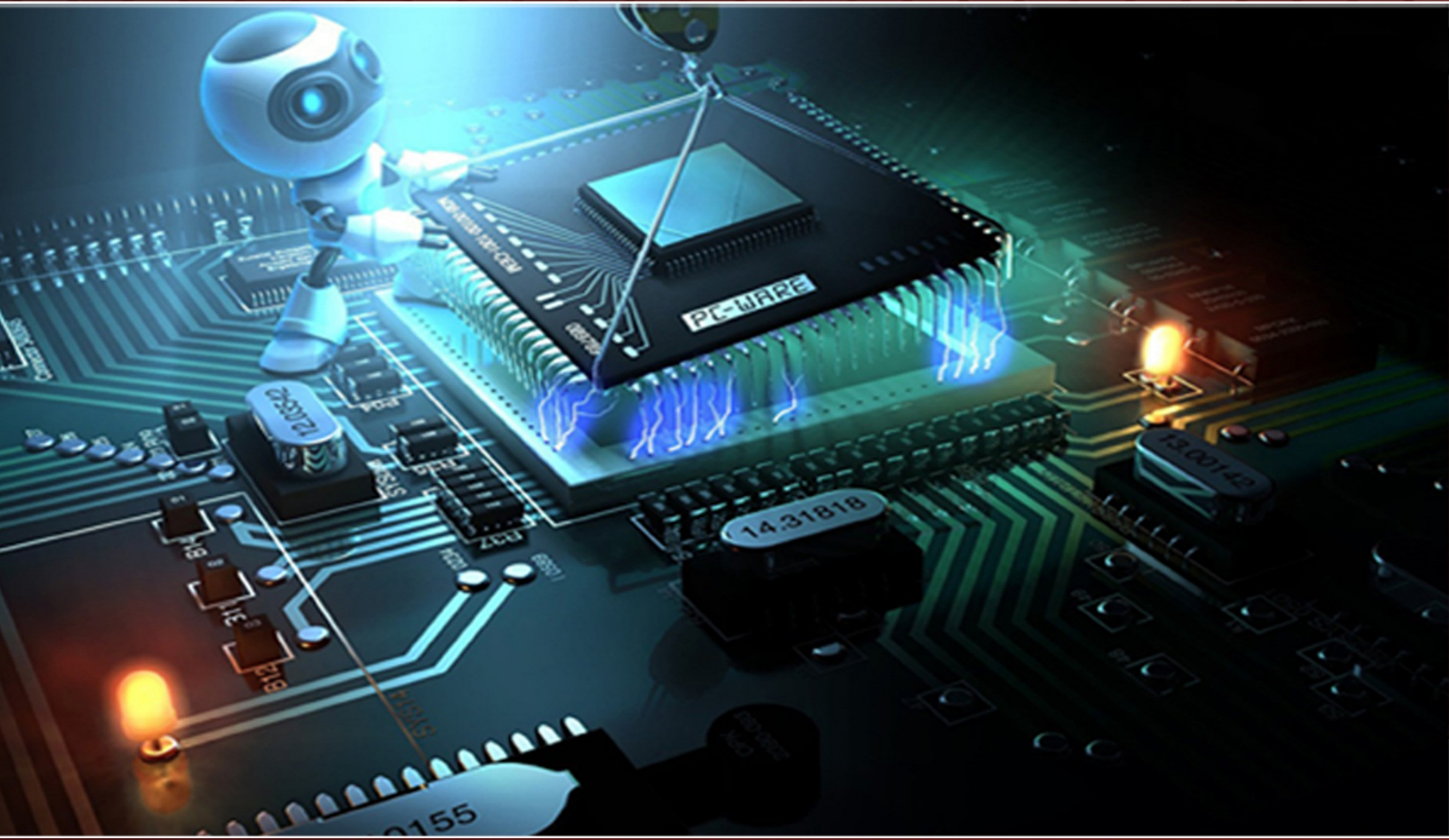
ISSN: 2054-7390

Volume 6 No 2, April 2018

Transactions on Machine Learning and Artificial Intelligence



Society for Science and Education
UNITED KINGDOM

TABLE OF CONTENTS

EDITORIAL BOARD

DISCLAIMER

Society for Science and Education
United Kingdom

TMLAI Transactions on Machine Learning and Artificial Intelligence

Volume 6 No. 2
ISSN 2054-7390

Artificial Human Optimization – An Introduction

Satish Gajawada
Alumnus, Indian Institute of Technology Roorkee, INDIA
Intern, ThinCI Semiconductor Technologies India Pvt. Ltd.
gajawadasatish@gmail.com

ABSTRACT

The goal of this article is;

1) To popularize "Artificial Human Optimization" field
2) To show opportunities that exist in "Artificial Human Optimization" field.
3) To Design an optimization method based on Artificial Humans
4) To show reviews of papers in “Artificial Human Optimization” field
5) To make corrections to my previous work in “Artificial Human Optimization” field
6) To encourage researchers across the globe to work in “Artificial Human Optimization” field
7) To give Artificial Human Optimization award to researchers who contributed to this new field

KEYWORDS: Artificial Intelligence, Machine Learning, Global Optimization Techniques, Genetic Algorithms, Particle Swarm Optimization, Ant Colony Optimization, Artificial Fish Swarm Optimization, Firefly Optimization Algorithm, Flower Pollination Algorithm, Artificial Human Optimization

1 Introduction

Recently a new field titled “Artificial Human Optimization” has been proposed in literature [1]. In this paper the focus is completely on “Artificial Human Optimization” field.

Rest of the paper is organized as follows:

1) Section 2: This section shows corrections to earlier work in Artificial Human Optimization field
2) Section 3: This section shows design of new optimization method based on Artificial Humans. “Multiple Strategy Human Optimization” is the title of this new method.
3) Section 4: This section shows reviews of experts in Artificial Human Optimization field.
4) Section 5: This section encourages future researchers to work in Artificial Human Optimization field.
5) Section 6: To show opportunities that exist in "Artificial Human Optimization" field.
6) Section 7: To show details related to the “Founder of Artificial Human Optimization” field.
7) Section 8: This section shows list of researchers who worked in Artificial Human Optimization field and got Artificial Human Optimization Award (which is given by the founder of Artificial Human Optimization).

DOI: 10.14738/tmlai.62.4182
Publication Date: 20th February, 2018
URL: http://dx.doi.org/10.14738/tmlai.62.4182

2 Corrections to Earlier Work

13 abstracts of papers in Artificial Human Optimization field are shown in [1]. In [1] it was given that 2012 was the year in which first paper in the field was proposed. But the first optimization method based on Artificial Humans was proposed in 2009 [2]. The abstracts of papers missed in [1] are shown below:

The abstract given in [2] is shown below as it is:

"The Human-Inspired Algorithm (HIA) is a new algorithm that uses a given population (a group of candidate solutions) to improve the search for optimal solutions to continuous functions in different optimization applications such as non-linear programming. HIA imitates the intelligent search strategies of mountain climbers who use modern techniques (such as binoculars and cell phones) to effectively find the highest mountain in a given region. Different from Genetic Algorithms (GAs) and Bees Algorithms (BAs), HIA divides a whole search space into multiple equal subspaces, evenly assigns the population in the subspaces, finds an elite subspace with the largest sum of function values, and uses more climbers (candidate solutions) to explore the elite subspace and fewer ones to explore the rest of the whole search space. BAs use random search in local neighborhood search, whereas HIA uses GAs in local neighborhood search to obtain better results. HIA locates a point with the largest function value among the elite sites and creates a hypercube with the point as its center. The assigned climbers in the hypercube and the elite subspace continue to search for the optimal solution iteratively. In each loop, the hypercube and the elite subspace become smaller to have a larger chance to pinpoint the optimal solution. Simulation results for three continuous functions with constraints and three continuous functions with box constraints can indicate that HIA is more efficient than GAs and BAs. Finally, conclusions and future works are given."

The abstract given in [3] is shown below as it is:

"This paper introduces Anarchic Society Optimization (ASO), which is inspired by a social grouping in which members behave anarchically to improve their situations. The basis of ASO is a group of individuals who are fickle, adventurous, dislike stability, and frequently behave irrationally, moving toward inferior positions they have visited during the exploration phase. The level of anarchic behavior among members intensifies as the level of difference among members' situations increases. Using these anarchic members, ASO explores the solution space perfectly and avoids falling into local optimum traps. First we present a unified framework for ASO, which can easily be used for both continuous and discrete problems. Then, we show that Particle Swarm Optimization (PSO), for which a general introduction was initially implemented for continuous optimization problems, is a special case of this framework. To evaluate the performance of ASO for discrete optimization, we develop an ASO algorithm for a challenging scheduling problem. The numerical results show that the proposed ASO algorithm significantly outperforms other effective algorithms in the literature. Our study indicates that developing an ASO algorithm is basically straightforward for any problem to which a PSO or Genetic algorithm has been applied. Finally, it is shown that under mild conditions an ASO algorithm converges to a global optimum with probability one."

3 Proposed Method

Please read "Terminology" section in paper [4] to understand the below explanation of proposed method: In initialization stage Locations of Humans, Guidance Locations of Humans, Love Array and Step are initialized. There are 2 generations inside the while loop shown in Figure 1. As shown in [4] computation of Generation x (Gen x) is done. The only difference is that a new method is designed for updating

Guidance Locations of Humans. Circle Best is the best fitness value among all fitness values of Guidance Locations of particular Human. Complete Best is the best fitness value among all fitness values of Guidance Locations of all Humans. Probabilities are assigned to Circle Best and Complete Best based on their fitness values. Probabilities are calculated from fitness values of Circle Best and Complete Best in the same way probabilities are calculated in [4]. A random number is generated to select either Circle Best or Complete Best. A particular Guidance Location moves towards the selected and this movement is similar to movement of Humans described in [4]. The difference is that Guidance Locations move towards the selected by the value of Step. The same strategy is used to update all Guidance Locations of all Humans. In Generation x+1, Guidance Locations are updated in the same way Guidance Locations are updated in Generation x (Gen x). The difference between Gen x and Gen x+1 lies in the movement of humans. In Gen x, humans move towards Guidance Locations of higher fitness values with higher probability. But in Gen x+1, humans move away from lower fitness values with higher probability. Functions Update Gen x() and Update Gen x+1() are shown in Figure 1. These functions update Guidance Locations, Love Array and Step values.

```
main (){
  Initialization();
        while(Termination_Condition_Reached not equal to true){
                Generation x : Human moves towards higher fitness Guidance Locations.
                        Update Gen x().
                Generation x+1 : Human moves away from lower fitness values.
                        Update Gen x+1().
        }
  }
```

Figure 1. Multiple Strategy Human Optimization

4 Reviews

Review 1 given in [5] is shown below as it is:

"This paper studies a so-called human optimization method which falls into the research topic of optimization. The proposed method was presented on the first page followed by some discussions. The paper clearly makes no novel contribution to the state of the art on optimization algorithms and techniques. Thus, because of this lack of new contribution, the paper is not appropriate for the conference."

Review 2 given in [5] is shown below as it is:

"Based on the review of your abstract, the following editorial comments should be taken into consideration:

Please submit an abstract. Change font type. Remove PhD from the title.

Please follow the abstract guidelines".

Review 3 given in [5] is shown below as it is:

“Nothing to evaluate.”

Review 4 given in [5] is shown below as it is:

“Funny paper, especially the notion of "love array" :)”

Review 5 given in [5] is shown below as it is:

“This is not a research paper. It should not have been submitted for review.

Rationale and results are completely lacking. I do not even think there is a research idea in there.”

Review 6 given in [5] is shown below as it is:

“General conclusion is ‘Accept without reservation’.

Further comments of the evaluator are below:

The title should be changed to be more comprehensive. The clarity and relevance of the problem is well stated. How is the problem scientifically analyzed through the text? the main propositions of the paper are crystal clear. The conclusion part should also contain more details expressing if other researches in the field support the results. The text needs to be re-considered by a native English speaker to edit the errors. It is recommended that the author adds more sources since the year 2012. The research method should be explained in more details.”

Review 7 given in [5] is shown below as it is:

“General conclusion is ‘Accept without reservation’.

Further comments of the evaluator are below:

The title is well in accord with the body of the text. The clarity and relevance of the problem is well stated. How is the problem scientifically analyzed through the text? Reasoning of main propositions are satisfying. In conclusion part, It is needed to support the result of the research by other recent researches. The English language needs little modification in abstract part. The references are good but it is recommended that the author uses more references from the recent years. The author needs to make the main goals crystal clear.”

Review 8 given in [5] is shown below as it is:

“Paper has been ACCEPTED.

Specific behavior of the human has to be specified for the model.

Few Examples/scenarios where this could be applied has to be explained.

The time complexity of the optimization algorithm has to be demonstrated over the brute force method.

Initialization of Guidance location and generalized form of updating the guidance location/love array should be explained in detail with appropriate formula.

Paper is very abstract about the idea discussed.”

Review 9 given in [5] is shown below as it is:

"Main advantages of the work:

1. Rather conceptual work pondering another interesting approach to optimization problem solution. Goals are clearly stated and the new algorithm is provided and explained.

Main disadvantages of the work:

1. Qualitative comparison to other optimization algorithms is not provided. Why proposed algorithm could be thought as specifically modeling human optimization is not fully explicated.
2. It is not clearly stated whether Guidance Locations and Love array are local or global, i.e. are they vectors or matrices? Seems like the latter.

Decision: this paper should be accepted for participation in the conference".

Review 10 given in [5] is shown below as it is:

"Main advantages of the work:

2. New method for the creation of innovative optimization algorithms is proposed in the work.
3. The function Update Locations of Humans in optimization algorithm explained in depth.
4. An overview of existing works on the same topic is provided.
5. Calculations of the fitness values of guidance locations of the Human are analyzed.

Main disadvantages of the work:

1. It is not demonstrated how PhD method have been applied for solving complex optimization problems.
2. It is not clear either there are some software implementation of Human Optimization that confirm practical feasibility of the method.

Decision: this paper should be accepted for participation in the conference."

Review 11 given in [5] is shown below as it is:

"Review 11 a: A very interesting paper.

Review 11 b: I have to admit that I had a hard time grasping the key concepts revealed in this manuscript. The author has set a very ambitious goal. But I am still searching for the elements that will make this goal a reality. The proposed algorithm is simply too abstract to be of substantial value."

5 Encouragement to Future Researchers

From section 4 it is clear that some experts are against to Optimization methods based on Artificial Humans whereas other experts are supporting Artificial Human Optimization field. The author of this paper received review "Very Interesting work" from IEEE TAAI 2013 conference for a work in Artificial Human Optimization field. Now there are already more than 15 papers published in this field. There is scope for many PhD's and PostDoc's in Artificial Human Optimization field.

For the sake of encouraging researchers, 15 titles of papers published in Artificial Human Optimization field are shown below. 13 abstracts are already shown in [1]. Titles of papers shown in [1] are given in double quotes:

"(1) Manoj Kumar Singh," A New Optimization Method Based on Adaptive Social Behavior: ASBO", AISC 174, pp. 823–831. Springer, 2012."

“(2) Satish Gajawada, “POSTDOC : The Human Optimization”, Computer Science & Information Technology (CS & IT), CSCP, pp. 183-187, 2013.”

“(3) Liu H, Xu G, Ding GY, Sun YB, “Human behavior-based particle swarm optimization”, The Scientific World Journal, 2014.”

“(4) Da-Zheng Feng, Han-Zhe Feng, Hai-Qin Zhang, “Human Behavior Algorithms for Highly Efficient Global Optimization”, https://arxiv.org/abs/1507.04718, 2015.”

“(5) Seyed-Alireza Ahmadi, “Human behavior-based optimization: a novel metaheuristic approach to solve complex optimization problems”, Neural Computing and Applications, 2016.”

“(6) Ruo-Li Tang, Yan-Jun Fang, "Modification of particle swarm optimization with human simulated property", Neurocomputing, Volume 153, Pages 319–331, 2015.”

“(7) Muhammad Rizwan Tanweer, Suresh Sundaram, "Human cognition inspired particle swarm optimization algorithm",2014 IEEE Ninth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2014.”

“(8) M. R. Tanweer, S. Suresh, N. Sundararajan, "Human meta-cognition inspired collaborative search algorithm for optimization", 2014 International Conference on Multisensor Fusion and Information Integration for Intelligent Systems (MFI), pp. 1-6, 2014.”

“(9) M.R. Tanweer, S. Suresh, N. Sundararajan, "Self regulating particle swarm optimization algorithm", Information Sciences: an International Journal, Volume 294, Issue C, Pages 182-202, 2015.”

“(10) M. R. Tanweer, S. Suresh, N. Sundararajan, "Improved SRPSO algorithm for solving CEC 2015 computationally expensive numerical optimization problems", 2015 IEEE Congress on Evolutionary Computation (CEC), pp. 1943-1949, 2015.”

“(11) Prakasha S, H R Shashidhar, Manoj Kumar Singh, G T Raju, ”Clustering of Text Document based on ASBO”, Wulfenia journal, Vol 20, No. 6; pp: 152-165, 2013.”

“(12) Sridhar N, Nagaraj Ramrao, Manoj Kumar Singh, "PID Controller Auto tuning using ASBO Technique”, Journal of Control Engineering and Technology, Vol. 4, Iss. 3, PP. 192-204, 2014.”

“(13) Devika P. D, Dinesh P. A, Rama Krishna Prasad, Manoj Kumar Singh, "ASBO Based Compositional in Combinatorial Catalyst", J. Math.Comput.Sci.5 (2015), No.3, 351-393, ISSN: 1927-5307, 2015.”

(14) L. M. Zhang, C. Dahlmann and Y. Zhang. Human-inspired algorithms for continuous function optimization. In IEEE International Conference on Intelligent Computing and Intelligent Systems, 2009, vol. 1, pp. 318-321.

(15) A. Ahmadi-Javid, "Anarchic Society Optimization: A human-inspired method", Proc. 2011 IEEE Congr. Evol. Comput., pp. 2586-2592, 2011.

6 Opportunities in Artificial Human Optimization Field

Following are some of the opportunities that exist in Artificial Human Optimization field:

1) International Association of Artificial Human Optimization (IAAHO)
2) Transactions on Artificial Human Optimization (TAHO)

3) International Journal of Artificial Human Optimization (IJAHO)
4) International Conference on Artificial Human Optimization (ICAHO)
5) www.ArtificialHumanOptimization.com
6) B.Tech in Artificial Human Optimization
7) M.Tech in Artificial Human Optimization
8) PhD in Artificial Human Optimization
9) PostDoc in Artificial Human Optimization
10) Artificial Human Optimization Labs
11) To become "Father of Artificial Human Optimization" field

7 Founder of Artificial Human Optimization Field

In December 2016, Satish Gajawada proposed a new field titled "Artificial Human Optimization" which comes under Artificial Intelligence. This work was published in "Transactions on Machine Learning and Artificial Intelligence".

8 Artificial Human Optimization Awards

The following list of researchers are awarded "Artificial Human Optimization Award" for their valuable contribution to AHO field:

1) Manoj Kumar Singh
2) Liu H
3) Xu G
4) Ding GY
5) Sun YB
6) Da-Zheng Feng
7) Han-Zhe Feng
8) Hai-Qin Zhang
9) Seyed-Alireza Ahmadi
10) Ruo-Li Tang
11) Yan-Jun Fang
12) Muhammad Rizwan Tanweer
13) Suresh Sundaram
14) N. Sundararajan
15) Prakasha S
16) H R Shashidhar
17) G T Raju
18) Sridhar N
19) Nagaraj Ramrao
20) Devika P. D
21) Dinesh P. A
22) Rama Krishna Prasad
23) L. M. Zhang
24) C. Dahlmann
25) Y. Zhang
26) A. Ahmadi-Javid

ACKNOWLEDGEMENTS

Thank you to everyone who directly or indirectly helped me to reach the stage where I am now.

9 Conclusion

This paper shows how to contribute to new field titled "Artificial Human Optimization". There are many opportunities in Artificial Human Optimization field. There is scope for many PhD's and PostDoc's in Artificial Human Optimization field.

MOTIVATION

Great leaders dont tell you what to do, they show you how it's done -- from internet.

REFERENCES

[1] Satish Gajawada; Entrepreneur: Artificial Human Optimization. Transactions on Machine Learning and Artificial Intelligence, Volume 4 No 6 December (2016); pp: 64-70.

[2] L. M. Zhang, C. Dahlmann and Y. Zhang. Human-inspired algorithms for continuous function optimization. In IEEE International Conference on Intelligent Computing and Intelligent Systems, 2009, vol. 1, pp. 318-321.

[3] A. Ahmadi-Javid, "Anarchic Society Optimization: A human-inspired method", Proc. 2011 IEEE Congr. Evol. Comput., pp. 2586-2592, 2011.

[4] Satish Gajawada, "POSTDOC : The Human Optimization", Computer Science & Information Technology (CS & IT), CSCP, pp. 183-187, 2013.

[5] Satish Gajawada, "CEO: Different Reviews on PhD in Artificial Intelligence", Global Journal of Advanced Research, vol. 1, no.2, pp. 155-158, 2014.

AUTHOR

In December 2016, Satish Gajawada proposed a new field titled "Artificial Human Optimization" which comes under Artificial Intelligence. This work was published in "Transactions on Machine Learning and Artificial Intelligence". He received a SALUTE and APPRECIATION from IEEE chair Dr. Eng. Sattar B. Sadkhan for his numerous achievements within the field of science. He completed his studies from world class institute "Indian Institute of Technology Roorkee (IIT Roorkee)". Below are some publications of author:

Satish Gajawada, Durga Toshniwal, Nagamma Patil and Kumkum Garg, "Optimal Clustering Method Based on Genetic Algorithm," International Conference on Soft Computing for Problem Solving (SocPros - 2011), Springer.

Satish Gajawada, Durga Toshniwal, "Hybrid Cluster Validation Techniques," International Conference on Computer Science, Engineering & Applications (ICCSEA - 2012), Springer.

Satish Gajawada, Durga Toshniwal, ”Projected Clustering Using Particle Swarm Optimization,” International Conference on Computer, Communication, Control and Information Technology (C3IT - 2012), Elsevier.

Satish Gajawada, Durga Toshniwal, ”GAP: Genetic Algorithm Based Projected Clustering Method”, 21st International Conference on Software Engineering and Data Engineering (SEDE 2012), USA.

Satish Gajawada, Durga Toshniwal, “Projected Clustering Particle Swarm Optimization and Classification”, International Conference on Machine Learning and Computing (ICMLC-2012), Hong Kong.

Satish Gajawada, Durga Toshniwal, “VINAYAKA: A Semi-Supervised Projected Clustering Method

Using Differential Evolution,” International Journal of Software Engineering and Applications (IJSEA),

2012.

Satish Gajawada, Durga Toshniwal, “A framework for classification using genetic algorithm based clustering”, The International Conference on Intelligent Systems Design and Applications (ISDA), 2012, IEEE.

You can read all the work of the author at - https://iitr-in.academia.edu/SatishGajawada

SOCIETY FOR SCIENCE AND EDUCATION
UNITED KINGDOM

TMLAI TRANSACTIONS ON
MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE

Volume 6 No. 2
ISSN 2054-7390

Intelligent Decision Support Machines for Business Decisions

Hans W. Gottinger, STRATEC Munich
stratec_c@yahoo.com, gottingerhans@gmail.com

"How can we move from the classical view of rational agent who maximizes expected utility over an enumerable state-action space to a theory of the decisions faced by resource bounded AI systems deployed in the real word…" --- Gershman, Horvitz and Tenenbaum (2015).

ABSTRACT

An intelligent decision support machine (IDSM) is a computer-based interactive tool of decision making for well-structured decision and planning situations that uses jointly decision-theoretic methods and machine learning techniques with access to structured data bases. An IDSM emerges from a model based system underlying a decision model (under uncertainty) imposing a normative (prescriptive) structure of decision making.

In the past years there has been substantial attention devoted to the use of artificial intelligence (AI) techniques, first most commonly rule-based expert systems, more recently methods of machine learning as tools for decision support. Based on those rules we design a machine learning algorithm guiding through principles of an IDSM.

1 Introduction

An intelligent decision support machine (IDSM) is a computer-based interactive tool of decision making for well-structured decision and planning situations that uses jointly decision-theoretic methods and machine learning techniques with access to structured data bases. An IDSM emerges from a model based system underlying a decision model (under uncertainty) imposing a normative (prescriptive) structure of decision making.

In the past years there has been substantial attention devoted to the use of artificial intelligence (AI) techniques, most commonly rule-based expert systems, as tools for decision support. These systems typically use production rules to develop a diagnosis of a disease or a system malfunctioning, for example.

Given the diagnosis, the system generates a recommended solution to the problem. The solution may be a drug therapy in a medical domain or a set of parts to replace in a trouble shooting application.

Rule based techniques have proven to be very attractive for a variety of problems, particularly those which have fairly well structured (though possibly large) problem spaces, which can be solved through the use of heuristic methods or rules of thumb, and are currently solved by human experts. In these domains the reasoning and explanation capabilities offered by rule based expert systems are very effective. Given various types of goal or payoff functions the system generates a range of decision outcomes to choose

DOI: 10.14738/tmlai.62.4372
Publication Date: 29th March, 2018
URL: http://dx.doi.org/10.14738/tmlai.62.4372

from. Sequential decision rules need to be updated when applied to time evolving difficult problems relating to the following situations:

(i) there is substantial uncertainty on various levels of decision making;

(ii) the preferred solution is sensitive to the specific preferences and desires of one or several decision makers;

(iii) problems of rationality and behavioral coherence are intrinsic concerns of decision systems.

In related established fields such as operations research and management science we have been developing methods for allocating resources under various conditions of time, uncertainty and rationality constraints. Central to these methods is the existence of an objective or utility function, as an indicator of the desirability of various outcomes. We will draw on this body of knowledge, especially elements related to the normative use of individual and group decision making to approach difficult decision problems.

Decision making is best viewed as a process of making a series of related, incremental observations, judgements, and decisions. In some cases simple, deterministic relations such as those used in many rule-based expert systems can be used for informed decision making, while in others explicit treatment of uncertainty or vagueness in the domain and the objectives of the decision maker are needed. Therefore, for a single complex decision situation it may be desirable to combine deterministic reasoning with uncertainty and decision calculi in the course of exploring the decision situation, as our understanding and insight into a problem evolves. It is a move toward a "machine economicus" (Parkes and Wellman, 2015).

2 Structuring Decisions for Computation

An early fusion of Artificial Intelligence and Decision-Making started with expert systems consisting of a Knowledge Base (KB) and Inference Engine (IE) (Gottinger, and Weimann, 1990).

Expert Systems were basically static constructs.. The KB had to be constructed and created manually on present and past knowledge structure --- there was no endogeneous learning from data.. The inference engine consisted of logical inference rules, i.e. if-then rules, that were operating repeatedly on given knowledge bases. Yet the architecture of interaction between KB and IE can still be used in machine learning (ML) mode by opening up the KB to dynamic time related changes and future (uncertain) events, in the same way as we perceive Markov chains and processes. Let logic rules work probabilistically on changing events in a decision-theoretic framework. Thus ML systems are extensions of ES in two-fold ways: they work as mechanism designs on variety, diversity of decision relevant events, and they operate on probabilistic events in future courses of action (Pearl, 1993).

A procedure of ML on decision tree structures could be roughly as follows. First observe time-dependent changes in probability estimates with Bayesian updating on initial probability and likelihoods for given sets of events affecting outcomes (expected utility). Then preference changes on alternatives induce changing utilities. Further, risk preferences change on probabilistic outcomes (risk proneness, aversion).

The basic result of the axioms of decision theory is the existence of a value function for scoring alternative sets of outcomes under certainty and a utility function for scoring uncertain outcome bundles. If the decision maker accepts the axioms (say, Savage's axioms, Savage 1954) in the sense that he would like his decision making to be consistent with these axioms, then the decision maker should choose that course of action which maximizes expected utility – with its early lead to stochastic learning models (Bush and Mosteller, 1955). The importance of these axioms is that encoding decision procedures based on these axioms provide a basis for recommendations by an intelligent decision aid under uncertainty. They provide an explicit set of norms by which the system will behave. Other authors have argued why an individual should accept the decision axioms for decision making. The acceptance of these axioms is implicit in the philosophy and design of decision methods described here.

In addition, an approach to decision making based on decision theory has a mechanism, at least in principle, for handling completely new decision situations. The construct ensures the existence of a value and utility function. If the current expression of the preferences in the system does not incorporate the attributes of a new decision situation, the system can resort to the construction of a higher level or more general preference structure. By following these principles we are able to use the richness of modern decision theory and their axiomatic foundation (Fishburn 1988).

Domains in which there is a well developed empirical and theoretical basis for development of utility functions (e.g. financial and engineering decision making and some areas in medicine) are most promising.

Thus the decision axioms, along with the fundamentals of first order logic, provide a normative basis for reasoning about decisions. It is in this light that both logical and probabilistic inference will be utilized in an intelligent decision system.

3 Decision model based Representation

For decision making, a model consists of the following elements

(1) alternative prospects

(2) state descriptions,

(3) relationships, and

(4) preferences.

There can be no decision without alternatives, the set of distinct resource allocations from which the decision maker can choose. Each alternative must be clearly defined. State descriptions are essentially collections of concepts with which the decision is framed. It includes the decision alternatives and the outcomes which are related to the choices. The state description forms the means of characterizing the choice and outcome involved in a decision. The state description is also intertwined with expression of relationships. Relationships are simply the mappings of belief in some elements of the state description to others. The relations could be represented as logic relations, if-then rules, mathematical equations, or conditional probability distributions. The final component of a decision model is preferences. These are the decision maker's rankings in terms of desirability for various possible outcomes. They include not only his rankings in terms of the various outcomes which may occur in a decision situation, but also his attitude toward risky outcomes and preferences for outcomes which may occur at various times. They also embody information identifying those factors in a decision situation that are of concern, whether a factor indicates a desirable or undesirable outcome, and how to make tradeoffs among alternative collections of outcomes.

4 Influence Diagrams

As a computationally convenient way for a decision model based representation we deal with influence diagrams. Influence diagrams are network depictions of decision situations (Howard & Matheson 1981). They lend themselves to a dynamic representation of decision situations. They also give rise to forms of neural network presentations facilitating machine learning with deep learning (Davis,2016). Until recently, their primary use has been in the professional practice of decision analysis as a means of eliciting and communicating the structure of decision problems. Each node in the diagram represents a variable or decision alternative; links between nodes connote some type of influence'. Decision makers and experts in a given domain can view a graphical display of the diagram, and readily apprehend the overall structure and nature of dependencies depicted in the graph. There has been additional attention devoted to influence diagrams based on their uses in providing a complete mathematical description of a decision problem and as representations for computation. In addition to representing the general structure of a decision model, information characterizing the nature and content of particular links is attached to the diagram (Holtzman, 1989). The diagram then presents a precise and complete specification of a decision maker's preferences, probability assessments, decision alternatives, and states of information. In addition the diagrammatic representations can be directly manipulated to generate decision-theoretic recommendations and to perform probabilistic inference. The formalism of Bayes networks (Pearl,1993) are identical graphical constructs which express probabilistic dependencies (no preferences or decisions). Following the notation of Shachter (1986) we define the syntax and semantics of influence diagrams.

Definition .An influence diagram is an acyclic directed graph G = (N,A) consisting of a set, N, of nodes and a set, A, of arcs. The set of nodes, N, is partitioned into subsets V, C, and D. There is one value node in V, representing the objective of the decision maker. Nodes in C, the chance nodes, represent uncertain outcomes. Nodes in D, the decision nodes, represent the choices or alternatives facing the decision maker.

A simple diagram appears in Fig.1. By convention, the value node is drawn as a diamond, chance nodes are drawn as circles, and decision nodes are drawn as rectangles.

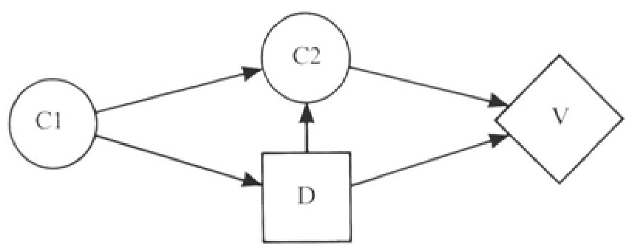


Figure.1 — A Simple Influence Diagram.

V is the value node, the proposition which embodies the objective to be maximized in solving the decision problem. C1 and C2 represent uncertainties, and D represents the decision. The semantics of arcs in the graph depend on the type of the destination node. Arcs into value or chance nodes denote probabilistic dependence. These arcs will be referred to as probabilistic links. Arcs terminating in decisions indicate the state of information at the time a decision is made. Thus, C1 is an uncertainty which is probabilistically influenced (conditioned) by C2 and the decision. The ultimate outcome V, depends on the decision D and C2.

Definition — Each node's label is a restricted proposition, a proposition of the form p (t_1 t_2 ... t_n) where each t_i, is either an object constant or alternative set.

We now define a set Ω(i) and a mapping π_i, for each node.

Definition — The set Ω(i) is the outcome set for the proposition represented by node i. It is a set of mutually exclusive and collectively exhaustive outcomes for the proposition.

Definition — The predecessors of a node i are the set of nodes j with arcs from j to i.

Definition — The successors of a node i are the set of nodes j such that there is an arc from i and j.

The mapping π_i, depends on node type. The domain of each mapping is the cross product of the outcome sets of the predecessors of node i. Let the cross product of predecessors of i be CP(i) where

$$CP(i) = \{ \Omega(i1) \times \Omega(i2) \ldots \times \Omega(in) | \text{nodes } i1, \ldots, in$$

ϵ predecessors of node i}

The range of each mapping π_i depends on the type of node i. Each is discussed in turn.

The value node

The value node expresses the decision maker's relative valuation of different possible combinations of outcomes for its predecessors. Since we require a cardinal measure for expected value calculations, the outcome set of value nodes has some restrictions. In terms of the previous definition of the outcomes set, the set Ω(i) for the value node is:

$$\Omega(i) = \{i\Theta | \Theta \in \{\{x_1/X_{11}\}\{x_1/X_{12}\}, \ldots, \{x_1/X_{1K}\}\}\}$$

where $X_{11} \ldots X_{1K} \in R$

Thus, there is only one restricted variable, x_1, and its values, the $X_{1K}{}^S$, are real valued. We can therefore associate with each member of Ω(i) exactly one real number. The value function π_i, for a value node is defined as follows:

$$\pi_i: CP(i) \rightarrow \text{alternative set of } X_1 \{X_{11}, X_{12}, \ldots, X_{1K}\}$$

This function maps each combination of outcomes for the predecessors into a single real number. This will be used to express the expected value or the expected utility as a function of the outcomes of the predecessor nodes.

Chance nodes

Chance nodes represent uncertain propositions that are not directly controlled by the decision maker. The members of Ω(i) are the possible outcomes for the proposition. There are two types of chance nodes. A stochastic chance node admits uncertainty regarding the outcome of the proposition given the values of its predecessors. In this case the mapping π_i is a conditional probability density function.

$$\pi_i: CP(i) \times \Omega(i) \rightarrow [0,1]$$

in probabilistic terms $\pi_i(\omega_i | \ \omega_{i1}, \ \omega_{i2}, \ \ldots \ \omega_{in})$. If node i has no predecessors, then $\pi_i(\omega_i)$ is a prior (unconditional) probability distribution.

The other type of chance node is a deterministic chance node. The outcome of a deterministic chance node is a deterministic function of the outcomes of its predecessors. In this case, the mapping π_i, is defined as follows.

$$\pi_i: CP(i) \rightarrow \Omega(i)$$

Thus, given the values of its predecessors, there is no uncertainty regarding the outcome of the proposition. If node z has no predecessors, the $\pi_i(\ )$ is constant.

Decision nodes

Decision nodes represent propositions which are under the direct control of the decision maker. The members of $\Omega(i)$ are the alternative outcomes from which the decision maker can choose. At the time the decision is made, the decision maker knows the outcomes of the predecessors of i. The mapping π_i therefore expresses the optimal decision choice as a function of what is known at the time the decision is made.

$$\pi_i: CP(i) \rightarrow \Omega(i)$$

The mapping is calculated in the course of manipulating an influence diagram and the associated maximization of expected value. Though the construct is similar to that of the mapping for a deterministic chance node, it differs in that it is the result of an optimization. For this reason we define a special construct.

Definition — A decision function $\pi_{d,i}$, is a function

$$\pi_{d,\, i}: CP(i) \rightarrow \Omega(i)$$

determined as the result of an optimization (see Transformations).

Transformations

An influence diagram is said to be a decision network if (1) it has at least one node, and (2) if there is a directed path which contains all the decision nodes (Howard & Matheson 1981). The second condition implies that there is a time ordering to the decisions, consistent with the use of an influence diagram to represent the decision problem for an individual. Furthermore, arcs may be added to the diagram so that the choices made for any decision are known at the time any subsequent decision is made. These are 'no-forgetting' arcs, in that they imply the decision maker (1) remembers all of his previous selections for decisions, and (2) has not forgotten anything that was known at the time of a previous decision.

The language of influence diagrams is a clear and computable representation for a wide range of complex and uncertain decision situations. The structure of dependencies (and lack thereof) is explicit in the linkages of the graph, as are the states of information available at each state in a sequence of decisions. The power of the representation lies, in large part, in the ability to manipulate the diagram to either (1) express an alternative expansion of a joint probability distribution underlying a particular model, or (2) to generate decision recommendations. The basic transformations of the diagram required to perform these operations are node removal and arc reversal. These operations will be illustrated and defined with respect to a generic set of node labels: i and j are chance nodes, v is the value node. The labels p1, p2, and p3 will in general represent groups of predecessors of i, j, or v as indicated by the figures. In the interest of simplifying the descriptions of the operations, they will be treated as individual nodes. More detailed descriptions of these operations appear in Shachter (1986) and Holtzman(1989).

Removal of a stochastic chance node, i, which is a predecessor of a value node, v, is performed by taking conditional expectation.

The new expected value function for v is calculated as follows:

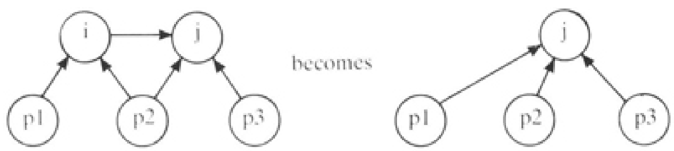
$$\pi_{\text{new},v}\,(\omega_{p1},\omega_{p2},\omega_{p3}) = \sum_{\omega_i \in \Omega(1)} \pi_{\text{old},v}(\omega_{p1},\omega_{p2},\omega_{p3})\ \pi_i(\omega_i | \omega_{p1},\omega_{p2})$$

The value nodes new predecessors are p 1, p2, and p3.

Removal of a deterministic chance node, i, which is a predecessor to the value node, v, is performed by substitution. The picture of this process is the same as the previous case. The new expected value function for v is:

$$\pi_{\text{new, v}}(\omega_{p1}, \omega_{p2}, \omega_{p3}) = \pi_{\text{old, v}}(\ \pi_i,\omega_{p2}),\ \omega_{p2},\ \omega_{p3})$$

Removal of a stochastic chance node, i, which is a predecessor to another chance node, j, is also performed by taking conditional expectation.

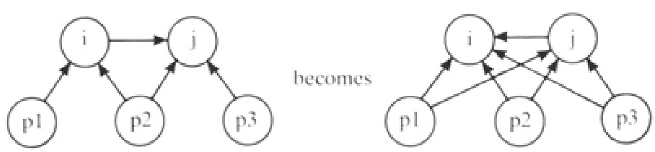


The new distribution for successor node j is calculated as:

$$\pi_{\text{new},j}(\omega_{p1},\omega_{p2},\omega_{p3}) = \sum_{\omega_i \in \Omega(i)} \pi_{\text{old},j}(\omega_j | \omega_i,\omega_{p2},\omega_{p3})\ \pi_i(\omega_i | \omega_{p1},\omega_{p2})$$

The new predecessors of j are the predecessors of j other than i, that is, p1, p2 and p3.

Removal of a decision node, i, predecessor to the value node v is performed by maximizing expected utility. The decision node can only be removed when all of its predecessors are also predecessors of the value node; that is, the choice is based the expectations for the value, given what is known.



After removal the new expected value function for v is:

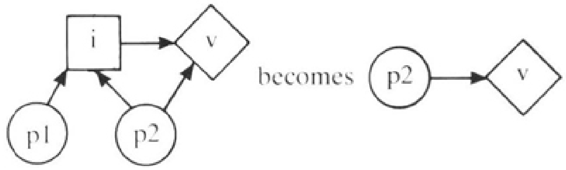
$$\pi_{\text{new},v}(\omega_{p2}) = \operatorname*{Max}_{\omega_i \in \Omega(i)}\ \pi_{\text{old},v}(\omega_i,\omega_{p2})$$

The new predecessors of v are the predecessors of i which are also predecessors of v, p2 as illustrated here. Note that there may be some informational predecessors of i, for example p1, which are not predecessors v before the removal. The values of these variables are irrelevant to the decision, since the expectation for the value is independent of their values. The optimal policy for the decision i is:

$$\pi_i = \operatorname*{argmax}_{\omega_i \in \Omega(i)}\ \pi_{\text{old},v}(\omega_i,\omega_{p2})$$

this is the calculated π_i for decision nodes. We will refer to this calculated mapping as the decision function for i, $\pi_{d,\,i}(\omega_{p2})$.

Reversal of a probabilistic link between chance nodes is an application of Bayes' rule. Reversing a link from node i to node j can be performed so long as there is no other path from i to j (this is necessary to prevent the reversal from creating a cycle).



In reversing, the new conditional probability description for i and j are calculated as:

$$\pi_{\mathrm{new},j}(\omega_j|\omega_{p1},\omega_{p2},\omega_{p3}) = \sum_{\omega_i \in \Omega(i)} \pi_{\mathrm{old},j}(\omega_j|\omega_i,\omega_{p2},\omega_{p3})\ \pi_{\mathrm{old},i}(\omega_i|\omega_{p1},\omega_{p2})$$

$$\pi_{\mathrm{new},j}(\omega_i|\omega_j,\omega_{p1},\omega_{p2},\omega_{p3}) = \frac{\pi_{\mathrm{old},j}(\omega_j|\omega_i,\omega_{p2},\omega_{p3})\ \pi_{\mathrm{old},i}(\omega_i|\omega_{p1},\omega_{p2})}{\pi_{\mathrm{new},j}(\omega_j|\omega_{p1},\omega_{p2},\omega_{p3})}$$

The operations of reversal and removal allow a well formed influence diagram to be transformed into another 'equivalent' diagram. The original and the transformed diagrams are equivalent in two senses. First, the underlying joint probability distribution and state of information associated with each is identical, since the diagram expresses alternative ways of expanding a joint distribution into a set of conditional and prior distributions (Howard & Matheson 1981). Secondly, the expectation for the value in the diagram and the sequence of recommended actions from decision node removal are invariant over these transformations (Shachter 1986, Holtzman 1989). In the next section, we focus on applying a sequence of these manipulations to obtain these recommendations.

Solution procedures

On the basis of these manipulations, there exist algorithms to evaluate any well formed influence diagram (Shachter 1986). For purposes of probabilistic inference, we need two separate algorithms. In one version, which applies to well formed diagrams, evaluation consists of reducing the diagram to a single value node with no predecessors, the value of which is the expected value of the decision problem assuming the optimal policy is followed. In the course of removing decisions, the optimal policy, that is, the set of decision functions $\pi_{d,\,i}$ associated with each decision is generated. In the other algorithm, the objective is to determine the probability distribution for a variable, as opposed to its expected value. Both versions of the algorithm are described below.

Procedure EXPECTED VALUE (diagram)

1. Verify that the diagram has no cycles.
2. Add 'no-forgetting' arcs between decision nodes as necessary.
3. WHILE the value node has predecessors

3.1 IF there exists a deterministic chance node predecessor whose only successor is the value node, THEN
Remove the deterministic chance node into the value node ELSE

3.2 IF there exists a stochastic chance node predecessor whose only successor is the value node, THEN
Remove the stochastic chance node into the value node ELSE

3.3 IF there exists a decision node predecessor and all the predecessors of the value node are predecessors of the decision node, THEN
Remove the decision node into the value node ELSE

3.4 BEGIN

3.4.1 Find a stochastic predecessor X to the value node that has no decision successors.

3.4.2 For each successor S_x of X such that there is no directed path from X to S_x
Reverse Arc from X to S_x

3.4.3 Remove stochastic predecessor X

4. END

At the conclusion of the EXPECTED VALUE procedure, the value node has no predecessors, and its single value is the expected value of the value node. Optimal decision functions are generated in the course of removing the decision nodes.

The algorithm to solve for a probability description (or lottery) for a node is as follows.

Procedure PROBABILITY-DISTRIBUTION (diagram)

1. Verify that the diagram has no cycles.
2. IF the value node is deterministic, THEN convert to a probabilistic chance node with unit probability on deterministic values.
3. WHILE the value node has predecessors

3.1 IF there exists a deterministic chance node predecessor whose only successor is the value node, THEN
Remove the deterministic chance node into the value node ELSE

3.2 IF there exists a stochastic chance node predecessor whose only successor is the value node, THEN
Remove the stochastic chance node into the value node ELSE

3.3 IF there exists a decision node predecessor and all the predecessors of the value node are predecessors of the decision node, THEN
Remove the decision node from the list of predecessor ELSE

3.4 BEGIN

3.4.1 Find a stochastic predecessor X to the value node that has no decision successors.

3.4.2 For each successor S_x of X such that there is no directed path from X to S_x
Reverse Arc from X to S_x

3.4.3 Remove stochastic predecessor X

4. END

The termination of this procedure is a probabilistic chance node with probabilities over the alternative possible outcomes of the original value node. Note that if decision predecessors are encountered in the algorithm, the distribution will be conditioned on the possible choice of the decision variables. The procedure does not remove decision nodes or generate decision functions.

5 Decision Language

Recall the elements that are necessary to represent a decision domain — alternatives, state descriptions, relationships, and preferences. We will summarize by indicating how each element of a decision description can be expressed with respect to the constructs generated above.

First, recall that propositions form the basic unit of representation for a decision domain. There are three levels of knowledge regarding a proposition expressible in the language. First, it is possible to express a fact for a proposition, that is, a set of values for the variables (as in a fact substitution) in the proposition that are asserted to be true with certainty. Second, the values of the variables in a proposition may be restricted to some set. Thus, the outcomes for that proposition are restricted to a collectively exhaustive, mutually exclusive set, termed the alternative outcomes. Finally, a probability distribution can be used to associate each possible outcome with a probability. We have also shown how probability distributions and outcome sets are expressed for conjunctions of propositions.

Alternatives, the decision maker’s options, are expressed in the set of outcomes for a proposition which is the consequent of an informational influence. The fact that a proposition has alternative outcomes and is the consequent of an informational influence defines it as a decision proposition. State descriptions consist of the set of facts and probabilities expressed within or deducible from a domain description. Relationships between states are expressed by the various types of influences available in the language; the logic, probabilistic, and informational influences expressed for the domain. Preferences are handled by identification of a particular proposition whose outcome incorporate the decision maker’s objectives. A real valued variable in the proposition is identified as the objective — i.e. the value to be maximized or minimized. A logical influence is defined which is capable of computing this value as a function of other propositions in the domain.

6 Example: A Decision Process

This section presents a simple example, using the decision language to describe a specific subproblem in a decision domain. Consider a security trader dealing in a single instrument, perhaps a particular Treasury security issue or foreign currency. The dealers’ task is to trade continually in the instrument in order to make a profit. The traders decisions are what quantity of the security to buy or sell at each instant of the trading day. The fundamental strategy is to ‘buy low, sell high,’ which is considerably easier to write down than to execute. The dealer’s primary uncertainty is that the price of the security will be in the future. Changes in the price are dynamic and dependent on the price in previous periods as well as some other economic conditions or market factors. The trader wishes to maximize his expected profit at some terminal time (Cohen et al. 1985)

The following basic decision alternatives represent the trader’s decision to buy, sell, or do nothing (hold) in each trading period. The set of propositions for this situation is shown below along with an interpretation for each. Alternative values for restricted variables are shown in brackets {}. These propositions constitute the means of expressing state descriptions for this domain:

(PROFIT profit time)	Trader’s net profit. This is the cumulative total of all the trader’s gains and losses in terms of profits since trading was initiated.
(POSITION value time)	Trader’s net holding of the security. This is the cumulative total of all the trader’s sales and purchases in terms of units of the security.
(TRADE {BUY SELL HOLD} time)	Trader’s decision alternatives.

(PRICE {90 91 92} time)	Range of security prices. This is a restriction on the assumed range of prices that the instrument can adopt.
(FUTURES-EXPIRE time)	Futures contract expiration. Futures are contracts for the delivery of a given security at a future date. Standard security future contracts expire on a predetermined date (e.g. the 3rd Friday in March, June, September, etc.). This proposition is true if 'time' occurs on a date when futures contracts mature.
(FUTURES-VOLUME	Indicator of activity level for futures markets.
{HEAVY MODERATE} time)	The level of activity in futures affects the levels of activity and prices in the 'cash' market (i.e. for current delivery) that is considered in this example.
(GURU{BULLISH BEARISH}	Forecast by a market prognosticator or analyst.
time)	This represents the information of some outside expert. The 'guru' is 'bullish' if he believes prices are likely to rise, and 'bearish' if prices are thought to fall.

We now describe the set of relationships which characterizes this domain.

The trader's profit and position are simply accounting relations, expressed as deterministic influences. We assume an initial position of zero units of the security, an initial profit of zero dollars, and a single trade quantity of 100 units. The facts (PROFIT 0.0 0)← and (POSITION 0.0 0)← indicate the trader starts with no holdings and no profit.

The net position of the trader is the difference between total sales and total purchases by the trader and depends on the trade made in the current period and net holdings in the previous period.

(POSITION new-position time) ←(– time 1 last-time) ∧
(POSITION old-position last-time) ∧
(TRADE BUY time) ∧
(+old-position 100 new-position)

(POSITION new-position time) ← (– time 1 last-time) ∧
(POSITION old-position last-time) ∧
(TRADE SELL time) ∧
(– old-position 100 new-position)

(POSITION old-position time) ← (– time 1 last-time) ∧
(POSITION old-position last-time) ∧
(TRADE HOLD time)

The profit level at any time is composed of the profit the trader has accumulated so far, plus an adjustment for the amount of the security the trader is holding (net position). If we identify maximizing profit as the objective of the trader, then his preferences among various outcomes (for PRICE, PROFIT, and POSITION) in terms of other propositions are expressed by the logic influence:

(PROFIT new-profit time) ← (— time 1 last-time) ∧
(PROFIT old-profit last-time) ∧
(PRICE old-price last-time) ∧
(PRICE price time) ∧
(POSITION old-position last-time) ∧
(— price old-price change-in-price) ∧

(*old-position change-in-price change-in-value) Λ
(+ old-profit change-in-value new-profit)

The PRICE proposition is the uncertain proposition in this example. The conditional probability distribution for PRICE is expressed by a series of probabilistic influences. A simple influence is:

(PRICE new-price time)$|_p$ (— time 1 last-time) Λ
(PRICE old-price last-time)

$= \pi_p(\omega_{\text{(PRICE new-price time)}} \mid \omega_{\text{(PRICE old-price last-time)}})$

This influence expresses a simple stochastic update of price given the previous periods price. If futures contracts for the security expire in a particular period, then the price is independent of the old price and is expressed by a different distribution:

(PRICE new-price time)$|_p$ (FUTURES-EXPIRE time) Λ
(FUTURES-ACTIVITY level time)

$= \pi_p(\omega_{\text{(PRICE new-price time)}} \mid \omega_{\text{(FUTURES-ACTIVITY level time)}})$

Note that since (FUTURES-EXPIRE time) is not restricted, the conditional probability distribution π_p will not have separate entries for alternative outcomes of (FUTURES-EXPIRE time), therefore the distribution can be written solely in terms of the alternative outcomes for (FUTURES-ACTIVITY level time). The term (FUTURES-EXPIRE time) is a condition that must be true for the conditional probability distribution $\pi_p()$ to be applicable. Note that the representation allows the expression of several conditional distributions simultaneously, and allows for the expression of conditions regarding which of the alternatives is appropriate by interweaving of deterministic information (e.g. FUTURES-EXPIRE) and uncertain outcomes (e.g. PRICE).

The trader also has information available from the market analyst (the 'GURU') regarding his view of the market is being bullish or bearish. The guru therefore provides the trader with an indicator of overall market trends. The traders opinion of this expert is expressed in the following influence:

(GURU assessment time)$|_p$ (+time 1 next-time)Λ
(PRICE price time) Λ
(PRICE next-price next-time)

$= \pi_p(\omega_{\text{(GURU assessment time)}} \mid \omega_{\text{(PRICE price time)}} \Lambda_{\text{(PRICE next-price next-time)}})$

This influence expresses the trader's probability distribution with respect to the guru's forcast for each possible set of prices for the current and subsequent period (i.e. the alternative outcomes of (PRICE price time)Λ(PRICE next-price next-time)).

The final component indicates the decision in this domain and the information available. The influence states that at time 2 the trader knows the price and guru assessment.

(TRADE action 2)$|_i$ (GURU assessment 2)Λ(PRICE price 2)

For all periods, the trader knows the price when making a trade.

(TRADE action time)$|_i$ (PRICE price time)

These other facts and priors define the state of the knowledge base at a given time. For example:

(PRICE price 0)$|_p \equiv \pi_p(\omega_{\text{(PRICE price 0)}})$
(FUTURES-ACTIVITY level time)$|_p \equiv \pi_p(\omega_{\text{(FUTURES-ACTIVITY level time)}})$

(FUTURES-EXPIRE 2)←

The first two statements express prior probability distribution for prices at time zero, and futures activity for any period respectively. The last is a fact expressing that futures expire in period 2.

7 Algorithms for ML on Decision Trees

Turning now toward algorithms on machine learning for decision trees in view of Markov or Bayesian networks we could further explore the structure as laid down in the previous sections. ML resembles neural networks for supervised or unsupervised (deep) learning. The decision tree would find the most similar training instances by a sequence of tests on different input attributes. The tree is composed of decision nodes and leaves , starting from the root, each decision node applies a splitting test to the input and depending on the outcome we take one alternating on the branches. Tree learning is nonparametric – the tree grows as needed and its size depends on the complexity of the problem underlying the data for a simple task, the tree is small, whereas a difficult task may grow a large tree. There are different decision tree models and learning algorithms depending on the splitting test used in the decision nodes and the interpolation done at the leaves: one very popular approach nowadays is the random forest.

Proceeding in several key steps reveals the structure of ML algorithms guided by training data in each of these steps.

(a) Starting with time-dependent changes in probability estimates ,i.e.,

$\pi(\omega_i | \omega_j) = \pi(\omega_i, \omega_j)/\pi(\omega_j)$ with time dependent conditionalization $\pi(t)(\omega_i | \omega_j) = \pi(t)(\omega_i, \omega_j)/\pi(t)(\omega_j)$ with $t = t_o < t < t^*$ and t^* maximum time limit. With chaining of events, as causal chains on multiple events form a network of things with a directed graph of several variables. Consider a distribution π on a Bayesian network of a conditionalized chain such that

$\pi(\omega_1, \ldots, \omega_n) = \pi(\omega_1), \pi(\omega_2 | \omega_l), \ldots, \pi(\omega_n | \omega_l, \ldots, \omega_{n-1})$

The joint probability is expressible in terms of a product of n functions.

(b) Expected utility $U = [\pi(t), u(\omega)]$.

In view of moving from probability estimates to calculating expected utility , learning about a random variable Ω can be decided before actually observing its value, i.e. let ω be the value of Ω then the utility of action a will will be

$U(a | \omega) = \sum_\omega u(\omega)\, \pi(\omega | a, \theta, \Omega = \omega)$

with $\omega|a$ as being consequence of action and θ as evidence or state. Choosing the best among the pending alternatives we get the value $U(\Omega=\omega) = \max_a u(a | \omega)$.

Since we are not sure about the actual outcome of Ω we must average $U(\Omega=\omega)$ over all possible values of Ω weighted by their probabilities. Thus, the utility over all Ω values is

$$U(\Omega) = \sum_\Omega \pi(\Omega=\omega|\theta)\, (U(\Omega=\omega)\ .$$

(c) Change of Preference. $u(x_t) > u(y_t) \Leftrightarrow x_t \, P \, y_t$ with x,y given prospects and P a strict preference relation. Inducing some minimal rationality on preferences of given prospects x,y , a weak utility preserving order subject to (i) shift parameter changes of tastes, (ii) tradeoff parameters, lexicographic vs. compensatory preferences, (iii) satisficing or computational constraints (bounded rationality).

(d) Risk Preference/Aversion Profile. Prior specification of utility function as concave (risk averse), convex (risk seeking) or linear (risk neutral) as impacting final decision outcomes.

Thus the outcomes change accordingly if either $U[\sum \pi(t)\ \omega] \ \geq\leq [\pi(t)\ u(\omega)]$.

(e) Basic Propagation. Dynamic learning would involve an interactive enccoding scheme based on some standardized assumptions: (i) all interactions between variables are linear, (ii)sources of uncertainty are normally (gaussian) distributed and serially uncorrelated, (iii) the influence diagram scheme as a causal network is singly connected. Each new piece of evidence can be viewed as a perturbation that propagates through the network via message- passing between neighboring processors.

Each variable Ω has a set of backward variables , say, X_1, X_2, … , X_n and a set of forward variables, say, Y_1, Y_2, … , Y_m. The relationship could be captured by the linear equation

$$\Omega = \ b_1 X_1 + b_2 X_2 + \dots ,+ X_n \ + \varepsilon_\Omega$$

ε_Ω being a stochastic residual component (noise) with zero mean and uncorrelated with any other noise variable.

Each processor Ω is sourced by the following set of parameters

(i) link coefficients installed for interaction between (a) – (d),

(ii) distribution of the random variables originating from Ω,

(iii) backward messages from roots of the tree network from each reverse link to Ω , $\pi_\Omega(x_i)$, I = 1, …, n ,

(iv) forward messages from descending variables of Ω , $\lambda_\Omega(y_j)$, j = 1,…,m .

8 Summary and Conclusions

We assert that the decision making process consists of a series of related activities and assessments with different types of reasoning. The representations and inference methods are designed to provide an integral set of methods for assisting decision makers in these activities and improving performance.

The techniques are grounded on the premise that for effective intelligent decision support, the representation of a decision situation in a computer must reflect the alternatives, beliefs, and preferences of the user of the system. Therefore, the approach developed here focuses on (1) the development of representations and techniques which construct a probabilistic or decision-theoretic model for a particular user, query, and state of information, and (2) support the exploration of alternative representations and models for various phenomena by the user.

Central to the development of powerful computer based decision aids is a means of expressing information and relationships important to describing a particular decision domain. We developed a language based on first order logic for the description of states, alternatives, beliefs, and preferences associated with a decision domain and decision maker. The language includes constructs for explicitly enumerating the alternative possible outcomes for uncertain propositions, probability distributions over these outcomes, the choices facing the decision maker, and his preferences regarding alternative outcomes of differing likelihood. The concept of a logic rule has been generalized to provide for the expression of conditional probabilities (i.e. the probability of a proposition over its alternative outcomes given some conjunctions of propositions is true) and of information availability at the time of decision.

Various deductive inference techniques for reasoning using the first order domain representation are described. The methods are grounded on the dynamic construction of a probabilistic or decision-theoretic model in response to a query and decision domain description in the representation language.

These techniques have the following characteristics which distinguish them from previous approaches:

- Probabilistic and decision theoretic reasoning are integrated with logical, deterministic inference.
- The techniques do not impose assumptions of conditional independence on the probabilistic representation.
- The control structure in the inference methods serves to minimize the size of a probabilistic representation.
- The approach is capable of constructing multiple models for the same phenomena. This enables reasoning about the performance and results of different models within the same environment.

Based on those rules we designed a machine learning algorithm guiding through principles of an IDSM. We could translate the sequential parts into proper codes through Python or Tensaflow.

Additional research areas relate to improving the performance and efficiency of the techniques as implemented. The current system relies on influence diagrams as a formalism for representing decision problems, and the algorithm developed by Shachter(1986) to solve for the optimal sequence of decisions. In the context of the Internet we could highlight intelligent decision procedures through IT platform management and data analytics (Gottinger, 2017).

REFERENCES

[1] Bush,R.R. and Mosteller, F.(1955),Stochastic Models for Learning, John Wiley: New York

[2] Cohen, P. R. (1985) Heuristic Reasoning about Uncertainty: An Artificial Intelligence Approach, Pitman, London.

[3] Fishburn, P. C. (1988) Nonlinear Preference and Utility Theory, The Johns Hopkins Univ. Press: Baltimore, MD.

[4] Gershman,S.J., Horvitz,E.J. and Tenenbaum,J.B. (2015), “Converging Rationality: A converging paradigm for intelligence in brains, minds and machines”, Science Magazine 349(6245), July 17, 273-278

[5] Gottinger,H.W and Weimann,P. (1990), Artificial Intelligence, A Tool for Industry and Management , Ellis Horwood: Chichester, Sussex, England

[6] Gottinger,H.W.(2017), Internet Economics: Models, Mechanisms and Management, Bentham Science: London

[7] Holtzman, S. (1989) Intelligent Decision Systems, Addison-Wesley: Reading, Mass.

[8] Howard, R. A. and Matheson, J. E. (1981) “Influence Diagrams”, 1981, In: Howard, R. A. and J. E. Matheson (eds), The Principles and Applications of Decision Analysis, SDG Publications, Strategic Decisions Group: Menlo Park, California, 1984.

[9] Parkes,D.C. and Wellman,M.P.(2015), “Economic Reasoning and Artificial Intelligence”, Science Magazine 349(6245), July 17, 273-278

[10] Pearl, J. (1988) Probabilistic Reasoning in Intelligent Systems, Morgan Kaufmann: Los Altos, Ca.

[11] Davis, R. (2016) Neural Networks and Deep Learning Explained, AWS: London

[12] Savage, L. J. (1954) The Foundations of Statistics, Wiley Publications: New York.

[13] Shachter, R. D. (1986) ‘Evaluating Influence Diagrams’, Operations Research 34.

SOCIETY FOR SCIENCE AND EDUCATION
UNITED KINGDOM

TMLAI TRANSACTIONS ON
MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE

Volume 6 No. 2
ISSN 2054-7390

Detailing Sentiment Analysis to Consider Entity Aspects: an Approach for Portuguese Short Texts

[1]José Saias, [2]Mário Mourão, [3]Eduardo Oliveira
[1]Universidade de Évora, Portugal;
[2] Decsis, Portugal;
[3]Digitmarket, Portugal;
jsaias@uevora.pt; mario.c.mourao@decsis.pt; eduardo.oliveira@bizdirect.pt

ABSTRACT

Sentiment analysis is useful for identifying trends, or for discovering user preferences, which can later be applied to campaign targeting or recommendations. In this paper, we describe an approach to classify the sentiment polarity regarding aspects, and how this technique was used in a previous system, for short texts in Portuguese, giving it greater sensitivity to detail.

Aspect extraction is done by locating candidates for aspect as expressions having a relationship with the entity and possibly some polarized term, through rules based on POS tags. For each aspect, the sentiment polarity is determined by a Maximum Entropy classifier, whose features depend on the entity mention, on the aspect and its support text, including negation detection, bigrams, POS tags, and sentiment lexicon-based polarity clues. For aspect sentiment, our classifier evaluation indicated a precision of 68% for the positive class and 73% for the negative class, with the dataset used in our research.

Keywords: Sentiment Analysis, Machine Learning, NLP, Text classification.

1 Introduction

Assessing the level of satisfaction, about a product or entity, is highly valued in marketing. Such information is important for those who decide the direction of a campaign, or to assist in brand or celebrity online image/reputation management, for example. The marketing sector is naturally aware of the phenomena of online social networks and studies the efficiency of advertising in its target users, and fine-tunes criteria for campaigns according to analysis on their data [1].

Sentiment Analysis (SA), sometimes referred to as Opinion Mining, deals with the search of opinion in text, automatically detecting evidences of sentiment with a certain polarity (positive, negative or neutral) [2]. These sentiments can be abstract, or not directed, evidencing something only about the state of mind of the author (E.g.: "*This work makes me sick!*"). They can also be concrete, and focused on a target entity, or some specific aspect of that target (e.g.: “*Restaurant X has fantastic food but prohibitive prices!*”).

SA can help at various levels. On one hand, facilitating the identification of trends, pointing out what everyone is liking (or not liking), or who/what is extremely popular (or unpopular), at a given day. In addition, SA can help to create user-specific models by identifying things he likes, his preferences, or aspects he does not like, important elements for tasks as client clustering or recommendation systems.

DOI: 10.14738/tmlai.62.4379
Publication Date: 12th April, 2018
URL: http://dx.doi.org/10.14738/tmlai.62.4379

This paper describes an approach for sensing the overall and the target oriented sentiment polarity over short texts from social media, in Portuguese. Our work is related to the SmartSeg project, which aims to create an intelligent targeting engine that can help marketers define target segments for campaigns. In addition to global polarity and polarity directed to an entity, a greater level of detail is now required, with the identification, whenever possible, of the target entity aspects on which an opinion is expressed. Our focus is an extension to an earlier SA system [3], adding aspect polarity classification, based on previous work by the first author on SA over English text [4, 5].

The following section mentions some recent related work. Section 3 describes how our system works, with more detail on how aspects are handled. Some preliminary results are given on Section 4, and the paper ends with remarks and conclusions in section 5.

2 Related Work

We can easily find a study on SA if we do a web search. However, finding an SA system able to detect different aspects of the same entity and classify their corresponding sentiment, is not easy, and even more if the goal is to work with Portuguese written content.

A doctoral thesis by Nádia Silva [6], in 2016, reports a study on the use of classifier ensembles to SA, feature engineering, and approaches to semi-supervised learning to overall polarity classification. On the Kaggle platform, we can find systems for SA, such as the one Leandro Silva developed for Brazilian Portuguese text [7], described with excerpts of code and pedagogical explanations on how to use NLTK, or a Bag-of-Words model for supervised learning, with Naive Bayes Multinomial and Random Forest classifiers on SkLearn. This is a demonstration purpose system, thought only for overall sentiment, not considering the different target entities. An empirical study of techniques for aspect extraction on SA is presented in Pedro Balage Filho's doctoral thesis [8], exploring three different approaches over English and Portuguese corpus.

Repustate [9] is a paid online service for text analytics, including SA for Portuguese texts. Its site mentions over 100 billion documents analyzed every month. Portuguese-specific tools are employed, including a part of speech tagger, a lemmatizer, and Portuguese-specific sentiment models. As far as we know, and as seen in the free demo, this system performs overall polarity classification only.

Conferences related to text processing often have challenges for SA. SemEval-2017 had two of these challenges: Task 4 and Task 5. The purpose in Task 4 was to identify the overall sentiment of a tweet, and the sentiment towards a topic with classification on a two-point and on a five-point polarity scale [10], with English and Arabic languages. For overall sentiment on English language, some teams used Deep Learning and neural network methods such as CNN and LSTM networks. Other supervised learning methods were also used, such as SVM, Maximum Entropy, Logistic Regression, and Random Forest based classifiers. Some systems combined SVM with neural networks. Five of the top-10 scoring systems used ensemble classifiers.

SemEval-2017 Task 5, or Fine-Grained Sentiment Analysis on Financial Microblogs and News, was a challenge that aimed to predict a sentiment score for each company/stock mentioned on text [11]. It was organized into separate subtasks, each with its texts collection: microblogs and news. Most systems used hybrid techniques, with Machine Learning and Lexicon-based approaches (as Jiang et al.'s system [12], which ranked first on subtrack 1), while others used a traditional Machine Learning approach, or a Deep Learning approach possibly combined with lexicon-based techniques.

3 System description

In our previous system [3], we used a supervised Machine Learning classifier for both overall and target oriented sentiment polarity, having a different configuration in each case. This paper describes an extension to that system, which aims to consider the particular entity's aspect about which an opinion is being expressed, and to improve the performance of the sentiment polarity classification.

3.1 Overview

Having as a priority the system sensitivity to aspects, we kept the previous underlying platform and preprocessing stages, coded in Java and MALLET [13], a Machine Learning for Language Toolkit. A REST API is used by the service clients, allowing them to send their requests with short texts, such as tweets or comments on social media. The text received in each request is processed according to the following sequence of steps:

1. Preprocessing: Natural Language Processing techniques, which include the division into sentences and a first level of syntactic analysis
2. Named entity recognition: to identify mentions to persons, brands and other opinion targets
3. Overall sentiment polarity classification: opinion polarity broadly conveyed by the text, even if there is no mention of any entity
4. Target oriented sentiment analysis
    a. Aspect detection for target entities
    b. Aspect sentiment polarity classification

Figure 1 shows the system pipeline outcome for the sentence "*Évora é bonita*." (In English: "*Évora is beautiful*."). It is a JSON structure where we can see the identified aspect "*beleza*" (beauty) and its sentiment polarity. This output can be formatted in JSON or XML.


```
{
 "id" : "2017122709511212",
 "overallPolarity" : 1.0,
 "targetCount" : 1,
 "targetPolarityList" : [ {
   "target" : "Évora",
   "polarity" : 0.7,
   "countPositiveRefs" : 1,
   "countNeutralRefs" : 0,
   "countNegativeRefs" : 0,
   "targetReferencesOverDoc" : [ {
     "aspect" : "beleza",
     "from" : 0,
     "to" : 5,
     "sentenceNumber" : 0,
     "aspectSupportTextTo" : 14,
     "aspectSupportTextFrom" : 6,
     "referencePolarity" : 0.7
   } ]
 } ]
}
```


Figure 1: System output for "Évora é bonita."

Step 4 of the pipeline, for target oriented SA, was completely reimplemented, and it is described in the following sections. The first three steps (preprocessing, target detection and overall sentiment) are processed as described in [3]. Preprocessing includes noise removal, tokenization, POS tagging and lemmatization.

The target detection comprises a mixed approach with an OpenNLP classifier for Named Entity Recognition (NER), and an entity catalog. The first has a model trained for Portuguese, and from its outcome, we filter entity categories, by choosing Person, Organization, Brand and Location, while discarding others (currency, time, numeric and abstract). The entity catalog is a lookup table whose entries have the entity canonical name, possible name aliases, and the entity type.

To detect the overall sentiment, we continue to use a supervised Machine Learning approach, in MALLET and a Maximum Entropy classifier. The training set has about 71000 labeled instances, with short texts of posts, tweets or comments on the web about TV celebrities, sports, music festivals and politics.

The overall sentiment classifier model is trained with features for: lemmas, (POS+polarity) pairs for each term, bigrams before/after polarized terms, lemmatized bigrams after verbs and after negation terms, presence of polarized terms in the last 5 tokens. To detect polarized terms or clues, in feature extraction, we use SentiLex-PT [14], a sentiment lexicon for Portuguese, and a complementary polarity table which aggregates a set of Portuguese expressions, including idioms, but also popular English expressions, Internet jargon, and common symbols and abbreviations.

3.2 Aspect based SA

At this point, the target entities have already been tagged throughout the text. In the previous version, we would extract features about the entity mention surrounding text, and, based on those features, deduce the sentiment polarity. This is where we change the procedure to be more meticulous, and to consider entity aspects.

3.2.1 Aspect Detection

As reported by Bing Liu [15], a document where the author gives a positive opinion about an entity or object does not mean that the author always has a positive opinion on all the characteristics of that entity. An aspect is a feature of the target entity. It is something particular about that entity, about which there may be an opinion in the text. For the same target entity, we can have different aspects, each with its sentiment polarity. As an example, in a text relative to a restaurant, there may be a positive opinion about the food, and a negative opinion about the service. In this case we would have only one target entity, the restaurant, and two aspects: the food and the service. Treating polarity over the whole document or sentence may not capture this degree of detail.

We address the aspect extraction as an Information Extraction task, using a relation based method. In the entity's surrounding text, the system looks for expressions having a relationship with the entity, and possibly some polarized term. The relationship between the aspect support expression and the target entity can be identified with the help of a dependency parser and with rules based on the POS tags of neighbouring tokens.

The main extraction rules in our system have the following patterns:

i. TARGET VTER POL Noun
ii. Noun FROM TARGET VSER POL
iii. Noun POL FROM TARGET

iv. VSER POL ART Noun FROM TARGET

v. TARGET VSER AQ

In each rule, TARGET is the mention to the entity, and Noun is the candidate aspect. POL is a polarized term, as an adjective with negative or positive sentiment. VTER is a verb semantically compatible with "*ter*" or "*possuir*" (to have). VSER is a verb semantically compatible with "*ser*" or "*estar*" (to be). FROM represents a preposition, such as "*de*", "*da*", and "*do*". ART is a definite article. AQ stands for aspect qualifier, and it is a term or expression that defines the aspect. To detect these, we use a fixed table, whose entries have triples "VERB AQ ASPECT". The example "*ser,barato,preço*" (to be,cheap,price) is one of these triples, which means that if you use the term "cheap" you are referring to "price" aspect.

All the five rules have variants: optional presence of articles, switching the position of POL and Noun, presence of intensity expressions (e.g. adverbs). Another variant is the extraction of aspects in sequence, for texts that describe several attributes of an entity in a single sentence.

Let us consider a concrete example:

> "*Este Nokia tem um display fantástico mas não tem uma boa bateria*." (In English: "*This Nokia has a fantastic display but does not have a good battery*.")

At this stage, the system will extract the aspects "*display*" and "*bateria*" (battery) related to entity mention "*Nokia*". With each aspect, the system also maintains its support text. The intent is to keep the surrounding text including a possible polarized expression.

If near the entity mention we do not find any aspect, then we assume the "general" default aspect, meaning that the opinion is towards the target entity as a whole.

3.2.2 Aspect Sentiment Polarity

In this step, the system will determine the sentiment polarity (positive, negative or neutral) for each aspect found for all detected target entities. In cases where the aspect is global, that is, a more specific aspect has not been extracted, polarity classification is done towards the target entity mention, considering its location in the text, as done in the previous version of the system. In more specific cases, with a concrete aspect, the polarity is focused on that aspect rather than on the entity, and so the system considers the aspect support text first, and then the remaining text.

Sentiment polarity is determined by a Maximum Entropy classifier, whose supervised learning model was now trained with 13600 instances. In each instance, we now needed tags with the target entity, the location of the mention to the target entity, the aspect and the aspect support text, and the sentiment polarity regarding the aspect. As we did not have all this detail in the previous model's training instances, it was necessary to reach a compromise solution, that would allow to take advantage of what already existed, and to add new instances, with the additional annotation tags.

For the first part, backward compatibility, some corrections were made on the previous instances, and some were eliminated because they did not fit properly. In these instances, without aspect annotation, the system assumes the "general" aspect, and therefore trains these cases with the polarity directed to the entity as a whole.

To get new instances, with aspects annotation, we used the outputs from *PolarityGame* [16], a system for building an annotated corpus. That system is based on gamification [17], which is the adoption of game techniques and dynamics in processes that do not have this nature. Such may encourage users to try a system, and instill elements of reward or competition, as a stimulus for the user to continue their participation. Tagging to SA is a time consuming process, so it is not easy to find volunteers. In this way, the use of gamification is an effort to capture some contributions to the annotation task. A *PolarityGame* session includes: insert a short text; tagging it; and validate or fix system tags for another existing instance that is chosen by the system. The scoring scheme of the game is based on the estimated contribution to the system. By contribution, we mean the insertion and tagging of a text that effectively adds value to the knowledge base, and that is supported by cross-validation.

To isolate the polarity of each aspect, before extracting the features, the system defines the classification chosen text, which consists of the cancellation of the remaining aspects. If there are multiple aspects for the same entity, the other aspects' support text is replaced by white space. The aspect polarity classifier's model is built with the following features:

- Bag-of-Words for each token inside the chosen text, and replacing the entity name by TARGET, to simplify pattern processing and be transparent to the variability of names.
- Lemma bigrams for tokens within the chosen text.
- Syntactic function associated with the target. When possible, indicate whether the target appears in the subject or object, according to sentence structure.
- Syntactic function associated with the aspect term.
- Subject/object polarity. As before, if a sentiment lexicon determines the polarity that some expression originates, on the subject or on the object part, we create a feature for it.
- Lemma bigrams, after and before the target mention.
- Lemma bigrams, after and before the aspect term.
- Lemma bigrams, after and before the aspect support text.
- Pairs and bigrams and trigrams of (POS tag, simple polarity) pairs, as before, for the chosen text.
- Bigrams before/after polarized expressions, as before, across the chosen text.
- Presence of negation terms before the target mention, and before the aspect support text.

In cases where a target entity has more than one aspect, the polarity assigned to the entity is the sum of the polarities of its aspects.

In Figure 2 we can see the detailed output returned by the system for the previous example. In this JSON data we can find an entity "*Nokia*", and the two aspects associated with it: "*display*" and "*bateria*". The aspect support text offset is also set in each case. As we can see in this example, the "*display*" aspect has a positive polarity, while for the "*bateria*" aspect the sentiment is negative.

4 Results

The training set has a class distribution as indicated in Table 1. For both the classifiers, most of the annotated instances belong to the negative class. As in the previous version, there is still a large imbalance between the positive and negative classes.

To evaluate the impact of this system's change with the approach for aspect support, we performed an evaluation similar to the previous one. We used a 10-fold evaluation, which means that each training round has 90% of the instances. The labeled instances are partitioned into 10 subsets. Then there are 10 rounds of evaluation, in which, and in turn, each of the 10 instance sets is used to test the classifier trained with the remaining 9 sets.


```
{
  "id" : "sample0031",
  "overallPolarity" : 0.08,
  "targetCount" : 1,
  "targetPolarityList" : [
   {
    "target" : "Nokia",
    "polarity" : 0.07,
    "targetReferencesOverDoc" : [ {
      "aspect" : "display",
      "from" : 5,
      "to" : 10,
      "sentenceNumber" : 0,
      "aspectSupportTextTo" : 36,
      "aspectSupportTextFrom" : 18,
      "referencePolarity" : 0.78
    }, {
      "aspect" : "bateria",
      "from" : 5,
      "to" : 10,
      "sentenceNumber" : 0,
      "aspectSupportTextTo" : 64,
      "aspectSupportTextFrom" : 53,
      "referencePolarity" : -0.71
    } ],
    "countPositiveRefs" : 1,
    "countNegativeRefs" : 1,
    "countNeutralRefs" : 0
  } ]
}
```


Figure 2: Sample output with two aspects for the same entity

Table 2 has the results of the evaluation, for both overall and target+aspect oriented SA, with precision, recall and F-measure metrics per class. As expected, the overall polarity classifier has the best performance, with high values for F1, as seen in the system's previous version. The results of the new aspect based polarity classifier are shown in the second half of the table. The positive class has the lowest values in all three metrics. This already happened in the previous system, but there was a 1% increase in precision, recall and F1, perhaps by the presence of new/more instances in the dataset. In the opposite direction, precision fell in both the negative class and the neutral class by approximately 2%, reaching now 73% on the negative class, and 75% on the neutral class. The best F1 measure in target oriented was obtained in the neutral class with 76%.

Table 1: Polarity class weight in the training instances

Analysis	positive	negative	neutral
overall	19%	72%	9%
target+aspect oriented	22%	41%	37%

Table 2: Sentiment polarity classifier evaluation

Sentiment	class	precision	recall	f-measure
overall	positive	0.96	0.96	0.96
	negative	0.98	0.98	0.98
	neutral	0.78	0.77	0.78
target+aspect oriented	positive	0.68	0.65	0.66
	negative	0.73	0.75	0.74
	neutral	0.75	0.78	0.76

5 Conclusion

We described a sentiment analysis system for short texts in Portuguese, and how it was improved to deal with entity aspects. Considering weight of each class in the target oriented training set, we can estimate a weighted average of the precision at 72.6%. This is roughly the same precision as the previous homologous classifier, but with the important difference of granularity in detail, which is now richer, having indication of polarity per each aspect of the target entity.

The evaluation presented for target oriented in section 4 refers to the appreciation of the sentiment classifier, on a large number of instances without aspects (“general” aspect) and approximately 1000 instances with a tagged aspect and its polarity. We believe that the number of instances with aspects is still reduced.

For future developments, we have many lines of work. We intend to collect more instances with aspects for the training set. The aspect extraction module needs a specific evaluation, which in this work did not happen. Still regarding aspect extraction, we can restrict more, limiting aspects to a well-known set for each type of entity, or be more permissive, allowing the discovery of unknown/unseen aspects, but adding the risk of increasing the error with false positives.

In future we also plan to experiment with other classification techniques such as deep neural networks. We did not do this already because the first experiences did not give good indicators. As we can see in the ranking of SemEval-Task 5’s participating systems, the use of Deep Learning does not automatically mean better results. See the case of the winning system for the microblog messages subtask, which was ahead of other systems that used Deep Learning methods. In our case, with few training instances, we have chosen to stabilize a system based on conventional Machine Learning. Later, we can appreciate the possible benefit of Deep Learning for this dataset, now having a reference for comparison.

ACKNOWLEDGMENTS

This research is partially supported by the SmartSeg project, which is co-funded through Portugal 2020’s "R&D Incentive System – Individual Projects" program, grant number "POCI-01-0247-FEDER-011192".

REFERENCES

[1] Catherine E. Tucker (2014) Social Networks, Personalized Advertising, and Privacy Controls. Journal of Marketing Research: October 2014, Vol. 51, No. 5, pp. 546-562.

[2] Pang, Bo & Lee, Lillian (2008). Opinion mining and sentiment analysis. Foundations and Trends in Information Retrieval, Vol. 2(1-2), pp. 1-135.

[3] J. Saias, R. Silva, E. Oliveira, and R. Ruiz (2015). "Combining overall and target oriented sentiment analysis over portuguese text from social media," Transactions on Machine Learning and Artificial Intelligence, vol. 3, pp. 46–55, June 2015.

[4] José Saias (2015). Sentiue: Target and Aspect based Sentiment Analysis in SemEval-2015 Task 12. In Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), Denver, Colorado, USA. June 2015. p. 767-771, ACL

[5] E. Dovdon and J. Saias (2017). "ej-sa-2017 at semeval-2017 task 4: Experiments for target oriented sentiment analysis in twitter," in Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), (Vancouver, Canada), pp. 635–638, Association for Computational Linguistics

[6] Silva, N. F. F. (2016). Análise de sentimentos em textos curtos provenientes de redes sociais. Tese de Doutorado, Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos. Brasil.

[7] Silva, L. (2018) Análise de Sentimentos em língua portuguesa do Brasil. Last accessed January 2018, at: https://www.kaggle.com/leandrodoze/sentiment-analysis-in-portuguese

[8] Balage Filho, P. P. (2017). Aspect extraction in sentiment analysis for portuguese language. Tese de Doutorado, Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, Brasil.

[9] Repustate. Sentiment analysis, social media sentiment and text analytics. Last accessed February 2018, at: https://www.repustate.com/portuguese-sentiment-analysis/

[10] S. Rosenthal, N. Farra and P. Nakov (2017). SemEval-2017 Task 4: Sentiment Analysis in Twitter. In Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017). Pages: 502–518. ACL, 2017.

[11] K. Cortis et al. (2017). SemEval-2017 Task 5: Fine-Grained Sentiment Analysis on Financial Microblogs and News. In Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017). Pages: 519–535. ACL, 2017.

[12] Mengxiao Jiang, et al. (2017). Ecnu at semeval-2017 task 5: An ensemble of regression algorithms with effective features for fine-grained sentiment analysis in financial domain. In Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval2017). Association for Computational Linguistics, Vancouver, Canada.

[13] McCallum, Andrew Kachites (2002). "MALLET: A Machine Learning for Language Toolkit". http://mallet.cs.umass.edu

[14] M. J. Silva et al., Building a Sentiment Lexicon for Social Judgement Mining. In Lecture Notes in Computer Science (LNCS) / Lecture Notes in Artificial Intelligence (LNAI), International Conference on Computational Processing of Portuguese (PROPOR), Coimbra, 2012.

[15] Bing Liu (2010). Sentiment analysis and subjectivity. In Handbook of Natural Language Processing, 2nd Edition. Taylor and Francis Group, Boca, 2010.

[16] José Saias (2017). Ludificação: experiências para construção e marcação de um corpus para Análise de Sentimentos. In I Congresso Luso-Extremadurense de Ciências e Tecnologia, Universidade de Évora. Outubro de 2017. ISBN: 978-989-8550-45-3

[17] Kim, B. (2015). Understanding Gamification. Library Technology Reports, 51(2), 1–35

SOCIETY FOR SCIENCE AND EDUCATION
UNITED KINGDOM

TMLAI TRANSACTIONS ON
MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE

Volume 6 No. 2
ISSN 2054-7390

Expert CF: Solving Data Matrix Sparsity and Computation Complexity Problems

Gangmin Li and Minghuang Chi
Department of Computer Science and Software Engineering
Xi'an Jiaotong-Liverpool University
Gangmin.li@xjtlu.edu.cn

ABSTRACT

Collaborative Filtering (CF) is widely used to provide recommendations in ecommerce systems. CF works on a large data set by constructing an item-user matrix through association analyses among items and similarity analyses among users. However, CF suffers from data sparsity and computation complexity. This paper introduces a concept of "experts" to overcome the two identified problems. An expert is an artificially created user, who represents a cluster of users in terms of behavior and taste. The construction of experts can be done off-line through data filtering and classification. In actual recommendation, when data are spars, a number of experts can be added as existing users to predicate shopping habit for a particular user. The mechanism of "off-line expert's construction" and "on-line expert's addition" in recommendation not only to overcome the data sparsity but also improving on scalability by reduce computation in recommendation phase.

Keywords: Recommendation, Collaborative Filtering, Artificial Intelligence, Experts, Data Matrix Completion

1 Introduction

Recommendation systems developed for commerce have been one of the great successes in Big Data application [1, 2]. Recommendation systems use input data about a customer's interests to generate a list of recommendations. The key problem associated with the recommendation system is to define customer's interests. Many applications use only the items that customers purchase and explicitly rate to represent their interests, but they can also use other attributes, including items viewed, demographic data, subject interests, and favorite artists etc. It is generally belief that recommendations that are more precise can be made if available data are increasingly large. As consequences, that larger and larger consumer's data are collected. Confronting such mass data, cloud computing and big data analytics methods provide e-commerce makers an alternative to large-scale data storage and HPC facilities [4, 5]. Collaborative Filtering used in the most popular recommender systems and has proved successful [3]. It offers an idea to recommend items to user from other similar users profiled together whose are found similar taste by defining their distance [6]. Distance definition among items and users suffers from data sparsity [7, 8] and cold-start problem [10], where a new item or user's entry does not have enough data to be used to make any valid recommendations [7].

DOI: 10.14738/tmlai.62.4337
Publication Date: 16th April, 2018
URL: http://dx.doi.org/10.14738/tmlai.62.4337

2 CF and EXPERTS

Proposed expert is built based on existing Collaborative Filtering (CF).

2.1 Collaborative Filtering (CF)

CF approach formulates problem as a large-scale item-user matrix, let us denote it as X and illustrated in Figure 1.

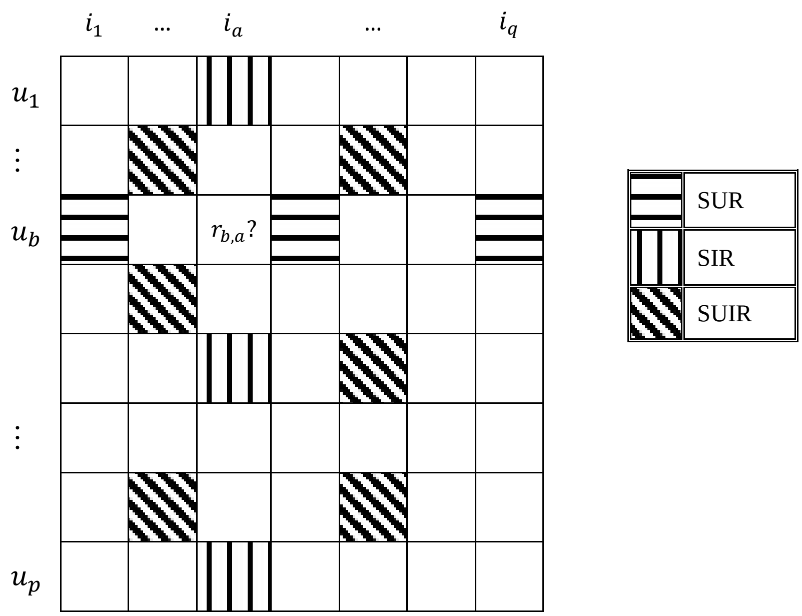


Figure 1. Traditional Collaborative Filtering Approach

As Figure 1 illustrates, user files are represented as an $Q \times P$ item-user matrix X, where Q, P are the sizes of items and users. CF intents to predict $r_{b,a}$, which is active item i_a made by active user u_b through finding other active users correlated to u_b and other active items (ratings made by active users with other existing ratings) correlated to i_a [8]. It would be easily realized when item-user matrix is filled. The two problem associated with CF are data sparsity and computation complexity.

Conversely, it is difficult to match user activities (ratings) in such enormous item set. It is also difficult to finding similarity between item sets (through item ratings) because that the rating data matrix is quite sparse. This problem becomes even serious when a user does not have much activities data at all, which is cold-start problem.

In order to formally describe the CF, we adopt a set of notations. Let

- *I* = $\{i_1, i_2, \dots, i_Q\}$ and *U* = $\{u_1, u_2, \dots, u_P\}$ be the sets of items and users in X,
- $\{C_u^1, C_u^2, \dots, C_u^L\}$ be *L* user clusters, and users in each cluster share some similar tastes (i.e., rate similar items in a similar way),
- *I*{*u*}, *I*$\{C_u\}$ and *U*{*i*} be the set of items rated by user *u*, the set of items rated by user cluster *Cu*, and the set of users who have rated item *i*,

- r_{u_b,i_a} denote the score that user u_b rates item i_a, $\overline{r_{i_a}}$ and $\overline{r_{u_b}}$ represent the average ratings of item i_a and user u_b,
- SI, SU and SUI be the sets of the similar items, like-minded users, and similar items and like-minded users,
- SIR, SUR and SUIR (see figure 1) denote predicting user interest over the entire item-user matrix from the ratings of the same user make on the similar items, the like-minded users make on the same item, and the like-minded users make on the similar items, i.e., SIR, SUR and SUIR predict unrated items for active users based on SI, SU and SUI, respectively.
- SR represent predicting user interest from all the ratings, i.e., SIR, SUR and SUIR,
- SIR′, SUR′, SUIR′ and SR′ be the counterparts of SIR, SUR, SUIR and SR, but they are calculated over the experts added item-user matrix X′.

Then, the item vector of the matrix X is:

$$X_i = \ [i_1, i_2, \dots, i_Q],\ i_q == [r_{1,q}, r_{2,q}, \dots, r_{P,q}]^T, \tag{1}$$

where q $\in [1,\ \mathrm{Q}]$. Each column vector i_m corresponds to the ratings of a particular item *m* by *P* users. Matrix X can also be represented by user vectors illustrated as:

$$X_u = [u_1, u_2, \dots, u_P]^T, u_P \ = [r_{p,1}, r_{p,2}, \dots, r_{P,Q}]^T, \tag{2}$$

where $p\ \in [1,\ P]$. Each row vector $u_P{}^T$ indicates a user profile that represents a particular user's item ratings.

Traditional CF can provide item-based CF, user-based CF and combination of both.

Item-based CF, represented as SIR in Figure 1, finds similar items among item vectors and then use their ratings made by the same user to predict the user's rating for new item. Given an active item i_a and a user u_b, Eq. 4 denotes the mechanism of item-based CF, where $sim_{i_a,\, i_c}$ is the similarity of items i_a and i_c that can be computed by any similarity calculation. We use Pearson Correlation Coefficient (PCC) as shown in Eq 3.

$$sim_{i_a,i_b} = \frac{\sum_{u\in U}(r_{u,\, i_a} - \overline{r_{i_a}}) \cdot (r_{u,\, i_b} - \overline{r_{i_b}})}{\sqrt{\sum_{u\in U}\left(r_{u,\, i_a} - \overline{r_{i_a}}\right)^2} \cdot \sqrt{\sum_{u\in U}\left(r_{u,\, i_b} - \overline{r_{i_b}}\right)^2}} \tag{3}$$

The SIR can then be expressed in Eq. 4 as follows,

$$SIR\colon r_{u_b,i_a} = \frac{\sum_{i_c \in SI} sim_{i_a,i_c} \cdot r_{u_b,i_c}}{\sum_{i_c \in SI} sim_{i_a,i_c}} \tag{4}$$

User-based CF, represented as SUR in Figure 1, predicts user's rating based on the ratings of like-minded users made on the active items. Eq. 5 shows the user-based rating prediction, where $sim_{u_b,u_c}$is the similarity of users u_b and u_c. Similarly, a number of similarity algorithms can be used to calculate similarity between two users.

$$SUR: r_{u_b,i_a} = \frac{\sum_{u_c \in SU} sim_{u_b,u_c} \cdot r_{u_c,i_a}}{\sum_{u_c \in SU} sim_{u_b,u_c}}. \quad (5)$$

Above the sim_{u_a,u_b} is the similarity between user u_a and u_b. It can be obtained by using PCC as shown in Eq. 6,

$$sim_{u_a,u_b} = \frac{\sum_{i \in I}(r_{u_a,i} - \overline{r_{u_a}}) \cdot (r_{u_b,i} - \overline{r_{u_b}})}{\sqrt{\sum_{i \in I}\left(r_{u_a,i} - \overline{r_{u_a}}\right)^2} \cdot \sqrt{\sum_{i \in I}\left(r_{u_b,i} - \overline{r_{u_b}}\right)^2}} \quad (6)$$

It is obvious that search active users and items through entire item-user matrix, which is needed for both calculate item and user similarity, is computationally expensive and time consuming. Its scalability is seriously in doubt.

2.2 Expert Generation

To solve data sparsity, concept of expert opinion was introduced by Amatriain et. al. [9]. It provides extra information, which is used to predict the behavior of the general population. In our research, we solidified the expert opinion by define experts as filtered users who are qualified to act as expert in the user matrix X_u must satisfy:

1. **Rating numbers exceed a threshold ρ.** Sufficient data samples are necessary to predict the general population. Expert set needs a general coverage on the item set, that is, less sparse and more even distributed than other users in the item-user matrix. i.e.

$$u_i \in \mathrm{U} \cap \mathrm{E} \Leftrightarrow \exists\, \rho \,\&\, u_i,\ Size(I\{u_i\}) \geq \rho\ , \quad \textbf{Con}. 1$$

   where E is set of expert.

2. **Ratings exclude individual bias.** Expert has a fixed marking scale and deviate, provided every item he or she rates fair, and deviate less than all other user's. i.e.

$$u_i \in \mathrm{U} \cap \mathrm{E} \Leftrightarrow \sigma(r_{u_i}) < \sigma(r_U)\ , \quad \textbf{Con .2}$$

Constructed Expert based on the two conditions needs a number of steps. The first step is to cluster active users.

Clustering Users ($C_i \leftarrow U$). In this step, all users are assigned into clusters using spectral clustering. Of course, user matrix X_u is actually represented by the rating matrix (see Eq. 2). As the name suggests, spectral clustering makes use of the spectrum (it also refers to eigenvalues) of the similarity matrix of user similarity. The similarity can be obtained by using PCC on X_u. Therefore, user similarity matrix can be created. Denote $S \in R^{P \times P}$ since there are *P* users, Here Gaussian function is used to describe similarity in previously generated similarity matrix S. Note that similarity in S is represented in a simple version S_{ab}, denote value in a's row and b's column. It is obvious that $S_{ab} = S_{ba}$ and $S_{aa} = 1$.

$$S_{ab} = \exp(-\frac{\|u_a - u_b\|}{2\sigma x^2}) \quad (7)$$

where, $\|u_a - u_b\|$, is the distance between user u_a and u_b. It can be obtained from the similarity matrix by inverting similarity into dissimilarity, which is distance matrix. σ is a scaling parameter that controls how rapidly the similarity S_{ab} reduces with the distance between u_a and u_b. Then constructing diagonal degree matrix *D* and diagonal elements *d* defined as Eq. 7:

$$d_{aa} = \sum_{b=1}^{p} S(u_a, u_b). \quad (8)$$

And calculate normalized Laplasse matrix L as Eq. 8:

$$L = I - D^{-\frac{1}{2}} S D^{-\frac{1}{2}} \quad (9)$$

Its elements are given by,

$$L_{a,b} = \begin{cases} 1 & \text{if } a = b \text{ and } d_a \neq 0 \\ -\frac{1}{\sqrt{d(S_{u_a})d(S_{u_b})}} & \text{if } a \neq b \text{ and } u_a \text{ is adjacent to } u_b \\ 0 & \text{otherwise.} \end{cases} \quad (10)$$

Known that L has k zero-eigenvalues that are also the k smallest ones. Denote $V \in R^{p\times k}$ as their corresponding eigenvectors:

$$V = [\boldsymbol{v_1}, \boldsymbol{v_2}, \dots, \boldsymbol{v_k}] = D^{\frac{1}{2}}E, \quad (11)$$

where

$$E = \begin{bmatrix} e_1 & & \\ & \ddots & \\ & & e_k \end{bmatrix},$$

where $e_i, i = 1, \dots. k$ are vectors of all ones. Thus, one needs to find the first eigenvectors of L (i.e., eigenvectors corresponding to the *k* smallest eigenvalues). The result is presented a single column matrix, the rows indicates orthogonal points on the unit sphere and can be clustered by k-means. Once all users have been assigned into different clusters, we can represent each user cluster with its centroid $\bar{u}$. The centroid is the artificially generated expert. The next step is to choose required number of experts from similarity clusters.

Choose Experts. Once we have all active users clustered into different user clusters, then we would store the similarity for each user to user-cluster and it gives every user similar clusters in a descending order. These clusters are used for selecting the top K like-minded users and generating a much smaller matrix that can be accessed quickly.

Given an item set $I = I\{u_a\} \cap I\{C_{u'}\}$, define the similarity between user u_a and user cluster $C_{u'}$ as follows it is again PCC similarity measurement.

$$\boldsymbol{sim_{u_a,C_{u'}}} = \frac{\sum_{i\in \mathrm{I}} \triangle\, \boldsymbol{r_{C_{u'},i}} \cdot (\boldsymbol{r_{u_a,i}} - \overline{\boldsymbol{r_{u_a}}})}{\sqrt{\sum_{i\in \mathcal{I}} (\triangle\, \boldsymbol{r_{C_{u'},i}})^2} \cdot \sqrt{\sum_{i\in \mathcal{I}} (\boldsymbol{r_{u_a,i}} - \overline{\boldsymbol{r_{u_a}}})^2}} \tag{12}$$

2.3 Use of Expert

In the on-line phase, requests need to be processed and responded quickly. Having experts make data matrix can be dense, but search larger space still time consuming. Therefore, a locally-reduced item-user matrix can be created and used. A locally-reduced item-user matrix selects the most similar items and the most similar active user supplemented with the most similar experts.

Creating a Locally-Reduced $\boldsymbol{M} \times \boldsymbol{K}$ Item-User Matrix. Locally-Reduced $\boldsymbol{M} \times \boldsymbol{K}$ Item-User Matrix has M top similar items and N top like-minded users and E top like-mined experts. Here N + E = K. It is relatively easy to select M top similar items SI, Following Eq. 3 similarity between two items can be calculated and all the similarity can be sorted in a descending order. The top M similar items can be selected as reduced item set for the online matrix. Similarly, the top N can be selected from a previous build link-minded user set SU. So the user set of the online matrix is also reduced. If there are E empty cells in the user set to form M × K Item-User Matrix, K experts can be selected from the similar experts set.

Note that ratings of expert and like-minded user can be in different scale from the original rating style and a weight may be needed during practical recommendation. A parameter $\boldsymbol{\omega}$ defined as:

$$\omega\colon\ \omega_{u,i} = \begin{cases} \varepsilon & if\ u\ rates\ i \\ 1-\varepsilon & otherwise \end{cases}, \tag{13}$$

After the top M similar items and K like-minded users are selected, related ratings will be extracted to fill the locally-reduced item-user matrix from original item-user matrix. So predicts user's request and makes recommendation can be made based on experts.

2.4 Predicted User Rate with Expert Added

Both like-minded user and expert's rate can be used to predict a user made on the same item. Four types of ratings can be obtained, ratings from the same user made on the similar items, like-minded user made on the same item, like-minded users made on similar items and like-minded expert made on the same item. Denote as SIR'， SUR', $SUIR'$, and SER'. Given an active i_a and u_b,:

$$SIR' = \frac{\sum_{s=1}^{M} \omega \cdot sim_{i_s,i_a} \cdot r_{u_b,i_s}}{\sum_{s=1}^{M} \omega \cdot sim_{i_s,i_a}}$$
$$SUR' = \frac{\sum_{t=1}^{K} \omega \cdot sim_{u_t,u_b} \cdot (r_{u_t,i_a} - \overline{r_{u_t}})}{\sum_{t=1}^{K} \omega \cdot sim_{u_t,u_b}} + \overline{r_{u_b}}$$
$$SUIR' = \frac{\sum_{t=1}^{K}\sum_{s=1}^{M} \omega \cdot sim_{(i_s,i_a),(u_t,u_b)} \cdot r_{u,i}}{\sum_{t=1}^{K}\sum_{s=1}^{M} \omega \cdot sim_{(i_s,i_a),(u_t,u_b)}} \tag{14}$$
$$SER' = \frac{\sum_{t=1}^{E} \omega \cdot sim_{e_t,u_b} \cdot (r_{e_t,i_a} - \overline{r_{e_t}})}{\sum_{t=1}^{E} \omega \cdot sim_{e_t,u_b}} + \overline{r_{u_b}}$$

where $sim_{(i_s,i_a),(u_t,u_b)}$ is defined:

$$sim_{(i_s,i_a),(u_t,u_b)} = \frac{sim_{i_s,i_a} \cdot sim_{u_t,u_b}}{\sqrt{sim^2{}_{i_s,i_a} + sim^2{}_{u_t,u_b}}} \tag{15}$$

In practice, SIR'， SUR', $SUIR'$, and SER' can then be fused to extract the prediction rating by a fusing function. λ, $\delta\ and\ \theta$ are three parameters introduced to balance SIR'， SUR', $SUIR'$, and SER' in order to achieve better recommendation and fit these ratings to real conditions. The function is defined:

$$\begin{aligned} SR': \widehat{r_{u_b,i_a}} &= \text{£}\{SIR',\ SUR', SUIR', SER'\} \\ &= (1-\lambda)\cdot(1-\delta)\cdot(1-\theta)\cdot SIR' \\ &\quad +(1-\lambda)\cdot(1-\delta)\cdot\theta\cdot SUR' \\ &\quad +(1-\lambda)\cdot\delta\cdot SER' \\ &\quad +\lambda\cdot SUIR' \end{aligned} \tag{16}$$

3 Evaluation

An *Expert*CF is implemented in Python, a succinct and fast programming language. The evaluation is done by using MovieLens, a data set from GroupLens Research at the University of Minnesota, It is one of the most popular machine learning data sets for recommender system. Its original dataset contains 138,000 users and 27,000 movies. Our evaluation only used partial data for training and partial data for testing. The division of the dataset is in proportion (60%:40%) and 60% used for training and 40% is used for testing. The program was run with Windows 10 64-bit Operating System with 16GB RAM and 3.40GHz.

MAE metric is adopted in our tests as it is the most used metric in recommender system offline testing. The function is defined as follows:

$$MAE = \frac{\sum_{u\in T}\left|r_{u,i} - \widehat{r_{u,i}}\right|}{|T|} \tag{17}$$

Where $r_{u,i}$ denotes ratings that user *u* rates on item *i* and $\widehat{r_{u,i}}$ is the predicted rating. *T* denotes the test set with $|T|$ presenting the size of it. As the function illustrates the difference between real value and predicted value comes smaller, the more accurate the recommender performs, which means lower MAE indicates higher accuracy.

3.1 Overall Performances in terms of accuracy

In this evaluation, *expert*CF was compared with traditional memory-based CF approaches: an item-based approach using PCC (SIR) and a user-based approach using PCC (SUR), both also used in *expert*CF as offline calculation to improve accuracy performance.

The dataset from MovieLens includes 27,000 movies and 138,000 users. We randomly extracted 500 users each user rated over 40 movies and they were grouped by selecting the first 100, 200 and 300 users, denoted as ML_100, ML_200 and ML_300, as the training set. The last 200 users were used as the test set. We varied the number of items rated by active users from 5, 10 to 20, denoted as Given5, Given10 and Given20. Few ratings were not adding any improvement to our predictions. Therefore, sparse users were cleaned though choosing artificially generated expert cluster E_i. As the relationship between minimum rating threshold and expert numbers selected among active users illustrated in Figure 2, a final threshold for number of ratings a user has to make to be an expert was set 250. The rating scale E_i was set to 8294 to be an expert since an expert has to have a fixed marking scale and deviate, to evidently demonstrate every item he or she rates fair, and deviate less from one's to another's.

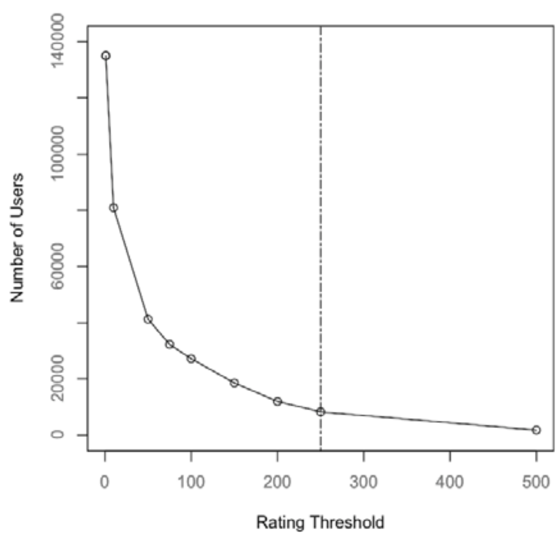


Figure 2 Relations Between Minimum Rating Threshold and Expert Numbers Selected

The other parameters of *expert*CF are previously set as follows: C=50, λ=0.7, δ=0.1, θ=0.1, σ=0.55, K=25, M=95, E=10 and ω=0.35. As **Table 2** demonstrated, *expert*CF outperforms the SUR and SIR considerably with respect to recommendation accuracy.

Table 2. MAEs on MovieLens among different CF Approaches

Training Set	Method	Given5	Given10	Given20
ML_300	*expert***CF**	0.765	0.744	0.721
	SUR	0.838	0.814	0.802
	SIR	0.870	0.838	0.813
ML_200	*expert***CF**	0.793	0.757	0.731
	SUR	0.843	0.822	0.807
	SIR	0.855	0.834	0.812
ML_100	*expert***CF**	0.802	0.779	0.770
	SUR	0.876	0.847	0.811
	SIR	0.890	0.801	0.824

3.2 Expert Cluster Coverage

In the previous study, it is found that in actual recommendation error includes a great portion of what brought by the noise in the users' explicit feedback (biased). *Expert* may not increase the accuracy rather to eliminate partial errors caused by the data sparsity. So it is interesting to see how expert can solve cold-start problem and to enhance the coverage. The second evaluation is to see how expert to make up coverage where experts typically are motivated to rate new item, it solves new item entry, in other words, new item leans to be recommended.

In this evaluation, supposed parameters stay the same, coverage rate for *expert*CF and other three state-of-the-art recommenders are given in Table 3. Where CFSF is CF only involves smooth and fusion [12]. Only ML_300 and given 20 are used.

Table 3. Coverage Rate on ML_300 for the different CF Approaches

Training Set	Method	Given20	Coverage
ML_300	*expert***SFCF**	0.721	**96.9%**
	CFSF	0.705	94.1%
	SUR	0.802	91.2%
	SIR	0.813	93.6%

The results show the coverage increase significantly in comparison with other approaches.

4 Conclusion

Expert CF build based on existing approached with aims to resolve problems of data sparsity and scalability associated with the CF in recommendation systems. This paper reports our efforts in artificially generated experts based on the existing data. Evaluations show the improvements on the solving problems. Accuracy is improved when data is sparse and coverage is also improved. However, a few deficiencies need future improvement. The original expectation intended to cover all users request, that is, to achieve 100% coverage, toward complete solution of cold-start issue. The system addressed this problem in which provided fusing function to fuse rating prediction with a fixed weight whereas, from a consideration of original goal, a separate function that deals with identification of an inactive user or new user would effectively solve this problem.

Meanwhile, dynamic expert set depends on user activity. Supposed using matrix factorization to detect each user's eigenvalue, which the result could indicate user's interest and make experts sample become entire original user set, each user could be an expert in a specific condition and the coverage rate could reach further improvement.

REFFERENCE

[1]. J.B. Schafer, J.A. Konstan, and J. Reidl, "E-Commerce Recommendation Applications," Data Mining and Knowledge Discovery, Kluwer Academic, 2001, pp. 115-153.

[2]. Greg Linden, Brent Smith, and Jeremy York， "Amazon.com Recommendations Item-to-Item Collaborative Filtering" IEEE INTERNET COMPUTING

[3]. Gediminas Adomavicius, Alexander Tuzhilin. (2005). Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. *IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING*. **17**, 734-749.

[4]. F. Chang, J. Dean, S. Ghemawat, W. C. Hsieh, D. A. Wallach, M. Burrows, T. Chandra, A. Fikes, R. E. Gruber. (2006). Bigtable: a distributed storage system for structured data. *Proceedings of OSDI*, Berkeley, CA, USA. USENIX Association. 205-218.

[5]. WY Chen， Y Song， H Bai， CJ Lin， EY Chang. (2011). Parallel spectral clustering in distributed systems. *IEEE Transactions on Software Engineering*. **33(3)**, 568-586.

[6]. G. Linden, B. Smith, J. York. (2003). Amazon.com Recommendations: Item-to-Item Collaborative Filtering, *IEEE Internet Computing, Jan./Feb.*

[7]. Nathan Nan Liu, Xiangrui Meng, Chao Liu, Qiang Yang. (2011). Wisdom of the Better Few: Cold Start Recommendation via Representative based Rating Elicitation. Proceedings of the 5th ACM Recommender Systems Conference.

[8]. Daqiang Zhang, Jiannong Cao, Jingyu Zhou, Minyi Guo and Vaskar Raychoudhury. (2009). An Efficient Collaborative Filtering Approach Using Smoothing and Fusing. *2009 International Conference on Parallel Processing*. 558-565.

[9]. X Amatriain， N Lathia， JM Pujol， H Kwak and N Oliver. (2009). The Wisdom of the Few A Collaborative Filtering Approach Based on Expert Opinions from the Web. International Acm Sigir Conference on Research & Development in Information Retrieval. 532-539.

[10]. Gangmin Li, Minghuang Chi and Gautam Pal. (2017). Expert CF: Sparse data matrix completion with artificial experts. International Conference on Recent Advancements in Computing, IoT and Computer Engineering Technology (CICET'17). Taipei, Taiwan October 23-25, 2017.

SOCIETY FOR SCIENCE AND EDUCATION
UNITED KINGDOM

TMLAI TRANSACTIONS ON MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE

Volume 6 No. 2
ISSN 2054-7390

A Tool to Create Assurance Case through Models

[1]Hiroyuki Utsunomiya, [2]Nobuhide Kobayashi, [1]Shuji Morisaki, [1]Shuichiro Yamamoto
[1]Nagoya University Graduate School of Information Science, Nagoya, Japan.;
[2]DENSO CREATE INC., Nagoya, Japan.;
utsunomiya.hiroyuki@k.mbox.nagoya-u.ac.jp; nobuhide@dcinc.co.jp;
morisaki@i.nagoya-u.ac.jp; yamamotosui@icts.nagoya-u.ac.jp

ABSTRACT

In this paper, an assurance case development tool is proposed to derive the argument decomposition structure from generic model definitions. The method solves O-DA issues for assuring business, application, and technology architecture of TOGAF. An example case study using the proposed tool is also shown for the system configuration model of the tool itself.

Discussions based on the case study showed the effectiveness and appropriateness of the proposed methods.

Future work includes the formalization of assurance case derivation process from ArchiMate, UML, and SysML models.

Keywords: dependability, architecture models, Enterprise Architecture, experimental tool evaluation, O-DA

1 Introduction

The Open Group Real Time & Embedded Systems Forum focuses on standards for high assurance, secure dependable and complete systems. The Open Group announced the publication of the Dependability through Assuredness™ Standard(O-DA) published by The Open Group Real-Time & Embedded Systems Forum[1]. At the heart of this O-DA(Open Dependability through Assuredness) standard, there is the concept of modeling dependencies, building assurance cases, and achieving agreement on accountability in the event of actual or potential failures. Dependability cases are necessary to assure dependable systems[2]. The DEOS process was proposed to manage dependability of complex systems by using dependability cases[3]-[5]. The dependability concept is able to define by quality properties[6].

Complex systems, especially where the boundaries of operation or ownership are unclear, are often subject to change: objectives change, new demands are made, regulations change, business partners are added, etc. So when the failure of the system can have a significant impact on lives, income or reputation, it is critical that a process is in place to identify these changes and to update the architecture by using the assurance cases and the agreements on accountability. It is also critical that a process is in place to detect anomalies or failures, to understand the causes, and to prevent them from impacting the system in the future.

DOI: 10.14738/tmlai.62.4428
Publication Date: 27th April, 2018
URL: http://dx.doi.org/10.14738/tmlai.62.4428

The O-DA standard outlines the criteria for mitigating risk associated with dependability of complex interoperable systems. It also outlines individual accountability. O-DA will benefit organizations relying on complex systems to avoid or mitigate the impact of failure of those systems. O-DA includes the DEOS process mentioned before. The Change Accommodation Cycle and the Failure Response Cycle that together provide a framework for these critical processes. O-DA brings together and builds on The Open Group vision of Boundaryless Information Flow. These concepts include O-DM(Open Dependency Modeling) and Risk Taxonomy of The Open Group Security Forum, and Enterprise Architecture(EA) models of The Open Group ArchiMate® Forum[7],[8]. However, the relationship between O-DA and ArchiMate concepts has not yet been clear. ArchiMate models include strategy, business, application, technology, and physical architecture as well as motivation of architecture. UML[27] only focusses to model software systems. SysML[28] extends UML by adding requirements and parametric diagrams for modeling systems engineering artifacts. Both UML and SysML are not able to model EA.

In this paper, an assurance case generation tool is proposed to argue the assuredness for these three kinds of architectures models. Section 2 describes related work on argument pattern approaches for assurance cases. Section 3 describes an assurance case creation tool which is proposed to generate the argument decomposition structure from various architecture models. In section 4, an example case study using the tool is presented. Discussions on the effectiveness of the tool are shown in section 5. Our conclusions are presented in section 6.

2 Related work

The safety case, the assurance case, and the dependability case are currently the focus of considerable attention for the purpose of providing assurance and confidence that systems are safe. Methods have thus been proposed for representing these using Goal Structuring Notation(GSN)[9]-[13]. GSN patterns were originally proposed by Kelly and McDermid[11]. In the absence of any clearly organized guidelines concerning the approach to be taken in decomposing claims using strategies and the decomposition sequence, engineers have often not known how to develop their arguments. It is against this backdrop that the aforementioned approaches to argument decomposition patterns —architecture, functional, attribute, infinite set, complete(set of risks and requirements), monotonic, and concretion—were identified by Bloomfield and Bishop[14]. When applying the architecture decomposition pattern, claims of the system are also satisfied for each constituent part of the system based on system architecture. Despotou and Kelly[15] proposed a modular approach to improving clarity of safety case arguments. Hauge and Stolen[16] described a pattern based safety case approach for the Nuclear Power control domain. Wardzinski[17] proposed safety assurance strategies for the autonomous domain. An experimental result of argument patterns was reported by Yamamoto and Matsuno[18]. Argument pattern catalogue was proposed based on the format of design patterns by Alexander, Kelly, Kurd and McDermid[19]. In their paper, Alexander and others showed a safe argument pattern based on failure mode analysis. Graydon and Kelly[20] observed that argument patterns capture a way to argue about interference management. Ruiz, Habli and Espinoza[21] proposed an assurance case reuse system using a case repository.

Hawkins, Habli, Kolovos, Paige and Kelly proposed a Model-Based Assurance Case development approach by weaving reference information models and GSN argument patterns[22]. They used a script language to define precise weaving procedures. These approaches assume specific adaptation mechanisms to generate assurance cases for reusing GSN patterns.

Although Yamamoto and others proposed the method to create assurance cases based on ArchiMate models[23],[24], the tool to automate the method was not mentioned. This paper proposed a tool based on the method.

3 Assurance case creation tool

3.1 Overview

The configuration of the tool named as UC2CT(Unified Context to Claim Tool) is shown in Fig.1. The tool reads architecture model, quality property, and risk measure definitions written in XML. UC2CT can decompose claims by using these three types of definitions. The generated assurance cases are represented in the SACM(Structured Assurance Case Metamodel) v1.0 XMI schema definition.

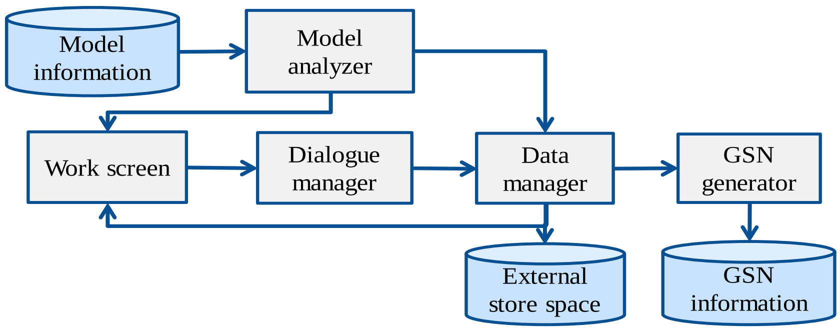


Figure.1 Configuration of the assurance case tool.

The generated XMI information is used to develop graphical structures of the assurance cases by using assurance case editors which support the XMI import facility. The tool was implemented by using Excel. These three xml definitions are used as input to decompose a top goal claim in table format. UC2CT is developed by extending the Microsoft Excel. The example screen is shown in Fig.2.

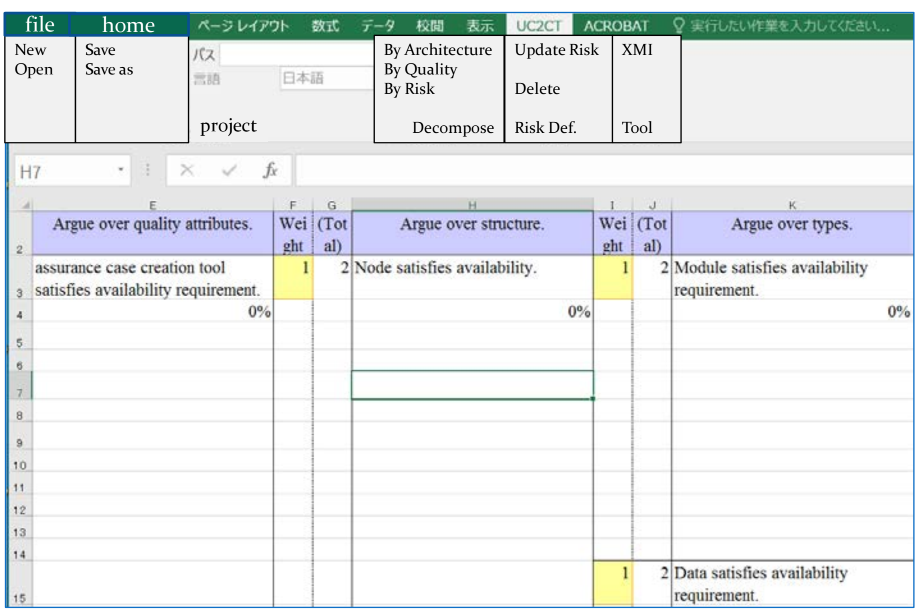


Figure.2 Display example of the assurance case tool.

The extended new menu commands are, “New,” “Open,” “Decompose,” “Risk Definition,” and “Tool.” “Decompose” menu consists of “By Architecture,” “By Quality,” “By Risk” sub menus. “Risk Definition” menu consists of “Update” and “Delete” risk sub menus. Tool menu command provides XMI export function to generate the assurance case information developed on UC2CT. As shown in the table, there are weight and total columns in the tool table. These are attributes of nodes and relationships of assurance cases proposed in [25]. The attributes are used to reduce numbers of assurance case nodes and conflicts among quality claims, such as safety and security.

3.2 Model definitions

In general, every model is defined by using nodes and their relationships. Therefor models can be defined by the following XML template.

```
-<modelDefinition>
  -<model name=“ModelName”>
    -<types>
      -<nodes>Node Name Definition Part </nodes>
      -<relations>Relation Name Definition Part </relations>
    </types>
    -<instances>
      -<nodes>Node instance definition Part </nodes>
      -<relations>Relation instance definition Part</relations>
    </instances>
  </model>
</modelDefinition>
```

Node Name Definition Part includes a list of the following statement.

```
  <node> Name Of Node </node>
```

Relation Name Definition Part includes a list of the following statement.

```
  <relation> Name Of Relation </relation>
```

Node instance definition Part includes a list of the following statement.

```
  <node id=“Id” type=“NameOfNode”>
    NodeInstanceName
  </node>
```

Relation instance definition Part includes a list of the following statement.

```
  <relation id=“Id” type=“NameOfRelation” target=“Id” source=“Id” />
```


The quality properties and risk measures are also defined by using XML in the same way. The XML notation can universally be applied to describe any models that has nodes and relationships among nodes.

3.3 Pattern of created assurance case

The tool is based on the structure of assurance case proposed in [22] as shown in the following table.

Table 1. Assurance case pattern.

Hierarchy	Description
Root goal	The root goal states that the model shall satisfy dependability principles
Node and relationships	Root goal is decomposed by nodes and relationship of the model
Types of nodes and relationships	Second level goals are decomposed by the types of nodes and relationships of the model
Instances of nodes and relationships	Third level goals are decomposed by instances of nodes and relationships of the model
Risk mitigation for instance risks	Fourth level goals are decomposed by risks for the corresponding instances
Evidence	Evidence supports to mitigate all the risks

The first level sub-goal claims state that concept elements and relationships of the model satisfy dependability principles. The second level sub-goal claim states that category of elements and their relationships among the model satisfy dependability principles. The third level goals are decomposed by instances of concepts and relationships of the models. The fourth level goals are decomposed by risks for the corresponding instances and are supported by the evidence to mitigate risks. Therefore, the fifth level of the assurance case consists of evidences for the fourth level goals.

4 Case Study

The example study was conducted to evaluate the effectiveness of the proposed assurance case creation tool for assuring the dependability of the tool itself.

4.1 Target system

The target system of the case study is the assurance case creation tool proposed in this paper. The model of the tool was defined described below in the form of the model definition in the previous section. In Fig.1, there are module and data. Therefore, node types in the definition are Module and Data. Module-Module and Module-Data are two types of relationships.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<modelDefinition>
  <model name="assurance case creation tool">
    <types>
      <nodes>
        <node>Module</node>
        <node>Data</node>
```

```
        </nodes>
        <relations>
            <relation>Module_Module</relation>
            <relation>Module_Data</relation>
        </relations>
    </types>
    <instances>
        <nodes>
            <node id="in-001" type="Module">Model Analyzer</node>
            <node id="in-002" type="Module">Dialogue manager</node>
            <node id="in-003" type="Module">GSN generator</node>
            <node id="in-004" type="Module">Data manager</node>
            <node id="in-005" type="Module">Work screen</node>
            <node id="in-006" type="Data">Model information</node>
            <node id="in-007" type="Data">External store space</node>
            <node id="in-008" type="Data">GSN information</node>
        </nodes>
        <relations>
            <relation id="ir-011" type="Module_Module" source="in-001" target="in-004" />
            <relation id="ir-012" type="Module_Module" source="in-001" target="in-005" />
            <relation id="ir-013" type="Module_Module" source="in-002" target="in-004" />
            <relation id="ir-014" type="Module_Module" source="in-004" target="in-005" />
            <relation id="ir-015" type="Module_Module" source="in-004" target="in-003" />
            <relation id="ir-016" type="Module_Module" source="in-005" target="in-002" />
            <relation id="ir-017" type="Module_Data" source="in-006" target="in-001" />
            <relation id="ir-018" type="Module_Data" source="in-007" target="in-004" />
            <relation id="ir-019" type="Module_Data" source="in-004" target="in-007" />
            <relation id="ir-020" type="Module_Data" source="in-003" target="in-008" />
        </relations>
    </instances>
</model>
```


</modelDefinition>

The dependability properties consist of availability, reliability, safety, integrity, consistency, and maintainability are also defined in XML. In addition, risks are defined for each nodes and relations in XML.

The XML model definition is loaded by the tool to create the assurance case based on the model. Then the following XMI information was generated to create the assurance case.

<?xml version="1.0" encoding="utf-8" standalone="no"?>

<ARM:Argumentation content="" description="" id="assurance case creation tool" xmi:id="38888871" xmlns:ARM=http://schema.omg.org/SACM/1.0/Argumentation xmlns:xmi=http://www.omg.org/XMI xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance xsi:version="2.0">

<argumentElement content="assurance case creation tool satisfies dependability requirement." description="" id="G0" xmi:id="9a9aeeb6-1eb2-4b2f-a07b-1b3797cf389b" toBeSupported="" assumed="" xsi:type="ARM:Claim" />

··············· *omitted for the limitation of space* ···············

</ARM:Argumentation>

Fig. 3 shows the top level view of the created assurance case with a GSN editor by importing the above xmi file.

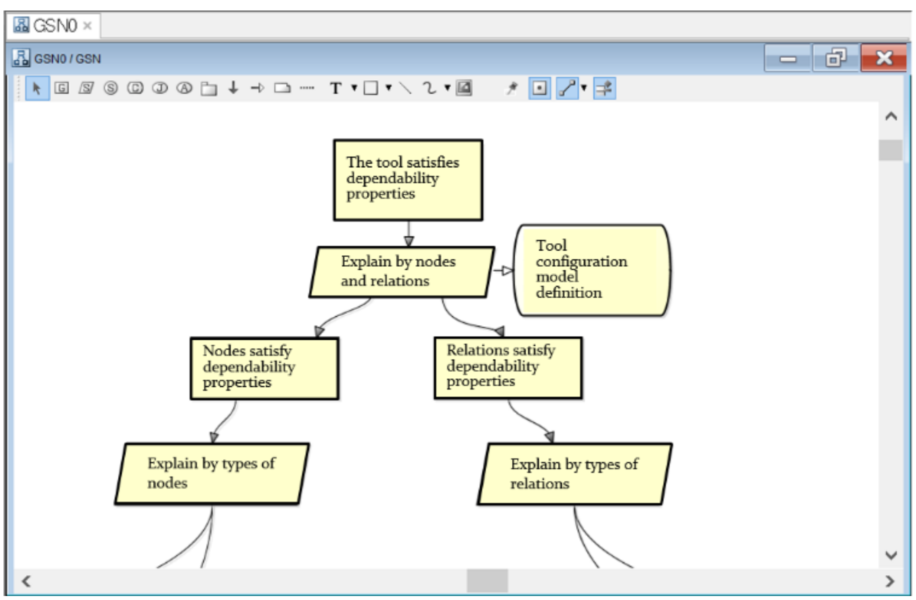


Figure.3 Example of created assurance case.

The created assurance case consists of 218 claim nodes, 53 strategy nodes, 47 context nodes and 165 evidence. The assurance case for the tool was also developed by human with the same method proposed in [22]. The both human maid and tool made assurance cases have the same nodes.

4.2 Comparison of development time

The assurance case development time for using the tool contains the model definition time and the tool operation time. Table.2 shows the comparison for developing the assurance case for the case study and the model checker. The time to create the assurance case for the case study was 275 min. In contrast, by using the tool it was only 47 min. Except for the model definition, it was 19 min. to create the assurance case. Node development productivity was 5.84 sec. per node, because 483 nodes are developed in 47 min.

Model checker is a tool to confirm the validity of software model shown in [26]. It took 110 min. to create an assurance case that confirms the completeness of error handling of the tool. However, with the tool, it was possible to create an assurance case in only 13 minutes, 7.72 sec. per node.

The comparison shoed that the assurance case tool can improve the development time to create assurance case.

Table 2. Comparison of the assurance case development time.

method	Work time of Case Study	Work time of Model Checker
Without Tool	275 min.	110 min.
XML definition	28 min.	8 min.
Tool	19 min.	5 min.

5 Discussion

5.1 Effectiveness

The case study on the assurance case creation tool was executed to evaluate the effectiveness of the tool proposed. The result showed the derivation from the model of the tool in XML to assurance case is easy and traceable. This showed the effectiveness of the creation method. Although the creation was only described for the tool, it is clear the same results can be derived for other models.

The XML model template is designed so that designers can describe models in the unified manner.

Moreover, if the XML model definitions was generated by modelling tools, the model definition time can be eliminated. For the case study, approximately 93% of the assurance case development time was reduced. This shows the tool has the capability to improve the assurance case productivity largely.

The table size of UC2CT can be extended to the limit of Excel. This shows UC2CT can be used to develop large scale assurance cases. As UC2CT exhaustively decompose assurance cases by architectures and quality properties, it may necessary to reduce the number of nodes. In this case, quantitative attributes are available to reduce unimportant claims by assigning low numbers.

5.2 Applicability

The applicability of the assurance creation tool to ArchiMate is clear by the above discussions. The BA, AA, and TA described in ArchiMate models can be easily defined in the form of XML template proposed by this paper. Any architecture models in ArchiMate contain nodes and relationships among nodes. Therefore, the decomposition hierarchy defined by Table 1 can be applied to any models consists of nodes and relationships.

In addition, every graph G can be represented by nodes and relationships among nodes. Nodes and their relationships may have categories. It is necessary to validate every instance of nodes and relationships according to the sort of categories, if we validate the G. Therefore, the proposed approach can be applicable for any models to assure the dependability properties.

5.3 Limitation

This paper only examines the effectiveness of the proposed method for two example architecture. More evaluations are necessary to generalize the effectiveness of the proposed tool and the method.

ACKNOWLEDGMENT

This work has been conducted as a part of "Research Initiative on Advanced Software Engineering in 2015" supported by Software Reliability Enhancement Center(SEC), Information Technology Promotion Agency Japan(IPA).

REFERENCES

[1] Real-Time and Embedded Systems, "Dependability through Assuredness™ (O-DA) Framework," Open Group Standard, 2013.

[2] D. Jackson, "Software for dependable systems– sufficient evidence?," NATIONAL RESEARCH COUNCIL, 2008.

[3] DEOS project, http://www.crest-os.jst.go.jp, 2013.

[4] DEOS project, JST White Paper DEOS-FY2011-WP-03J, www.dependable-os.net/ja/topics/file/White_Paper_V3.0J.pdf , 2011.

[5] M. Tokoro, eds., "Open Systems Dependability, Dependability Engineering for Ever-Changing Systems," CRC Press, 2012.

[6] Avizienis, Laprie, J., Randell, B., Landwehr, C., "Basic concepts and taxonomy of dependable and secure computing," IEEE Transactions on Dependable and Secure Computing, vol.1. No.1, pp.11-33, 2004.

[7] Josely, A., "TOGAF® Version 9.1 A Pocket Guide," 2011.

[8] Josely, "ArchiMate®3.0, A Pocket Guide," The Open Group, Van Haren8 Publishing, 2016.

[9] T. Kelly, "A Six-Step Method for the Development of Goal Structures," York Software Engineering, 1997.

[10] T. Kelly, J. McDermid, "Safety Case Construction and Reuse using Patterns," University of York, 1997.

[11] T. Kelly, "Arguing Safety, a Systematic Approach to Managing Safety Cases," PhD Thesis, Department of Computer Science, University of York, 1998.

[12] J. McDermid, "Software safety: where's the evidence?, " in SCS '01: Proceedings of the Sixth Australian workshop on Safety critical systems and software, pp. 1-6, Darlinghurst, Australia, Australian Computer Society, Inc., 2001.

[13] T. Kelly, and R. Weaver, "The Goal Structuring Notation – A Safety Argument Notation," Proceedings of the Dependable Systems and Networks 2004 Workshop on Assurance Cases, 2004.

[14] R. Bloomfield, and P. Bishop, "Safety and Assurance Cases: Past, Present and Possible Future," Safety Critical Systems Symposium, Bristol, UK, 2010.

[15] G. Despotou, and T. Kelly, "Extending the Concept of Safety Cases to Address Dependability," in proceedings of the 22nd International System Safety Conference (ISSC), Providence, RI USA, 2004.

[16] Hauge, and K. Stolen, "A Pattern-Based Method for Safe Control Systems Exemplified within Nuclear Power Production," SAFECOMP 2012, LNCS 7612, pp.13-24, 2012.

[17] Wardzinski, "Safety Assurance Strategies for Autonomous Vehicles, "SAFECOMP 2008, LNCS 5219, pp.277-290, 2008.

[18] S. Yamamoto, and Y. Matsuno, "An evaluation of argument patterns to reduce pitfalls of applying Assurance Case," Assure2013.

[19] R. Alexander, T. Kelly, Z. Kurd, and J. McDermid, "Safety Cases for Advanced Control Software: Safety Case Patterns," Technical report, University of York, 2007.

[20] P. Graydon, and T. Kelly, "Assessing Software Interference Management When Modifying Safety-Related Software," in Proceedings of the Next Generation of System Assurance Approaches for Safety-Critical Systems (SASSUR) Workshop, SAFECOMP 2012, Springer, 2012.

[21] Ruiz, I. Habli, and H. Espinoza, "Towards a Case-Based Reasoning Approach for Safety Assurance Reuse," SAFECOMP 2012 Workshops, LNCS 7613, pp. 22–35, 2012.

[22] R. Hawkins, I. Habli, I., D. Kolovos, R. Paige, and T. Kelly, "Weaving an Assurance Case from Design: A Model-Based Approach," HASE15, pp.110-117, 2015.

[23] S. Yamamoto, "An approach to assure Dependability through ArchiMate," SAFECOMP 2015 Workshops, LNCS 9338, PP.50-61, Assure 2015, DOI: 10.1007/978-3-319-24249-1_5.

[24] Shuichiro Yamamoto and Nobuhide Kobayashi, Mobile Security Assurance through ArchiMate, Vol. 4, No. 3 of IT Convergence Practice, pp.1-8, (INPRA), 2017, http://inpra.yolasite.com/vol4no3.php

[25] Shuichiro Yamamoto, Assuring Security through Attribute GSN, ICITCS 2015, 5th International Conference on IT Convergence and Security (ICITCS), pp.1-5, 2015

[26] Nobuhide Kobayashi, Assurance case development method using SPRME on software review, ER2016, 2016.

[27] OMG, UML, http://www.uml.org/

[28] OMG, SysML, http://www.omgsysml.org/