |   |   Age > 24 (31.66% [221/698])
  |   |   Foreign_Worker = A201 (28.83%
[226/784])
  |   |   |   Age > 23 (31.34% [220/702])

Installment_Rate_in_Percentage_of_Disposa
ble_Income > 1 (26.39% [228/864])
  |   Age > 24 (29.62% [221/746])
  |   Foreign_Worker = A201 (26.82%
[225/839])
  |   |   Age > 23 (28.99% [220/759])
  Credit_Amount <= 6458 (25.86%
[226/874])
  |   Age > 23 (28.31% [220/777])
  |   Foreign_Worker = A201 (26.43%
[222/840])
  Other_Debtors_Guarantors = A101 (25.8%
[234/907])
  |   Age > 25 (30.12% [222/737])
  |   Foreign_Worker = A201 (26.25%
[231/880])
  |   |   Age > 24 (29.81% [223/748])
  Foreign_Worker = A201 (25.75%
[248/963])
  |   Duration_Months <= 36 (26.23%
[230/877])
  |   |   Age > 24 (29.8% [222/745])
  Duration_Months <= 36 (25.74%
[235/913])
  |   Age > 26 (31.75% [220/693])
```

Another configuration demonstrates a new aspect of a rule such as a greater age requirement in comparison to previous models.
The supported segment size was defined as seventeen percent but only one branch was allowed. However, we can see that only one branch will provide a new view towards the data but it will be mostly too imprecise.

```
Present_Employment_Since=A75 (25.3%
[253/1000])
  Age > 35 (43.93% [181/412])
  |   Present_Residence_since > 1 (45.53%
[173/380])
  |   |   Age <= 66 (46.2% [170/368])
```

Concerning the occurrence of attributes it was a clear result that the age attribute was used over proportional in comparison to another attribute such as another debtors guarantors.

A part of the parser and the properties within the current prototype were developed on the basis of knowledge discovery made with WEKA.

VI. Problem vs Case

A new given problem will be an important part of a case but will not be the whole case because a solution has to be attended.
Therefore, a definition of a first simple input query for a problem will be given in this section. A solution and a case are defined in addition to have a complete basis definition for a case base.

At least seven attributes must be provided for a minimum query regarding a new given problem (1) into the prototype concerning the case base.

$$\text{Problem} = \{\text{Age, Credit Amount, Credit History, Duration, Income, Purpose}\} \quad (1)$$

These arguments which were used within (1) will be used for both pre-processing with static banking rules and the reasoning process itself. A query for a problem (2) can be extended when append an additional attribute such as another debtors guarantors.

$$\text{Problem} = \{\text{Age, Credit Amount, Credit History, Duration, Income, Other Debtors Guarantors, Purpose}\} \quad (2)$$

A solution (3) will contain two elements.

$$\text{Solution} = \{\text{Cost Factor, Recommendation}\} \quad (3)$$

The cost factor will be a numerical value which demonstrates the reasoned costs when granting a credit to the customer. The recommendation will be a percentage value in relation to retained cases which indicates the occurrence of this kind of customer.

Notes will be provided to the case in addition to explain different circumstances of a customer, for instance unpunctual and reticent when having a discussion concerning loan.

Therefore, we can define our case (4) which includes three different elements.

$$Case=\{Problem, Solution, Notes\} \quad (4)$$

VII. Components of the prototype

An overview about the reasoning cycle and implementation components are displayed within Figure . C# will be used for the whole proof of concept namely Retrieve, Reuse, Revise and Retain – and for pre-processing. These steps are presenting the software application part. The graphical user interface will be defined in extensible application markup language (XAML) as a Windows Presentation Foundation application. It has mainly an importance for the pre-processing, retrieve and revise phase. Outliers will be displayed within pre-processing. The retrieve step can provide and display a range of cases. The revise phase can be used to evaluate a new solution by a user which requires an adequate user interface. Language Integrated Query (LINQ) will be a technique to deal with cases when select and insert them. Knowledge obtained with WEKA and R will be basis for the decision making within the reuse phase if a solution will be usable for a new given problem.

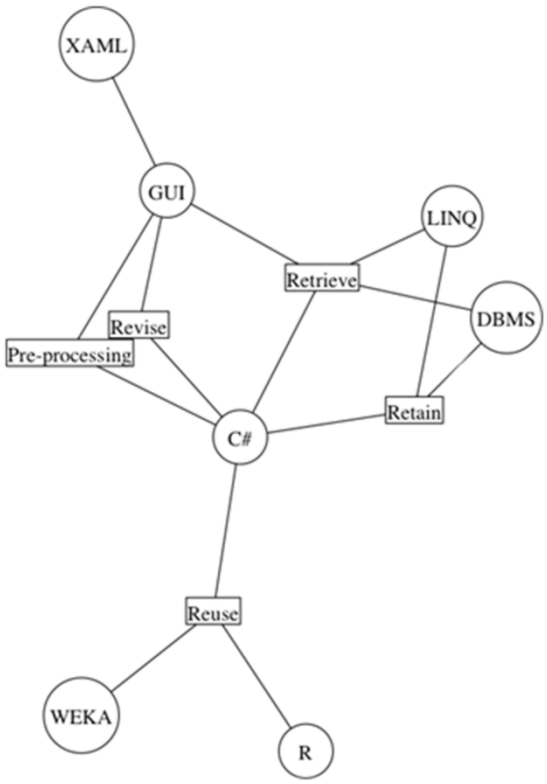

Figure : Overview

Simple static banking rules are integrated which avoid an inflation of the case base and enable a feedback of an application user. Statistical methods can be obtained by reusing statistical computing sources developed in R. Therefore, a port for R was defined. Oracle Database or SQL Server can be used in addition if the case base will grow. Database management system (DBMS) was written instead of Oracle Database and SQL Server because the decision towards a DBMS wasn't done heretofore.

VIII. Prototype

Static rules will be used within pre-processing to provide asserting of some queries which contain values out of regular borders. For instance an unemployed person without any bankable collateral but with a rather negative credit history will be marked and displayed within the user interface. Therefore, it is not suitable to build a new case when out of border values are occurring to reduce runtime of the reasoning process for other new given problems and subsequent cases. The misogynistic rule of the mentioned Japan credit data set within the data analysis section was not used. In addition these static rules will be suitable concerning the attribute age. If the age of a person plus credit duration will be greater than sixty, then this entry will be marked to show it within the pre-processing phase to a user.

Visual Studio 2010 Ultimate and C# are used to develop the proof a prototype for a doctoral thesis. The output of the risk assessment will be showed regarding to the role of the user. A bank manager would see only a numeric value which demonstrates either a value of the cost matrix or a percentage value regarding the risk.

The application which is under development has currently the version 0.1 pre-Alpha and includes a parser for the mentioned German data set and some static rules. Therefore, many issues are noted within the future work section.

IX. Conclusion

Old systems such as BANKER were state of the art many years ago but nowadays almost everything has changed. Using mainly static rules for a reasoning application will not be adequate.

Developing a similarity measure or another method which provides a good classification with low (runtime) costs would be cost intensive regarding the return of investment because this area was researched concerning many aspects. Therefore, an existing similarity measure such as k-NN or the C 4.5 decision tree will be adequate for reuse existing sources but for special purposes a modeling and development of a similarity measure can increase the results within the retrieve phase of a Case-Based Reasoning system.

The definition of a case, problem and solution was a pre-condition for the basis of a case base. Therefore, a generic query which fits any given new problem can be used for

requests to the CBR system. There is still a need for additional queries which will be covered in detail within the future work section. Cases bases must not contain different kinds of customers due to their different representations of their data. For instance a single customer doesn't provide attributes such as working capital, market value of equity or sales which will be the suitable for a small and medium enterprise.

The association models have to keep a balance between an excessive amount of rules and only a single rough rule. Too many rules are really specific for a data set and cannot be used for a generic association model. However, using only one rule will lead to a successful association model because that's too imprecise.

X. Future Work

Gain knowledge about a further data set of a financial institute to extend the case base will be an issue.
On the other site many issues have to be developed regarding both the graphical user interface and the backend. The user interface has to extend with additional controls for different roles of users. The backend has to be developed towards a full support of Aamodt's R4 model which includes current missing sub-methods such as Adapt which can be used to transform an inadequate solution to a suitable one. Attend a port for Graphviz - Graph Visualization Software - can be seen as an additional activity which would generate graphs to explain relations between cases. Integrate an R port for using time series will be another task to gain forecasts which improves the quality of the reasoning result.

Extend the query for a new given problem will be suitable to gain more similar cases instead of using a minimum query. However, similar cases which are not containing extended attributes will not be retrieved as similar which would be a drawback. Therefore, tests will be made to compare an extended query and a minimum query concerning the results.

Model and implement a similarity measure will be an additional step which encapsulates different attributes regarding loan lending.

Different case bases have to establish to avoid decrease the quality of the reasoning process. A case base contains customers such as the German credit data set which was demonstrated. Another case base can contain small and medium enterprises. These cases can be validated and partially revised with Altman's Z-score to gain a comparison between reasoning and scoring. The Z-score provides three zones of discrimination at the end of the scoring namely safe zone, zone of ignorance and distress zone which can be seen as a rather good classification to compare it with reasoning results. [7] In addition to revise a case by a score it will be evaluated by a user to avoid and reduce both alpha and beta errors.

References

[1] G. A. Miller, "The Magical Number Seven, Plus or Minus Two Some Limits on our Capacity for Processing Information," 1955, the American Psychological Association.

[2] M. R. Klein and L. B. Methlie, *Knowledge-Based Decision Support Systems With Applications in Business*. Wiley, 1995.

[3] R. Bergmann, K. D. Althoff, S. Breen, M. Göker, M. Manago, and S. Wess, *Developing Industrial Case Based Reasoning Applications: The INRECA Methodology*. Springer Verlag, 2003, vol. Lecture Notes in Artificial Intelligence Berlin, LNAI 1612, Berlin.

[4] J. Hönigl, H. Kosorus, and J. Küng, "On Reasoning within Different Domains in the Past, Present and Future," in *23rd Database and Expert Systems Applications (DEXA), 2012. 2nd International Workshop on Information Systems for Situation Awareness and Situation Management - ISSASiM'12*, 2012.

[5] W. Wilke, "CreditCBR: Fallbasierte Entscheidungsunterstützung mit INRECA," 1996.

[6] M. Glantz and J. Mun, *Credit Engineering for Bankers, Second Edition: A Practical Guide for Bank Lending*. Academic Press, 2010.

[7] E. I. Altman, "Predicting Financial Distress of Companies: Revisiting the Z-Score and Zeta Models," 2000.

[8] A. Aamodt and E. Plaza, "Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches," *AI Commun.*, vol. 7, no. 1, pp. 39–59, 1994.

[9] P. Zezula, G. Amato, V. Dohnal, and M. Batko, *Similarity Search: The Metric Space Approach (Advances in Database Systems)*. Springer, 2010.

[10] D. Michie, S. D.J., and T. C.C., *Machine Learning, Neural and Statistical Classification*. Statlog Project Report, 1994.

[11] A. Frank and A. Asuncion, "UCI Machine Learning Repository," 2010. [Online]. Available: http://archive.ics.uci.edu/ml

[12] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA Data Mining Software: An Update," *SIGKDD Explor. Newsl.*, vol. 11, no. 1, pp. 10–18, Nov. 2009. [Online]. Available: http://doi.acm.org/10.1145/1656274.1656278