Published in Expert Systems, Vol. 33, No. 2, pp. 145-160, 2016

Assessment of Life Insurance Applications: An Approach Integrating Neuro-Symbolic Rule-Based with Case-Based Reasoning

Jim Prentzas[a*], **Ioannis Hatzilygeroudis**[b]
[a] Democritus University of Thrace, School of Education Sciences
Department of Education Sciences in Early Childhood
68100 Nea Chili, Alexandroupolis, Greece
dprentza@psed.duth.gr, jnprentzas@gmail.com
Phone number: +30-25510-30086

[b] University of Patras, School of Engineering
Department of Computer Engineering & Informatics
26500 Patras, Greece
ihatz@ceid.upatras.gr
Phone number: +30-2610-996937

*Corresponding author

Abstract. Assessment of applications for life insurance is an important task in the insurance sector that concerns estimation of potential risks underlying an application, if accepted. This task is accomplished by specialized personnel of insurance companies. Due to recent financial crises this task is more demanding and intelligent computer-based methods could be employed to assist. In this paper, we present an intelligent approach to assessment of life insurance applications, which is based on an integration of neurule-based with case-based reasoning. Neurules are a type of neuro-symbolic rules that combine a symbolic (production rules) and a connectionist (adaline unit) representation. A characteristic of neurules is that in contrast to other hybrid neuro-symbolic approaches, they retain the naturalness and modularity of symbolic rules. Neurules are produced from available symbolic rules that represent general knowledge, which however do not completely cover the domain. We use health condition, age, gender, annual income, profession, insurance type and primary life insurance benefit as assessment parameters used in rule conditions. The integration of neurules and cases employs different types of indices for the cases according to different roles they play in neurule-based reasoning. This results in its accuracy improvement. Experimental results demonstrate the effectiveness of the approach.

1. Introduction

Insurance companies handle many applications submitted by individuals, companies, organizations, banks and financial institutions. There are different types of insurance. For instance, individuals on the one hand may have insurance for assets such as buildings, vehicles, artwork and jewelry and they may also have personal insurance such as life insurance and accident insurance. Applications for insurance are assessed by the personnel of insurance companies. Assessment is a process rating the potential risks underlying an application. If no such risks exist or they are considered acceptable, the application is approved, otherwise it is rejected. Assessment of

© Wiley-Blackwell

insurance applications is an important task in insurance companies resulting in discriminating between risky applications that may result in financial losses and promising ones that are likely to bring in revenue. Recent financial crises have been a catalyst for insurance companies to investigate methods which improve application assessment.

A common type of insurance for individuals is life insurance. Applications for life insurance need to satisfy a number of critical requirements in order to be approved. The assessment of submitted applications takes into consideration various parameters related to the applicant. Quality of assessment is based on the experience of the personnel. High level personnel for application assessment are not always available. Also, sometimes assessment may be not perfectly consistent. So, a computer based decision making approach could be useful in both cases and also as an expert level support tool.

Artificial Intelligence (AI) methods could be used to support such a decision making process. There are some requirements in designing an intelligent system for the assessment of life insurance applications. First, experience of insurance companies' personnel specialized in life insurance is useful in order to outline insurance attributes, applicant attributes and assessment criteria. Second, available cases from past insurance applications are required to design and test the system. Third, explanations concerning the reached decisions are necessary.

Various AI approaches have been applied to life insurance processes and to application assessment specifically (e.g. Byczkowska-Lipinska et al., 2009; Hsieh and Wang, 2011; Kumar and Pandley, 2012). Life insurance is a financial domain in which AI methods have proved fruitful. In most cases, a single AI method is used, e.g. a rule-based approach (Byczkowska-Lipinska et al., 2009) or a back propagation neural network approach (Kumar and Pandley, 2012). However, an interesting case would be to use an AI approach that combines more than one intelligent method. A recent research direction in AI involves the integration (or combination) of two or more intelligent methods (Sahin et al., 2012; Hatzilygeroudis and Prentzas, 2011). These efforts aim to overcome the disadvantages of the combined methods by exploiting their cumulative advantages. Several types of integrated approaches have been proposed. Examples of such combinations include neuro-symbolic approaches integrating neural networks with symbolic methods (Garcez and Lamb, 2011; Hatzilygeroudis and Prentzas, 2004b), neuro-fuzzy approaches integrating neural networks with fuzzy methods (Chattopadhyay, 2014; Lin et al., 2012), approaches combining neural networks and genetic algorithms (Belciug and Gorunescu, 2013) and approaches combining case-based reasoning with other intelligent methods (Prentzas and Hatzilygeroudis, 2009).

An approach that seems to fit in making decisions on life insurance applications is one that combines (general) rule-based knowledge of the life insurance domain with (specific) knowledge based on available past cases. Such approaches, integrating rule-based and case-based reasoning, have proven effective and are becoming increasingly popular in various fields (Prentzas and Hatzilygeroudis, 2007). The complementary advantages and disadvantages of rule-based and case-based reasoning are a good justification for their possible combination. Symbolic rules represent general knowledge and exhibit advantages such as naturalness, modularity and ease of explanation. Their major drawbacks involve difficulties in knowledge acquisition that may result in imperfections and deficiencies in covering the full complexity of the domain and difficulties in reasoning with unknown or unexpected inputs (Prentzas and Hatzilygeroudis, 2007). Case-based reasoning exploits specific knowledge incorporated in stored cases. It offers advantages such as comprehensible knowledge representation, easier knowledge acquisition compared to rule-based approaches and accuracy improvement during system operation with insertion of new cases. However, cases lack the compactness of rules in representing domain knowledge.

Neurules, a type of hybrid rules integrating symbolic rules with neurocomputing have been combined with case-based reasoning. The combination of neurule-based with case-based reasoning results in accuracy improvement by employing different types of indices for the cases according to different roles they play in neurule-based reasoning (Prentzas et al., 2008a, 2008b). In this way, an improved knowledge representation scheme is derived as various types of gaps in neurules' representation of domain knowledge are filled in by indexed cases.

In this paper, we present an approach combining neurule-based with case-based reasoning for the assessment of life insurance applications. Neurules were produced from available symbolic rules elicited from an insurance expert. However, the available symbolic rules do not cover the full complexities of the domain and this is reflected to the produced neurules. The integration of neurules with available cases according to the approach presented in (Prentzas et al., 2008a, 2008b) improves the overall accuracy. The approach has a general interest from a knowledge representation viewpoint as it can be employed to improve the accuracy of rule-based expert systems given the availability of domain cases. The indexing construction process and the inference process combining neurule-based with case-based reasoning presented here are revised and extended versions of those presented in (Prentzas et al., 2008a) (for which authors retain the copyright) and (Prentzas et al., 2008b). We also present related work regarding application of AI methods to life insurance and approaches combining rule-based with case-based reasoning.

The rest of the paper is organized as follows. Section 2 discusses related work. Section 3 discusses issues involving the assessment of life insurance applications. Section 4 presents neurules and Section 5 presents the method for constructing an indexed case base combined with a neurule base. Section 6 describes the inference process combining neurule-based and case-based reasoning. Section 7 presents experimental results regarding accuracy of the inference process. Finally, Section 8 concludes.

2. Related Work

There are two main areas of related research. One concerns the application domain that is, assessment of life insurance applications. In this part of related work, we specify life insurance processes to which intelligent approaches have been applied focusing on those applied to assessment of life insurance applications. The other research direction refers to hybrid knowledge representation schemes that, like the one used in our approach, integrate rule-based and case-based reasoning.

Before discussing related work, we outline some basic notions concerning life insurance applications handling. For each approved application for life insurance, an insurance contract is signed. The terms ‘policies’ and ‘policyholders’ are also used for ‘contracts’ and ‘insured persons’ respectively. A life insurance contract involves an amount of money that will be given as benefit to the designated person. The insured person pays a premium. A benefit is provided upon the death of the insured person. A benefit may also be provided to the beneficiary when other specified events occur. For instance, an individual may also be insured for personal accidents that may cause permanent total disability, impermanent total disability or income losses due to inability to work. A life insurance company is interested in making profits from its portfolio of insurance contracts. This means that the insurance company should be able to: (a) cover the designated benefits (along with other potential payments) to the beneficiaries, (b) cover administrative costs and (c) make an overall profit.

2.1 Intelligent Approaches to Life Insurance

AI approaches have been applied to different (interrelating) aspects of life insurance processes enhancing decision making. Such aspects, among others, involve the following: (a) assessment of applications for life insurance and setting of policy parameters, (b) knowledge extraction from current policyholders’ data, (c) service quality, customer relationship management and servicing of policyholders, (d) marketing, (e) assessment of the solvency of insurance companies. Various AI approaches have been applied to life insurance, but the most popular ones are fuzzy methods, neural networks, rule-based expert systems as well as machine learning and data mining methods. Different life insurance processes impose different requirements. It seems that exploitation of various intelligent approaches is necessary in order to meet these requirements. We briefly outline main aspects involving life insurance processes as well as intelligent methods that have been applied recently. To the best of our knowledge, such a survey covering the different aspects involving life insurance and the intelligent methods that have been applied has not been presented till now in relevant literature.

Insurance companies have large amounts of stored data involving their customers that can be exploited to extract useful knowledge, as customers are a valuable asset to companies. Analyzing and extracting knowledge from current policyholders’ data is important according to various viewpoints such as increase of company profits, cost reduction, customer satisfaction, enhancement of service quality, identification of customer needs and desires, retention of customers that are likely to bring profits, identification of policyholders that are likely to surrender their contracts. Taking into consideration the high degree of competitiveness in the life insurance sector, the survival and success of life insurance companies depend on their relationships with customers. Extracted knowledge can be integrated with product and marketing knowledge (Liao et al., 2009). Knowledge extracted from customer data has been shown to be useful in customer relationship management, development of innovative and customer-oriented products, product promotion, market segmentation and exploration of new marketing possibilities. Machine learning and data mining methods have been exploited in order to extract knowledge from data involving policyholders. Knowledge incorporated in data involving policyholders may be also exploited for policyholder servicing. Table 1 outlines corresponding intelligent approaches and tasks performed.

Assessment of life insurance companies' solvency and provision of detailed information is an important process for analysts, regulators and shareholders (Baione et al., 2010). There are national and international regulators that define objectives, methodologies and prepare quantitative studies for insurance undertakings. The insurance business is risky and highly competitive. However, policyholders expect the agreed benefits in return for premiums and shareholders expect return on their investment. It is useful to provide early warnings when performance of companies deteriorates in order to handle problems in their infancy and avert crises. Regulators supervising life insurers, financial analysts, (existing and potential) policyholders and investors are interested in systems performing solvency monitoring and prediction of life insurance companies. For assessment of life insurance companies' solvency, AI approaches such as fuzzy methods (Baione et al., 2010) and neural networks (Shuang and Wei, 2011; Hsiao and Whang, 2009; Brockett et al. 2006) have been employed.

The handling and assessment of applications for life insurance are important processes for insurance companies. Companies need to assess the insurability of applicants with the assistance of relevant systems. Furthermore, in certain cases risk assessment may be used to set policy parameters. Knowledge extracted from current policyholders' data may be exploited. Health condition parameters may be examined for mortality forecasting and assessing risks of serious diseases (e.g. breast cancer, cardiovascular diseases) using intelligent methods such as neural networks (Shah and Guez, 2009) and fuzzy approaches (Tatari et al., 2012; Baser et al., 2011). In (Byczkowska-Lipińska et al., 2009) a rule-based expert system evaluates (medical and life) insurability taking into consideration the health condition, profession and hobbies. In (Hsieh and Wang 2011), fuzzy approaches have been used to perform risk assessment taking into consideration age and habits (i.e. smoking, drinking, sleeping, working, sport and driving habits). Intelligent methods have also been used to set life insurance policy parameters by assessing candidate policyholder characteristics and policy features. For instance in (Anzilli, 2012) a fuzzy approach is used for pricing investment-oriented life insurance policies taking into consideration demographic and financial data. In (Tatari et al., 2012), a fuzzy approach sets insurance premium according to breast cancer risk assessment. In (Baser et al., 2011) a neuro-fuzzy approach is used to set life insurance premiums based on cardiovascular risk assessment. In (Kumar and Pandley, 2012) a neural network evaluates the proper benefit taking into consideration socio-metric features (i.e. age, gender, income, profession, living area). In MetLife Inc. which receives thousands of applications that include many free-form text fields, an intelligent text analyzer is used to automate the process (Kantardzic, 2011). Table 2 outlines corresponding intelligent approaches and tasks performed.

It should be mentioned that compared to other approaches assessing applications for life insurance, our approach differs according to the following aspects:

- It combines integrated neuro-symbolic rules with case-based reasoning. From the recent published approaches, only the approach presented in (Başer et al. 2011) combines multiple intelligent schemes (i.e. it employs a neuro-fuzzy approach).
- It takes into consideration more parameters compared to most other recent approaches in order to perform the assessment. This is outlined in Section 3. It should be mentioned that certain approaches focus on specific risks (e.g. disease risks, mortality forecasting).
- Certain approaches also set policy features such as premiums (Tatari et al., 2012; Baser et al., 2011) and benefits (Kumar and Pandley, 2012). This task is done by the insurance personnel in our approach and therefore such aspects could be a future direction in our work.
- Our approach considers a set of parameters in assessing applications, but there are others, like living area (Kumar and Pandley, 2012), hobbies (Byczkowska-Lipińska et al., 2009) and habits (Hsieh and Wang 2011), which are not used.

2.2 Approaches that Integrate Rule-based and Case-based Reasoning

According to the categorization scheme analyzed in (Prentzas and Hatzilygeroudis, 2007), most integrated approaches follow a coupling model. In coupling approaches, different representation formalisms (i.e., rules or cases) are applied to the tasks composing the reasoning process. There are three main categories of coupling approaches: (i) sequential processing, (ii) co-processing and (iii) embedded processing approaches. In the following, we mainly cite recent representative approaches for each one of the coupling categories not cited in (Prentzas and Hatzilygeroudis, 2007).

The sequential processing category refers to coupling approaches in which the flow of information (produced by reasoning) between the integrated modules is sequential or semi-sequential (Cheung et al., 2011; Peng et al., 2011; Zhang et al., 2013; Lao et al., 2012; Schlüter and Conrad, 2012).

The co-processing category refers to approaches that are discerned into two types: cooperation-oriented and reconciliation-oriented. In the former type, the integrated components cooperate with each other (usually by interleaving their reasoning steps) for the production of the final result (Strobbe et al., 2012; Stevens et al., 2011; Noss et al., 2012). In the latter, each component produces its own conclusion, possibly differing from the conclusion of the other component, and thus a reconciliation process is necessary (Agre, 1995; Golding and Rosenbloom, 1996; Lee, 2002; Man et al., 2012; Iqbal et al., 2010).

In embedded processing approaches, a component based on one representation formalism embeds one or more components based on the other representation method to handle its internal reasoning tasks. Case-based reasoning systems often include one or more rule-based components to perform tasks such as case retrieval (Xiong, 2011; Tung et al., 2010) and adaptation (Zhou et al., 2011).

The approach presented in this paper can be classified as a reconciliation-oriented approach. Neurules however instead of simple rules are combined with cases. Neurules exploit advantages from both symbolic rules and neural networks. In contrast to other integrated approaches besides (Agre, 1995), the presented approach uses different types of indexed cases in the integrated inference process. Cases are indexed according to different roles they can play during inference and this improves accuracy. Approaches (Lee, 2002) and (Golding and Rosenbloom, 1996), which follow the reconciliation approach, store only (or place emphasis on) exception cases. Other approaches which follow the reconciliation approach such as (Man et al., 2012; Iqbal et al., 2010) and others discussed in (Prentzas and Hatzilygeroudis, 2007) invoke the case-based and rule-based components in parallel without indexing exception cases. Conflicts in latter approaches are resolved by weighing the rule-based and case-based outputs (Iqbal et al., 2010), by applying the Dempster Shafer method (Man et al., 2012), by averaging the rule-based and case-based outputs, by invoking a knowledge-based coordinator or through possibilistic reasoning (Prentzas and Hatzilygeroudis, 2007).

3. Assessing Applications for Life Insurance

The assessment process examines relevant data regarding the applicant to highlight potential risks in case the application is approved. This paper focuses on assessment of applications for life insurance covering personal accidents. The following attributes are taken into account in order to assess an insurance application:

- The *type of life insurance*. There are three basic types of insurance which additionally cover personal accidents:
    - *Temporary life insurance*. It provides insurance coverage for a specified term. It provides a benefit in case the insured person dies (or in case of an accident) during the designated term.
    - *Investment life insurance*. It combines life insurance with a funding program. The insured person selects the duration of insurance. Besides the benefit in case of death (or in case of an accident), this type of

    insurance also provides profits from the funding program when the insurance term ends.
  - *Permanent life insurance*. It is similar to the temporary life insurance with the exception that the duration of insurance is not restricted but is lifelong.
- *Gender*. There is a slight discrimination in the insurance of men and women especially in the calculation of premium. This is due to the fact that women tend to live longer than men.
- *Age*. Generally speaking, young persons have fewer health problems than elders and insurance risks are lower. There are also insurability regulations connecting specific insurance types and age limits. Individuals are discerned to very young, young, old and very old.
- *Primary life insurance benefit*. It is the benefit provided upon the death of the insured person. The corresponding system variable takes five discrete values: very low, low, average, high and very high.
- *Type of occupation*. The applicant's occupation affects the daily risks he/she is exposed to. Occupations are discerned into three main categories: (a) occupations with negligible or low risks, (b) occupations with moderate risks and (c) risky occupations. Individuals practicing occupations with negligible or low risks are considered safer to insure. The other types of occupations result into augmented premiums. Furthermore, according to the policy of insurance companies, individuals with risky occupations may not be insured at all. The system variable takes three discrete values corresponding to the three aforementioned categories: category-01, category-02 and category-03.
- *Annual income*. The annual income is discerned into low, low-to-average, average-to-high and high.
- *Personal accident benefit upper threshold*. A benefit may be provided in case of a personal accident. Insurance companies may impose an upper threshold to the provided benefit. The implemented system includes a 'yes/no' variable denoting whether the upper threshold of the personal accident benefit is exceeded or not.
- *Health condition*. The applicant's health condition is derived from medical examination results and can take three discrete values: bad, average and good. Different types of medical examinations are required according to applicant characteristics (e.g. age, selected insurance benefit).

Table 3 outlines the parameters considered by our approach and other AI methods used to handle and assess applications for life insurance.

4. Neurules

Symbolic rules (Ligeza, 2006; Hatzilygeroudis et al. 2006) and neural networks (Haykin, 2008) have been successfully applied to numerous domains. The objective of neuro-symbolic approaches is to combine (to a certain degree) advantages from both approaches. Neurules are a type of hybrid rules integrating symbolic rules with an adaline unit (Prentzas and Hatzilygeroudis, 2011). Neurules give pre-eminence to the symbolic component and their main advantages compared to other neuro-symbolic approaches, are modularity, naturalness (Hatzilygeroudis and Prentzas, 2000, 2001), more efficient inferences compared to symbolic rules (Hatzilygeroudis and Prentzas, 2000) and other hybrid approaches (Hatzilygeroudis and Prentzas, 2010) and provision of explanations for drawn conclusions (Hatzilygeroudis and Prentzas, 2015).

4.1 Syntax and Semantics

The syntax of a neurule is illustrated in Fig.1a. Each condition C_i is associated with a number sf_i, called its *significance factor* and each rule itself is also associated with a number sf_0, called its *bias factor*. The internal structure of a neurule is that of an adaline unit (Fig.1b). The *inputs* C_i (i=1,...,n) of the unit are the *conditions* of the rule. The weights of the unit are the significance factors of the neurule and its bias is the bias factor of the neurule. Inputs can take as values: 1 (true), -1 (false), 0 (unknown). So, neurules can clearly represent the falsity and the absence of a condition in contrast to symbolic rules. The *output D*, which represents the *conclusion* (decision) of the rule, is calculated via the standard formulas:

$$D = f(\mathbf{a}), \quad \mathbf{a} = sf_0 + \sum_{i=1}^{n} sf_i \ C_i$$

$$f(\mathbf{a}) = \begin{cases} 1 & if\ \mathbf{a} \geq 0 \\ -1 & otherwise \end{cases}$$

where **a** is the *activation value* and *f(x)* the *activation function*, a threshold function. Thus, output takes one of: ‘-1’, ‘1’, representing failure and success of the rule respectively.

The general syntax of a condition C_i and the conclusion D is:

<condition>::= <variable> <l-predicate> <value>
<conclusion>::= <variable> <r-predicate> <value>

where <variable> denotes a *variable*, that is a symbol representing a concept in the domain, e.g. ‘annual-income’, ‘age’ etc, in the life insurance domain. Variables are single-valued, that is, they may not take simultaneously multiple values during inference. <l-predicate> denotes a symbolic or a numeric predicate. The symbolic predicates are {is, isnot} and the numeric predicates are {<, >, =}. <r-predicate> can only be a symbolic predicate. <value> denotes a value. It can be a *symbol* or a *number*. The significance factor of a condition represents the significance (weight) of the condition in drawing the conclusion(s). Table 5 (Section 5) presents two example neurules, for assessing insurability in the life insurance domain. The conditions of the neurules are ordered according to the descending order of the absolute values of their significance factors. This facilitates inference.

Variables are designated as input, intermediate or output ones. An input variable takes values from the user (input data), whereas intermediate or output variables take values through inference since they represent intermediate and final conclusions respectively. We distinguish between intermediate and output neurules. An intermediate neurule is a neurule having at least one intermediate variable in its conditions and intermediate variables in its conclusion. An output neurule is one having an output variable in its conclusion. Although the neurule formalism supports intermediate variables, conclusions and neurules this may not be needed in every domain. In the specific application domain described in this paper there are no intermediate variables (see Section 4.2). Thus the application domain involves input and output variables, final conclusions and output neurules.

Neurules can be constructed either from symbolic rules, thus exploiting existing symbolic rule bases (Hatzilygeroudis and Prentzas, 2000), or from empirical data (i.e., training examples) (Hatzilygeroudis and Prentzas, 2001a). An adaline unit is initially assigned to each possible conclusion. Each unit is individually trained via the Least Mean Square (LMS) algorithm. When the training set is inseparable, more than one neurule having the same conclusion are produced. If neurules are constructed from available symbolic rules, each neurule usually merges two or more symbolic rules. In this way, the size of the rule base is reduced as far as the number of rules and conditions are concerned resulting in more efficient inferences compared to the symbolic source knowledge (Hatzilygeroudis and Prentzas, 2000).

The neurule-based inference engine gives pre-eminence to symbolic reasoning, based on a backward chaining strategy (Hatzilygeroudis and Prentzas, 2000). Conclusions are reached based on the values of the condition variables and the weighted sums of the conditions. A neurule fires if the output of the corresponding adaline unit is computed to be ‘1’ after evaluation of its conditions. A neurule is said to be ‘blocked’ if the output of the corresponding adaline unit is computed to be ‘-1’ after evaluation of its conditions. A condition evaluates to ‘true’ (‘1’), false (‘-1’) or unknown (‘0’) based on facts contained in the working memory. Given that variables may not take simultaneously multiple values during inference, when a condition evaluates to ‘true’, the conditions containing the same variable and different value may evaluate to ‘false’. For instance, in case variable ‘age’ takes the value ‘young’ then condition ‘age is young’ evaluates to ‘true’ whereas conditions ‘age is old’ and ‘age is very-old’ evaluate to false. Furthermore, the neurule semantics includes unknown inputs but in the assessment of life insurance applications this was not needed.

According to the two aforementioned formulas, evaluation of a neurule requires evaluation of all its conditions. However, a neurule may be evaluated without having evaluated all of its conditions speeding up inference. This is based on the values of two metrics of the neurule, the known sum and the remaining sum. The known sum represents the contribution of the (already) evaluated conditions of the neurule in drawing its conclusion which is expressed as the sum of the contributions of each condition. The known sum includes the bias factor of the neurule. The remaining sum represents the largest possible contribution of the unevaluated conditions of the neurule in drawing its conclusion, if they were evaluated. The largest possible contribution of each unevaluated condition is the absolute value of its significance factor. The known and remaining sums of a neurule are updated whenever a condition of the neurule is evaluated. A neurule may evaluate if the absolute value of the known sum exceeds the remaining sum. In this case, if the known sum is positive (negative) the neurule fires (is blocked). More details are presented in (Hatzilygeroudis and Prentzas 2010, 2015).

4.2 Construction of the System’s Neurule Base

The neurules used to assess insurability were constructed from available symbolic rules. The symbolic rules were elicited from a domain expert through interviews. The conclusions of the rules involve the output variable ‘insurability’ that takes two discrete values (i.e., yes, no). Each rule contains three to five conditions involving the input domain variables described in Section 3. In total, 22 symbolic rules were elicited. Nine of them have as conclusion ‘insurability is no’ and the rest ‘insurability is yes’.

The available symbolic rules were converted to neurules using the conversion mechanism presented in (Hatzilygeroudis and Prentzas, 2000). For the conversion, we used an expert system tool we have implemented (Hatzilygeroudis and Prentzas, 2001b). The conversion process grouped the symbolic rules into two initial merger sets. Each merger set contained all rules having the same conclusion (i.e. ‘insurability is no’ or ‘insurability is yes’). A merger was constructed for each merger set. A merger is a neurule having as conditions all the conditions of the symbolic rules in the corresponding merger set without duplications and significance factors as well as bias factor set to a proper initial value. For each merger, a training set was extracted from the truth table of the combined logical function of the rules in the set (the disjunction of the conjunctions of the conditions of each rule). Unacceptable training patterns are eliminated since certain conditions cannot be simultaneously true or false (Hatzilygeroudis and Prentzas 2000). Each merger was individually trained via the Least Mean Square (LMS) algorithm. If training is successful, a corresponding neurule is produced. Otherwise, the merger set was split into two merger subsets containing rules close to each other (Prentzas and Hatzilygeroudis, 2005) and the process is recursively executed for each subset.

Nine neurules were constructed from the available symbolic rules reducing the total number of rules and the total number of conditions in the rule base approximately by 59% and 40% respectively. Four neurules have as conclusion ‘insurability is no’ and the remaining five ‘insurability is yes’. Inferences from both rule bases produce the same output given the same input variable values. However, inferences from the neurule base are shorter in terms of the rules visited and the evaluated conditions.

We present an example of applying the conversion process. Let us suppose that we have the symbolic rules shown in Table 4. The rules are grouped into the following two initial merger sets corresponding to the conclusions ‘insurability is no’ and ‘insurability is yes’: MS_N={$rule_{1N}$, $rule_{2N}$, $rule_{3N}$, $rule_{4N}$} and MS_Y={$rule_{1Y}$, $rule_{2Y}$, $rule_{3Y}$, $rule_{4Y}$}.

The merger corresponding to MS_N contains the ten distinct conditions of the rules in the set. Its training set is extracted from the truth table of the combined logical function of the rules in the set: $F = (C_1 \wedge C_2 \wedge C_3) \vee (C_1 \wedge C_2 \wedge C_4 \wedge C_5 \wedge C_6) \vee (C_1 \wedge C_2 \wedge C_4 \wedge C_7 \wedge C_8 \wedge C_9 \wedge C_{10}) \vee (C_1 \wedge C_2 \wedge C_4 \wedge C_7 \wedge C_6)$, where C_1≡pers-acc-benefit-upper-threshold-exceeded is no, C_2≡health-condition is average, C_3≡annual-income is average-to-high, C_4≡annual-income is high, C_5≡age is old, C_6≡primary-life-insurance-benefit is very-high, C_7≡age is very-old, C_8≡primary-life-insurance-benefit is high, C_9≡insurance-type is investment and C_{10}≡gender is woman. Patterns that are eliminated as unacceptable are for instance, the ones in which the values corresponding to conditions C_5 and C_7 are simultaneously true because variable ‘age’ may not take simultaneously the values ‘old’ and ‘very-old’. Training of the merger is successful and neurule NR_1 shown in Table 5 is produced.

Similarly, the merger corresponding to MS_Y is constructed and its training set is extracted. Training of this merger is also successful and neurule NR_2 shown in Table 5 is produced.

5. Indexing Construction Process

The architecture of the integrated system consists of the following main modules: (a) the module converting a symbolic rule base to a neurule base, (b) the module assigning indices to cases, (c) the integrated inference mechanism and (d) the explanation mechanism.

To integrate neurule-based and case-based reasoning, indices need to be assigned to cases in order to retrieve relevant cases during inference. Indexed cases may assist neurule-based reasoning in avoiding reasoning errors by handling the following situations:

(a) Examining whether a neurule misfires. If sufficient conditions of the neurule are satisfied so that it can fire, it should be examined whether the neurule misfires for the specific facts, thus producing an incorrect conclusion.

(b) Examining whether a specific conclusion was incorrectly not drawn. A conclusion is not drawn when none of the neurules containing it fires. This happens when: (i) all neurules containing the conclusion have been examined and are blocked or/and (ii) a neurule containing an alternative conclusion for the specific variable fires instead. To make these aspects clearer, we give an example. For instance, if all neurules containing the conclusion ‘insurability is yes’ have been examined and are blocked, then this conclusion is not drawn. If a neurule containing e.g. the alternative conclusion ‘insurability is no’ fires, then conclusion ‘insurability is yes’ is not drawn.

To achieve this, different types of indices are assigned to cases according to the roles they play in neurule-based reasoning (Prentzas et al., 2008a, 2008b). Thus, cases assist in filling in different types of gaps in the knowledge representation by neurules. Assigning different types of indices to cases can produce an effective approach combining symbolic rule-based with case-based reasoning (Agre, 1995).

According to the approach presented in (Prentzas et al., 2008a, 2008b), indices are associated with neurules and neurule base conclusions. In particular, a case may be indexed as:

(a) *False positive (FP)*, by a neurule whose conclusion is contradicting. Such cases represent exceptions to neurules and may assist in handling neurule misfirings. A case constitutes an exception to a neurule if its attribute values satisfy sufficient conditions of the neurule (so that it can fire) but the neurule’s conclusion contradicts the corresponding attribute value of the case.

(b) *True positive (TP)*, by a neurule whose conclusion is endorsing. The attribute values of such a case satisfy sufficient conditions of the neurule (so that it can fire) and the neurule’s conclusion agrees with the corresponding attribute value of the case. Such cases may assist in endorsing correct neurule firings.

(c) *False negative (FN)*, by a conclusion incorrectly not drawn by neurules. Such cases may assist in reaching conclusions that ought to have been drawn by neurules (and were not drawn). If neurules with alternative conclusions containing this variable were fired instead, it may also assist in handling neurule misfirings. ‘False negative’ indices are associated with conclusions and not with specific neurules because there may be more than one neurule with the same conclusion in the neurule base.

In the approach presented in (Hatzilygeroudis and Prentzas, 2004a), indexed cases were used only to handle neurule misfirings. More specifically, the neurules indexed cases representing their exceptions. The approach presented in (Prentzas et al., 2008a, 2008b) results in accuracy improvement.

The indexing construction process takes as input available neurules and non-indexed cases. It assigns indices to cases by performing neurule-based reasoning for the neurules based on the attribute values of cases. To illustrate how the indexing process works, we present the following example. Suppose that we have a neurule base containing the two neurules in Table 5 and the example cases shown in Table 6 (only the most important attributes of the cases are shown). ‘insurability’ is the output attribute that corresponds to the neurules’ conclusion variable. Table 6 also shows the types of indices associated with each case at the end of the indexing construction process.

To acquire indexing information, the input values corresponding to the attribute values of the cases are presented to the example neurules. Recall that when a neurule condition evaluates to ‘true’ it gets the value ‘1’, whereas when it is false gets ‘-1’.

The known sum and remaining sum metrics are also used for the evaluation of neurules.

For example, given the input case Case_1, the conditions of NR_1 are gradually evaluated according to the input values corresponding to the attribute values of Case_1. The order of the evaluation of the conditions is according to the order shown in Table 5. The known and remaining sums are also gradually updated according to the contribution of the evaluated conditions of NR_1. Evaluation of the conditions of NR_1 stops when the absolute value of the known sum exceeds the value of the remaining sum. Given the input values corresponding to the attribute values of Case_1, the conditions of NR_1 will evaluate (if needed) to either ‘true’ or ‘false’. More specifically, the conditions that (if needed) will evaluate to ‘true’ are the following: ‘annual-income is average-to-high’, ‘health-condition is average’, ‘pers-acc-benefit-upper-threshold-exceeded is no’, ‘primary-life-insurance-benefit is very-high’ and ‘age is old’. The conditions that (if needed) will evaluate to ‘false’ are the following: ‘annual-income is high’, ‘age is very-old’, ‘primary-life-insurance-benefit is high’, ‘gender is woman’ and ‘insurance-type is investment’.

Table 7 depicts the values of the known and remaining sums after evaluation of each one of the first three conditions of NR_1. Initially the known sum is set to the bias factor and the remaining sum to the sum of the significance factors of all conditions. All of the first three conditions of NR_1 evaluate to ‘true’. The known sum after evaluation of the first three conditions is: $-49.60 + 37.5 + 37.4 + 37.30 = 62.6$. The remaining seven conditions of NR1 do not need to be evaluated since their largest possible contribution (i.e. remaining sum) is: $15.7 + 15.5 + 12.2 + 11.9 + 4.6 + 1.3 + 0.8 = 62 < 62.6$. The known sum after evaluation of the third condition of NR_1 is positive and greater than the remaining sum. This means that sufficient conditions of NR_1 are satisfied so that it can fire. Furthermore, the corresponding output attribute value of the case matches the conclusion of NR_1 and therefore Case_1 is associated as ‘true positive’ with NR_1.

Similarly, Case_4 and Case_5 are associated as ‘true positive’ with NR_1 and NR_2 respectively. Furthermore, when the input values corresponding to the attribute values of Case_2 are given as input to the neurule base, sufficient conditions of neurule NR_1 are satisfied so that it can fire. However, the corresponding output attribute case values contradict the conclusion of NR_1. Therefore Case_2 is associated as ‘false positive’ with NR_1. Case_2 is also associated as ‘false negative’ with conclusion ‘insurability is yes’. In addition, conclusion ‘insurability is yes’ cannot be drawn when the input values corresponding to the attribute values of Case_3 are given as input because the only neurule with the corresponding conclusion (i.e., NR_2) is blocked. Therefore, Case_3 is associated as ‘false negative’ with conclusion ‘insurability is yes’. Finally, conclusion ‘insurability is no’ cannot be drawn when the input values corresponding to the attribute values of Case_6 are given as input because the only neurule with the corresponding conclusion (i.e., NR_1) is blocked. Therefore, Case_6 is associated as ‘false negative’ with conclusion ‘insurability is no’. Note that both neurules NR_1 and NR_2 are blocked when the input values corresponding to the attribute values of either $Case_3$ or $Case_6$ are given as input.

The indexing process may take into consideration updates in the symbolic source knowledge of the neurule base. In (Prentzas and Hatzilygeroudis 2005) approaches to efficiently updating a neurule base are presented. The presented approaches update a neurule base when: (a) new symbolic rules are inserted in the symbolic rule base or (b) existing symbolic rules are removed from the symbolic rule base. Information concerning the splitting process is stored alongside the neurule base and is used in performing the updates. Due to the modularity of neurules, the updates require re-conversion of only the affected part of the neurule base leaving the rest intact. The update process for the neurule base is used to update case indices in the integrated knowledge base.

In cases where a new symbolic rule is inserted into the symbolic rule base, the symbolic rule is either merged with existing rules in a neurule or is not merged with other rules and is converted itself to a neurule. If the other neurules are not affected, then only one new neurule is constructed (i.e. the one merging the new symbolic rule). To determine the significance factors of the new neurules, the mergers corresponding to the new merger sets are constructed and trained via the LMS algorithm using the corresponding training set. The parts of the neurule base that is unaffected remains intact. All cases are presented to the new neurule merging the new symbolic rule to acquire additional indexing information. This information concerns insertion of ‘true positive’ indices with removal of corresponding ‘false negative’ indices (if any), insertion of potential new ‘false positive’ indices with insertion of corresponding ‘false negative’ indices. No further actions are required in cases where the symbolic rule is not merged with other rules and is converted itself to a neurule without affecting the other neurules. If the symbolic rule is merged with other rules in a neurule and the other neurules are not affected, the ‘true positive’ and ‘false positive’ indices corresponding to the previous version of the neurule are retained. Further actions are required if more than one new neurules are constructed. All cases indexed by the previous versions of the neurules produced by the corresponding merger subset are presented to the new neurules to reassign their ‘true positive’ and ‘false positive’ indices.

In cases where an existing symbolic rule is removed from the symbolic rule base, the ‘true positive’ and ‘false positive’ indices of the neurule that had merged the rule are examined. The ‘true positive’ indices of cases corresponding to the symbolic rule are removed and if they are not indexed as ‘true positive’ by other neurules they are replaced by ‘false negative’ indices. Potential ‘false positive’ indices of cases and corresponding ‘false negative’ indices are removed. If more than one neurules are affected by removing the rule, indices to all cases indexed by the previous versions of the neurules produced by corresponding merger subset are reassigned.

6. Inference Combining Neurule-based and Case-based Reasoning

The integrated inference process is primarily based on neurules. The indexed cases are considered when: (a) sufficient conditions of a neurule are fulfilled so that it can fire, (b) all neurules with a specific conclusion variable are blocked and thus no conclusion containing this variable is drawn.

In case (a), firing of the neurule is suspended and case-based reasoning is performed for cases indexed as ‘false positive’ and ‘true positive’ by the neurule and cases indexed as ‘false negative’ by all conclusions containing the neurule’s conclusion variable. Cases indexed as ‘true positive’ by the neurule and as ‘false negative’ by the neurule’s conclusion endorse its firing. If such a case is the best matching one, the neurule fires and its conclusion is inserted into the working memory. Cases indexed as ‘false positive’ by the neurule and as ‘false negative’ by alternative conclusions containing the neurule’s conclusion variable prevent its firing. Therefore, if such a case is the best matching one, the conclusion supported by the case is inserted into the working memory and the neurule is marked as ‘blocked’.

In case (b), the case-based module will focus on cases indexed as ‘false negative’ by conclusions containing the specific variable. The conclusion supported by the best matching case is inserted into the working memory.

The similarity measure between two cases c_k and c_l is calculated via a simple distance metric. The best-matching case to the problem at hand is the one having the maximum similarity with the input case.

Let us present now two simple inference examples concerning the combined neurule base (Table 5) and the indexed example cases (Tables 6 and 7). Suppose that during inference sufficient conditions of neurule NR_1 are satisfied so that it can fire. Firing of NR_1 is suspended and the case-based reasoning process focuses on the cases contained in the union of the following sets of indexed cases:

- the set of cases indexed as ‘true positive’ by NR_1: $\{Case_1, Case_4\}$,
- the set of cases indexed as ‘false positive’ by NR_1: $\{Case_2\}$ and
- the set of cases indexed as ‘false negative’ by the conclusions containing variable ‘insurability’: $\{Case_2, Case_3, Case_6\}$.

So, in this example the case-based reasoning process focuses on the following set of indexed cases: $\{Case_1, Case_4\} \cup \{Case_2\} \cup \{Case_2, Case_3, Case_6\} = \{Case_1, Case_2, Case_3, Case_4, Case_6\}$.

Suppose now that during inference both output neurules in the example neurule base are blocked. The case-based reasoning process will focus on the cases contained in the union set of the following sets of indexed cases:

- the set of cases indexed as ‘false negative’ by conclusion ‘insurability is no’: $\{Case_2, Case_3\}$.
- the set of cases indexed as ‘false negative’ by conclusion ‘insurability is yes’: $\{Case_6\}$.

Therefore, in this example the case-based reasoning process focuses on the following set of indexed cases: $\{Case_2, Case_3\} \cup \{Case_6\} = \{Case_2, Case_3, Case_6\}$.

Finally, explanations for reached conclusions are provided. Explanations for neurule-based reasoning are produced in the form of if-then explanation rules containing the conditions that were necessary for the evaluation of neurules (Hatzilygeroudis and Prentzas, 2015). The explanation mechanism also returns the most relevant retrieved case with an indication on its role in the reasoning process.

7. Experimental Results

In this section, we present experimental results regarding the approach combining neurule-based with case-based reasoning used for assessment of applications for life insurance. Experiments were performed to evaluate the presented approach combining neurule-based and case-based reasoning and compare it with our previous approach presented in (Hatzilygeroudis and Prentzas, 2004a).

The indexed case base was constructed from past cases acquired from the insurance company. In total, 269 cases corresponding to applications for life insurance that had been assessed in the past were acquired. The decision for insurability concerning 197 cases was positive and negative for the remaining 72 cases. Each case mainly consisted of values involving the following nine variables: health-condition, annual-income, age, primary life insurance benefit, insurance-type, gender, profession-category and personal-accident-benefit-upper-threshold-exceeded and insurability. Cases also consisted of further attribute values.

75% of the available cases were used as training and the remainder as testing sets. Each training set was used to create an indexed case base. The indexing construction process presented in this paper took as inputs the neurules produced from available symbolic rules (as described in Section 4.2) and non-indexed cases corresponding to the training set. The output was a combined neurule base and an indexed case base that will be referred to as NBRCBR. Neurules and non-indexed cases were also used to produce a combined neurule base and an indexed case base according to (Hatzilygeroudis and Prentzas, 2004a) which will be referred to as NBRCBR_PREV. On average, about 20 cases are indexed by a neurule as ‘true positive’ and ‘false positive’. The accuracy of pure neurule-based reasoning for the test set is 86.76%. More specifically, 86.76% of the test cases were classified correctly, 4.41% were ‘false positive’ and 8.83% resulted in having all output neurules blocked (i.e. ‘false negative’).

Inferences were run for both NBRCBR and NBRCBR_PREV using the testing sets as inputs. Inferences from NBRCBR_PREV were performed using the inference process combining neurule-based and case-based reasoning as described in (Hatzilygeroudis and Prentzas, 2004a). Inferences from NBRCBR were performed according to the inference process described here. No test case was stored in the case bases. Experimental results concerning inferences from NBRCBR and NBRCBR_PREV are presented in Tables 8-10.

Table 8 presents results involving classification accuracy of the integrated approaches and the percentage of test cases resulting in neurule-based reasoning errors that were successfully handled by case-based reasoning. Column '% FPs correctly handled' refers to the percentage of test cases resulting in neurule misfirings (i.e., 'false positive') that were successfully handled by case-based reasoning. Column '% FNs correctly handled' refers to the percentage of test cases resulting in having all output neurules blocked (i.e., 'false negative') that were successfully handled by case-based reasoning. 'False negative' test cases are handled in NBRCBR_PREV by retrieving the best-matching case from the library of indexed cases. As can be seen from Table 8, the presented approach results in improved classification accuracy. Moreover, in inferences from NBRCBR the percentages of both 'false positive' and 'false negative' test cases successfully handled are greater than the corresponding percentages in inferences from NBRCBR_PREV.

Cases indexed as 'false negative' are useful in handing successfully 'false positive' and 'false negative' test cases that could not be handled in NBRCBR_PREV. This results in improvement of the overall accuracy. Table 9 depicts the percentage of 'true positive', 'false positive' and 'false negative' test cases handled by each type of indexed cases. All three types of indexed cases are useful in successfully handling the three types of test cases. NBRCBR has an advantage over NBRCBR_PREV for the following reasons. First, in NBRCBR there are three different types of indices (i.e. 'false positive', 'true positive' and 'false negative') whereas in NBRCBR_PREV there are two types of indices (i.e. 'false positive' and 'true positive'). Both approaches assign 'false positive' and 'true positive' indices to the same sets of cases. However, in NBRCBR 'false positive' training cases are indexed as 'false positive' by corresponding neurules and as 'false negative' by corresponding conclusions. Second, the indexed case library in NBRCBR contains more training cases compared to the indexed case library in NBRCBR_PREV. The additional stored cases are the ones that result in having all output neurules blocked. These cases are indexed as 'false negative' by the corresponding conclusions in NBRCBR. As shown in Table 9, cases indexed as 'false negative' in NBRCBR are useful in handling successfully more 'false positive' and 'false negative' test cases compared to NBRCBR_PREV.

Table 10 depicts the average number of indexed cases for NBRCBR and NBRCBR_PREV that are considered when sufficient conditions of a neurule are fulfilled so that it can fire and when all output neurules are blocked and no conclusion concerning insurability can be drawn. As shown in the table, more indexed cases are considered to examine whether a neurule misfires or not in NBRCBR compared to NBRCBR_PREV. Cases indexed as 'false negative' by the two alternative conclusions are also considered in NBRCBR besides cases indexed as 'true positive' and 'false positive' by a neurule and misfirings are handled better. Furthermore, less indexed cases are considered when all output neurules are blocked in NBRCBR compared to NBRCBR_PREV. In NBRCBR_PREV, all indexed cases (i.e. 'true positive’ and ‘false positive’) are retrieved. In NBRCBR only cases indexed as ‘false negative’ are retrieved and this enables the inference mechanism to focus on more relevant situations resulting in having all output neurules blocked.

A nearest neighbor approach working alone in the insurance dataset was also tested. The same training and testing sets as well as the same similarity measure were used in the two integrated approaches. The approach classified the input case to the conclusion supported by the best-matching case retrieved from the case base. Classification accuracy was 88.23%. Therefore, the two integrated approaches outperform the nearest neighbor approach since indices facilitate the reasoning process to focus on specific parts of the case base.

8. Conclusions

In this paper, we present an approach used to assess applications for life insurance. The approach integrates general domain knowledge with empirical knowledge relevant to life insurance. More specifically, the approach integrates neurule-based and case-based reasoning. Neurules are a type of hybrid rules integrating symbolic rules with neurocomputing. In contrast to other neuro-symbolic approaches, neurules retain the naturalness and modularity of symbolic rules. Neurules were produced from available symbolic rules assessing insurability. Integration of neurules and cases is done in order to improve the accuracy of the inference mechanism. Cases are indexed according to the roles they can play during neurule-based inference. More specifically, they are associated as ‘true positive’ and ‘false positive’ with neurules and as ‘false negative’ with neurule base conclusions. We outline life insurance processes to which intelligent approaches can be employed and we refer to some such approaches. We also outline recent work involving integration of rule-based and case-based reasoning.

The integrated approach has improved accuracy compared to pure rule-based reasoning, our previous approach combining neurule-based and case-based reasoning and a nearest neighbor approach. This is due to the different types of indices assigned to cases. Cases that are more relevant to the reasoning situation at hand are retrieved handling misfirings and inability to draw conclusions.

An innovation of our approach is the application of a hybrid intelligent system to a new setting. Very few hybrid intelligent systems have been applied to the life insurance domain and specifically to application assessment. Hybrid intelligent systems could be applied in other financial domains as well assisting in improving different types of financial processes. Another financial domain to which hybrid approaches could be employed involves credit scoring. For example, neurules have been employed in (Hatzilygeroudis and Prentzas, 2011) to assess bank loan applicants.

In financial domains, empirical cases are usually available and also general domain knowledge can be acquired. Therefore, it is likely that more approaches integrating rule-based and case-based reasoning will be developed in financial domains.

It is also very likely that further AI approaches to life insurance processes will be employed. This is due to the fact that there is a high degree of competitiveness in the insurance sector creating the need to improve decision making. Life insurance is a financial sector that is thriving in several countries around the world. In a recent survey (Muñoz-Leiva et al., 2012), it is mentioned that innovation related to life insurance will be developed in the coming years. According to this survey, it is anticipated that further research regarding life insurance will be conducted the following years.

An aspect of our future research work involves improving the functionality of the assessment system towards four main directions. First, the system could be extended by incorporating additional parameters in the representation of application assessment. Such parameters could be the pollution and delinquency levels of the living area, habits affecting health condition (e.g. smoking) and hobbies that could impose risks (e.g. sport hobbies). Second, the system could propose the most appropriate insurance policy for approved applicants. Third, the system could be extended to set policy parameters such as premium and benefit. Fourth, an intelligent approach to assist doctors in the assessment of the applicants' health condition may be developed. Currently, in our approach health condition assessment is performed exclusively by doctors.

Another aspect of our future work involves implementation of an Intelligent Tutoring System based on the integration of neurules and cases. Such an integrated approach satisfies several of the knowledge representation requirements of Intelligent Tutoring Systems (Hatzilygeroudis and Prentzas, 2006). More specifically, we intend to extend our previous work on Intelligent Tutoring Systems (Prentzas, Hatzilygeroudis and Koutsojannis, 2001) to early childhood. Few e-learning systems incorporating AI methods and addressed to early childhood have been developed (Prentzas, 2013).

References

AGRE, G. (1995) KBS maintenance as learning two-tiered domain representation, in M.M. VELOSO and A. AAMODT (Eds.), Case-based Reasoning Research and Development, Lecture Notes in Computer Science, Berlin/Heidelberg, Germany: Springer-Verlag, Vol. 1010, 108–120.

ANZILLI, L. (2012) A possibilistic approach to evaluating equity-linked life insurance policies, in S. GRECO et al. (Eds.), IPMU 2012, Part IV, Communications in Computer and Information Science, Berlin/Heidelberg, Germany: Springer-Verlag, Vol. 300, 44–53.

BAE, S.M., S.H. HA and S.C. PARK (2005) A web-based system for analyzing the voices of call center customers, Expert Systems with Applications, 28, 29–41.

BAIONE, F., P. DE ANGELIS and R. OTTAVIANI (2010) On a decision model for a life insurance company rating, in S. GRECO et al. (Eds.), Preferences and

Decisions, Studies in Fuzziness, Berlin/Heidelberg, Germany: Springer-Verlag, Vol. 257, 123–138.

BASER, F., T.E. DALKILIÇ, K.S. KULA and A. APAYDIN (2011) An approach to adaptive network based fuzzy inference system to risk classification in life insurance, in B. HU et al. (Eds.), Operations Research Proceedings, Berlin/Heidelberg, Germany: Springer-Verlag, 27–32.

BELCIUG, S. and F. GORUNESCU (2013) A hybrid neural network/genetic algorithm applied to breast cancer detection and recurrence, Expert Systems, 30, 243–254.

BROCKETT, P.L., L.L. GOLDEN, J. JANG and C. YANG (2006) A comparison of neural network, statistical methods, and variable choice for life insurers' financial distress prediction, The Journal of Risk and Insurance, 73, 397–419.

BYCZKOWSKA-LIPIŃSKA, L., M. SZYDŁO and P. LIPIŃSKI (2009) Expert systems in the medical insurance industry, in E. KACKI, M. RUDNICKI and J. STEMPCZYNSKA (Eds.), Computers in Medical Activity, Advances in Intelligent and Soft Computing, Berlin/Heidelberg, Germany: Springer-Verlag, Vol. 65, 189–199.

CHATTOPADHYAY, S. (2014) Neurofuzzy models to automate the grading of old-age depression, Expert Systems, 31, 48–55.

CHEUNG, C.F., W.M. WANG and Z.C.S. LEUNG (2011) A pilot study on a knowledge-based case library to support suicide risk assessment, International Social Work, 56, 208–227.

D'AVILA GARCEZ, A.S. and L.C. LAMB (2011) Cognitive algorithms and systems: reasoning and knowledge representation, in V. CUTSURIDIS, A. HUSSAIN and J.G. TAYLOR (Eds.), Perception-action Cycle: Models, Architectures, and Hardware, Springer Series in Cognitive and Neural Systems, Berlin/Heidelberg, Germany: Springer-Verlag, 573–600.

GOLDING, A.R. and P.S. ROSENBLOOM (1996) Improving accuracy by combining rule-based and case-based reasoning, Artificial Intelligence, 87, 215–254.

HATZILYGEROUDIS, I. and J. PRENTZAS (2000) Neurules: improving the performance of symbolic rules, International Journal on AI Tools, 9, 113–130.

HATZILYGEROUDIS, I. and J. PRENTZAS (2001a) Constructing modular hybrid rule bases for expert systems, International Journal on AI Tools, 10, 87–105.

HATZILYGEROUDIS, I. and J. PRENTZAS (2001b) HYMES: a HYbrid Modular Expert System with efficient inference and explanation, in Proceedings of the 8th Panhellenic Conference on Informatics, Vol.1, 422–431.

HATZILYGEROUDIS, I. and J. PRENTZAS (2004a) Integrating (rules, neural networks) and cases for knowledge representation and reasoning in expert systems, Expert Systems with Applications, 27, 63–75.

HATZILYGEROUDIS, I. and J. PRENTZAS (2004b) Neuro-symbolic approaches for knowledge representation in expert systems, International Journal on Hybrid Intelligent Systems, 1, 111–126.

HATZILYGEROUDIS, I. and J. PRENTZAS (2006) Knowledge representation in intelligent educational systems, in Z. MA (Ed.), Web-based Intelligent e-Learning Systems: Technologies and Applications, Hershey, PA: Idea Group, 175–192.

HATZILYGEROUDIS, I., C. KOUTSOJANNIS, C. PAPAVLASOPOULOS, J. PRENTZAS (2006) Knowledge-based adaptive assessment in a Web-based intelligent educational system, in, Proceedings of IEEE International Conference on, Advanced Learning Technologies, New York: IEEE, 651–655.

HATZILYGEROUDIS, I. and J. PRENTZAS (2010) Integrated rule-based learning and inference, IEEE Transactions on Knowledge and Data Engineering, 22, 1549–1562.

HATZILYGEROUDIS, I. and J. PRENTZAS (2011) Combinations of Intelligent Methods and Applications, Smart Innovation, Systems and Technologies, Vol. 8. Berlin/Heidelberg, Germany: Springer-Verlag.

HATZILYGEROUDIS, I. and J. PRENTZAS (2011) Fuzzy and neuro-symbolic approaches to assessment of bank loan applicants. In L.S. ILIADIS, I. MAGLOGIANNIS and H. PAPADOPOULOS (Eds.): Artificial Intelligence Applications and Innovations, IFIP Advances in Information and Communication Technology, Berlin/Heidelberg, Germany: Springer-Verlag, Part II, 82–91.

HATZILYGEROUDIS, I. and J. PRENTZAS (2015) Symbolic-neural rule based reasoning and explanation, Expert Systems with Applications, 42, 4595–4609.

HAYKIN, S.O. (2008) Neural Networks and Learning Machines, Third edition, Upper Saddle River, New Jersey: Prentice Hall.

HSIAO, S.-H. and T.-J. WHANG (2009) A study of financial insolvency prediction model for life insurers, Expert Systems with Applications, 36, 6100–6107.

HSIEH, C.H. and P.P. WANG (2011) Linguistic evaluation system and insurance, New Mathematics and Natural Computation, 7, 383–411.

HSU, K.-T. (2011) Using a back propagation network combined with grey clustering to forecast policyholder decision to purchase investment-linked insurance, Expert Systems with Applications, 38, 6736–6747.

IQBAL, A.M., RAHMAN, M., KARIM, R. and M. AL-KHATARI (2010) A hybrid intelligent decision support system for swine flu, in Proceedings of the IADIS International Conference e-Health 2010, Part of the IADIS Multi Conference on Computer Science and Information Systems, 234–238.

KANTARDZIC, M. (2011) Data Mining: Concepts, Models, Methods and Algorithms, Second edition, Hoboken, New Jersey: John Wiley & Sons.

KUMAR, K. and D.K. PANDLEY (2012) ANN model for effects of socio-metric scales on life insurance, International Journal of Computer Science & Information Technology, 2, 446–453.

KUMAR, P. and D. SINGH (2011) Integrating data mining and AHP for life insurance product recommendation, in V.V. DAS and N. THANKACHAN (Eds.), CIIT 2011, Communications in Computer and Information Science, Berlin/Heidelberg, Germany: Springer-Verlag, Vol. 250, 596–602.

LAO, S.I., CHOY, K.L., HO, G.T.S., YAM, R.C.M., TSIM, Y.C. and T.C. POON (2012) Achieving quality assurance functionality in the food industry using a hybrid case-based reasoning and fuzzy logic approach, Expert Systems with Applications, 39, 5251–5261.

LEE, M.R. (2002) An exception handling of rule-based reasoning using case-based reasoning, Journal of Intelligent and Robotic Systems, 35, 327–338.

LIAO, S.-H., CHEN, Y.-N. and Y.-Y. TSENG (2009) Mining demand chain knowledge of life insurance market for new product development, Expert Systems with Applications, 36, 9422–9437.

LIGEZA, A. (2006) Logical Foundations for Rule-based Systems, Studies in Computational Intelligence, Vol. 11. Berlin/Heidelberg, Germany: Springer-Verlag.

LIN, W.-B. (2010) Service failure and consumer switching behaviors: evidence from the insurance industry, Expert Systems with Applications, 37, 3209–3218.

LIN, X., J. SUN, V. PALADE, W. FANG, X. WU and W. XU (2012) Training ANFIS parameters with a quantum-behaved particle swarm optimization algorithm, in Lecture Notes in Computer Science, Berlin/Heidelberg, Germany: Springer-Verlag, Vol. 7331, 148–155.

MAN, X., HAIYAN, Y. and S. JIANG (2012) A new algorithm for CBR-RBR fusion with robust thresholds, Chinese Journal of Mechanical Engineering, 25, 1255–1263.

MILHAUD, X., LOISEL, S. and V. MAUME-DESCHAMPS (2011) Surrender triggers in life insurance: classification and risk predictions, Bulletin Français d’ Actuariat, 11, 5–48.

MUÑOZ-LEIVA, F., J. SÁNCHEZ-FERNÁNDEZ, F.J. LIÉBANA-CABANILLAS and A.G. LÓPEZ-HERRERA (2012) Applying an automatic approach for showing up the hidden themes in financial marketing research (1961–2010), Expert Systems with Applications, 39, 11055–11065.

NOSS, R., A. POULOVASSILIS, E. GERANIOU, S. GUTIERREZ-SANTOS, C. HOYLES, K. KAHN, G.D. MAGOULAS and M. MAVRIKIS (2012) The design of a system to support exploratory learning of algebraic generalization, Computers & Education, 59, 63–81.

PENG, G., G. CHEN, C. WU, H. XIN and Y. JIANG (2011) Applying RBR and CBR to develop a VR based integrated system for machining fixture design, Expert Systems with Applications, 38, 26–38.

PRENTZAS, J., I. HATZILYGEROUDIS and C. KOUTSOJANNIS (2001) A Web-based ITS controlled by a hybrid expert system, in Proceedings of IEEE International Conference on Advanced Learning Technologies, New York: IEEE Press, 239–240.

PRENTZAS, J. and I. HATZILYGEROUDIS (2005) Rule-based update methods for a hybrid rule base, Data and Knowledge Engineering, 55, 103–128.

PRENTZAS, J. and I. HATZILYGEROUDIS (2007) Categorizing approaches combining rule-based and case-based reasoning, Expert Systems, 24, 97–122.

PRENTZAS, J., HATZILYGEROUDIS, I. and O. MICHAIL (2008a) Improving the accuracy of neuro-symbolic rules with case-based reasoning, in Proceedings of the 1st International Workshop on Combinations of Intelligent Methods and Applications, ECAI-08, 49–54.

PRENTZAS, J., HATZILYGEROUDIS, I. and O. MICHAIL (2008b) Improving the integration of neuro-symbolic rules with case-based reasoning, in J. DARZENTAS et al. (Eds.), Artificial Intelligence: Theories, Models and Applications, Lecture Notes in Artificial Intelligence, Berlin/Heidelberg, Germany: Springer-Verlag, Vol. 5138, 377–382.

PRENTZAS, J. and I. HATZILYGEROUDIS (2009) Combinations of case-based reasoning with other intelligent methods, International Journal of Hybrid Intelligent Systems, 6, 189–209.

PRENTZAS, J. and I. HATZILYGEROUDIS (2011) Neurules – A type of neuro-symbolic rules: an overview. In I. Hatzilygeroudis and J. Prentzas, (Eds.), Combinations of Intelligent Methods and Applications, Smart Innovation, Systems and Technologies, Berlin/Heidelberg, Germany: Springer-Verlag, Vol. 8, 145–165.

PRENTZAS, J. (2013). Artificial intelligence methods in early childhood education, in X.-S. Yang (Ed.), Artificial intelligence, evolutionary computation and metaheuristics – in the footsteps of Alan Turing, Studies in computational intelligence, Berlin/Heidelberg: Springer-Verlag, Vol. 427, 169–199.

SAHIN, S., M.R. TOLUN and R. HASSANPOUR (2012) Hybrid expert systems: a survey of current approaches and applications, Expert Systems with Applications, 39, 4609–4617.

SCHLÜTER, T. and S. CONRAD (2012) An approach for automatic sleep stage scoring and apnea-hypopnea detection, Frontiers of Computer Science, 6, 230–241.

SHAH, P. and A. GUEZ (2009) Mortality forecasting using neural networks and an application to cause-specific data for insurance purposes, Journal of Forecasting, 28, 535–548.

SHUANG, P. and L. WEI (2011) The early warning of life insurance company based on BP artificial neural network, in Y. WU (Ed.), ICCIC 2011, Part III, Communications in Computer and Information Science, Berlin/Heidelberg, Germany: Springer-Verlag, Vol. 233, 30–38.

STEVENS, C., V. BAROT and J. CARTER (2011) The next generation of legal expert systems – new dawn or false dawn?, in M. BRAMER et al. (Eds.), Research and Development in Intelligent Systems XXVII, London, UK: Springer-Verlag, 439–452.

STROBBE, M., O. VAN LAERE, B. DHOEDT, F. DE TURCK and P. DEMEESTER (2012), Hybrid reasoning technique for improving context-aware applications, Knowledge and Information Systems, 31, 581–616.

TATARI, F., M.-R. AKBARZADEH-T and A. SABAHI (2012) Fuzzy-probabilistic multi agent system for breast cancer risk assessment and insurance premium assignment, Journal of Biomedical Informatics, 45, 1021–1034.

TUNG, Y.-H., S.-S. TSENG, J.-F. WENG, T.-P. LEE, A.Y.H. LIAO and W.-N. TSAI (2010) A rule-based CBR approach for expert finding and problem diagnosis, Expert Systems with Applications, 37, 2427–2438.

WU, K.-S., F.-K. WANG and J.-Y. SHYNG (2013) Applied data mining techniques in insurance company: a comparative study of rough sets and decision tree, African Journal of Business Management, 7, 2309–2318.

XIONG, N. (2011) Learning fuzzy rules for similarity assessment in case-based reasoning, Expert Systems with Applications, 38, 10780–10786.

ZHANG, X.H., Z.H. DENG, W.K. AN and H. CAO (2013) A methodology for contour error intelligent precompensation in cam grinding, The International Journal of Advanced Manufacturing Technology, 64, 165–170.

ZHOU, Y., Y. LI and W. WANG (2011) A feature-based fixture design methodology for the manufacturing of aircraft structural parts, Robotics and Computer-Integrated Manufacturing, 27, 986–993.

Table 1. AI methods and life insurance general tasks handled

AI Methods	Tasks handled
Decision trees and Self-Organizing Maps (Bae et al., 2005)	Analysis of customer voices in call centers.
Decision trees and rough sets (Wu et al., 2013)	Rules, factors affecting Customer Lifetime Value (CLV), CLV calculation.
Association rules and k-means clustering (Liao et al., 2009)	New product development, market segmentation, demand chain, analysis.
k-means clustering (Kumar and Singh, 2011)	Product recommendation.
Back propagation neural networks (Hsu, 2011)	Identification of policyholders willing to obtain an investment-linked insurance.
Fuzzy neural networks (Lin, 2010)	Analysis of policyholders' switching behaviors.
Logistic regression and Classification and Regression Trees (Milhaud et al., 2011)	Features useful for identification of policyholders likely to surrender contracts.
Combination of rule-based and case-based reasoning (Lee, 2002)	Determines whether a claim amount should be granted to a policyholder based on a specific cause.

Table 2. AI methods and life insurance assessment tasks handled

AI Methods	Tasks handled
Back propagation neural networks (Shah and Guez, 2009)	Mortality forecasting.
Fuzzy sets approach (Tatari et al., 2012)	Breast cancer risk assessment and insurance premium setting.
Fuzzy rules approach (Hsieh and Wang 2011)	Risk assessment taking into consideration age and habits (i.e. smoking, drinking, sleeping, working, sport and driving habits).
Neuro-fuzzy approach (Başer et al. 2011)	Cardiovascular risk assessment and insurance premiums setting.
Rule-based expert system (Byczkowska-Lipińska et al., 2009)	Insurability assessment taking into consideration the health condition, age, profession and hobbies.
Back propagation neural networks (Kumar and Pandley, 2012)	Sets the benefit taking into consideration socio-metric features (i.e. age, gender, income, profession, living area).
Intelligent text analyzer in MetLife Inc. (Kantardzic, 2011)	Automates processing of applications that include many free-form text fields.

Table 3. AI methods and parameters for life insurance application assessment

AI Methods	Parameters considered
Back propagation neural networks for mortality forecasting (Shah and Guez, 2009)	health condition, age, gender
Fuzzy sets approach (Tatari et al., 2012) for breast cancer risk assessment and insurance premium setting	health condition, age, gender, race, diet, physical exercises, drinking habits, etc.
Fuzzy rules approach (Hsieh and Wang, 2011) for risk assessment	age and habits (i.e. smoke, drink, sleep, work, sport and drive)
Neuro-fuzzy approach (Baser et al., 2011) for cardiovascular risk assessment and insurance premiums setting	health condition (systolic blood pressure, cholesterol level, obesity), average cigarette consumption per day
Rule-based expert system (Byczkowska-Lipińska et al., 2009) for insurability assessment	health condition, age, profession and hobbies
Back propagation neural networks (Kumar and Pandley, 2012) for benefit setting	age, gender, monthly income, profession, living area (i.e. urban, rural)
Our approach for insurability assessment	health condition, age, gender, annual income, profession, insurance type, primary life insurance benefit

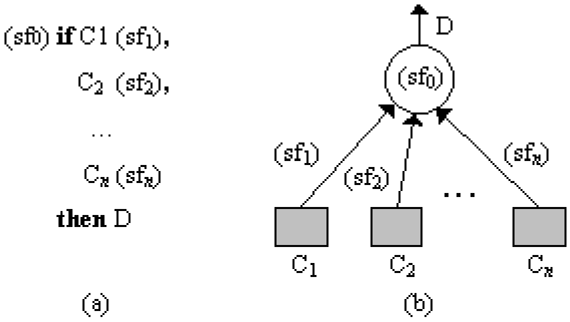

Fig. 1. (a) Form of a neurule (b) a neurule as an adaline unit

Table 4. Example symbolic rules for life insurance assessment

rule$_{1N}$ **if** pers-acc-benefit-upper-threshold-exceeded is no, health-condition is average, annual-income is average-to-high **then** insurability is no	rule$_{1Y}$ **if** pers-acc-benefit-upper-threshold-exceeded is no, health-condition is average, annual-income is high, age is young **then** insurability is yes
rule$_{2N}$ **if** pers-acc-benefit-upper-threshold-exceeded is no, health-condition is average, annual-income is high, age is old, primary-life-insurance-benefit is very-high **then** insurability is no	rule$_{2Y}$ **if** pers-acc-benefit-upper-threshold-exceeded is no, health-condition is average, annual-income is high, age is old, primary-life-insurance-benefit is high **then** insurability is yes
rule$_{3N}$ **if** pers-acc-benefit-upper-threshold-exceeded is no, health-condition is average, annual-income is high, age is very-old, primary-life-insurance-benefit is high, insurance-type is investment, gender is woman **then** insurability is no	rule$_{3Y}$ **if** pers-acc-benefit-upper-threshold-exceeded is no, health-condition is average, annual-income is high, age is very-old, primary-life-insurance-benefit is high, insurance-type is temporary **then** insurability is yes
rule$_{4N}$ **if** pers-acc-benefit-upper-threshold-exceeded is no, health-condition is average, annual-income is high, age is very-old, primary-life-insurance-benefit is very-high **then** insurability is no	rule$_{4Y}$ **if** pers-acc-benefit-upper-threshold-exceeded is no, health-condition is medium, annual-income is high, age is very-old, primary-life-insurance-benefit is high, insurance-type is investment, gender is man **then** insurability is yes

Table 5. Example neurules for life insurance assessment

NR_1
(-49.6) **if** annual-income is average-to-high (37.5),
health-condition is average (37.4),
pers-acc-benefit-upper-threshold-exceeded is no (37.3),
annual-income is high (15.7),
age is very-old (15.5),
primary-life-insurance-benefit is very-high (12.2),
age is old (11.9),
primary-life-insurance-benefit is high (4.6),
gender is woman (1.3),
insurance-type is investment (0.8)
then insurability is no

NR_2
(-65.4) **if** pers-acc-benefit-upper-threshold-exceeded is no (35.8),
age is young (35.7),
annual-income is high (32.2),
health-condition is average (31.5),
age is old (20.9),
primary-life-insurance-benefit is high (14.1),
age is very-old (14.1),
insurance-type is temporary (6.3),
gender is man (3.1),
insurance-type is investment (2.9)
then insurability is yes

Table 6. Example cases for life insurance assessment and indices assigned to them

		Case$_1$	***Case$_2$***	***Case$_3$***	***Case$_4$***	***Case$_5$***	***Case$_6$***
Case Attributes	***annual-income***	average-to-high	average-to-high	high	average-to-high	high	high
	health-condition	average	average	good	average	average	average
	pers-acc-benefit*-upper-threshold-exceeded*	no	no	no	no	no	no
	age	old	young	very-old	very-old	old	very-old
	primary-life-insurance-benefit	very-high	very-high	high	very-high	high	high
	profession-category	category-01	category-01	category-01	category-02	category-01	category-02
	gender	man	man	man	man	woman	man
	insurance-type	temporary	investment	investment	investment	temporary	permanent
	insurability	no	yes	yes	no	yes	no
Neurules & Case Indices	***Neurule NR$_1$ (concludes: 'insurability is no')***	Index: 'True positive'	Index: 'False positive'	Index: -	Index: 'True positive'	Index: -	Index: -
	Neurule NR$_2$ (concludes: 'insurability is yes')	Index: -	Index: -	Index: -	Index: -	Index: 'True positive'	Index: -
Conclusions & Case Indices	***Conclusion 'insurability is no'***	Index: -	Index: -	Index: -	Index: -	Index: -	Index: 'False negative'
	Conclusion 'insurability is yes'	Index: -	Index: 'False negative'	Index: 'False negative'	Index: -	Index: -	Index: -

Table 7. Evaluation of NR1 given the input values corresponding to attribute values of Case$_1$

Evaluated Condition	Evaluated Condition Value	Significance Factor	Known Sum	Remaining Sum
-	-	-49.6 (bias factor)	-49.6	174.2
annual-income is average-to-high	‘true’	37.5	-49.6 + 37.5 = -12.1	174.2 – 37.5 = 136.7
health-condition is average	‘true’	37.4	-12.1 + 37.4 = 25.3	136.7 – 37.4 = 99.3
pers-acc-benefit-upper-threshold-exceeded is no	‘true’	37.3	25.3 + 37.3 = 62.6	99.3 – 37.3 = 62

Table 8. Experimental results involving accuracy

	NBRCBR			NBRCBR_PREV		
Dataset	**Classification Accuracy**	**% FPs Successfully Handled**	**% FNs Successfully Handled**	**Classification Accuracy**	**% FPs Successfully Handled**	**% FNs Successfully Handled**
Insurance dataset (269 patterns)	95.59 %	66.66%	66.66%	92.65%	33.33%	50,00%

Table 9. Experimental results showing how test cases are handled by indexed cases

Test cases	NBRCBR	NBRCBR_PREV
TPs successfully handled by indexed TPs	91.52%	100%
TPs successfully handled by indexed FNs	8.48%	-
FPs Unsuccessfully handled by indexed TPs	33.33%	66.66%
FPs successfully handled by indexed FPs	33.33%	33.33%
FPs successfully handled by indexed FNs	33.33%	-
FNs successfully handled by indexed FNs	66.66%	-
FNs Unsuccessfully handled by indexed FNs	33.33%	-
FNs successfully handled by indexed TPs	-	50%
FNs Unsuccessfully handled by indexed TPs	-	50%

Table 10. Average number of cases considered in reasoning by NBRCBR and NBRCBR_PREV

Reasoning situation	NBRCBR	NBRCBR_PREV
Sufficient conditions of a neurule are fulfilled to fire	46	20
All output neurules are blocked	27	179