A Bayesian Network-based Uncertainty Modeling (BNUM) to Analyze and Predict Next Optimal Moves in Given Game Scenario

Vinayak Jagtap
College of Engineering, Pune, Maharashtra, India
vinayak5jagtap@gmail.com

Parag Kulkarni
iKnowlation Research Labs Pvt Ltd, Tokyo International University, Japan
paragindia@gmail.com

Abstract: *As machine learning emerged, it is being used in a variety of applications like speech recognition, image recognition, sequence modeling, etc., Sequence modeling is one type of application where resultant sequences are generated based on historical data inputs provided. These sequences are fairly work in an uncertain environment like games or sports. In the case of a game or a sport, there is a sequence of moves selected by multiple players. There is a statistical uncertainty observed for simple to more complex games. For example, while playing chess, a simple statistical modeled uncertainty would be enough to choose the next possible. This move selection is dependent on available free spaces of pieces or pawns. The sports like tennis, cricket, and other games need a more complex design for uncertainty modeling for next move selection. A Bayesian Network model will work if there is fairly less uncertainty in the selection of the next move. A Bayesian Network-based model will be best fitted if all possible moves are included before training any machine learning or deep learning model. This will be achieved with the usage of the Context-Li model. The proposed Bayesian Network-based Uncertainty Modeling (BNUM) is used to incorporate uncertainty, for next move selection. BNUM is a multi-variable, multi-level association to incubate uncertainty in learning. It helps to predict the next move in an uncertain gaming environment. Different case studies are incorporated to verify the hypothesis and the results are a sequence of moves represented in the context graph.*

Keywords: *Bayesian network, uncertainty modeling, deep learning, context graph, next move.*

Received April 15, 2021; accepted June 12, 2022
https://doi.org/10.34028/iajit/20/2/6

1. Introduction

In the era of computation and intelligent systems, huge processing of data is required. To automate these tasks, several algorithmic strategies were built over the years. Artificial intelligence and machine learning techniques have emerged to automate and solve various problems in recent times. These problems are generally to process data find patterns and net input trials to find out a similar outcome that is a close match for the historical pattern. To incorporate these patterns, several rules are produced, which ultimately handle decision-making [19, 24].

This huge processing is based on domain knowledge and individual understanding, intern impact biased decision making. Every problem has some variation in either of inputs or outputs expected. Henceforth, algorithmic strategies are updated for more efficient results [9]. Most of the latest expert system algorithms are designed to give results in a personalized way for various cases from small mechanical automation to the complex behavioral analysis of human behavior. Complex automation like human behavioral analysis is a much more tedious task and needs multiple inputs. The expert system designed for it may find multiple patterns in input or some expert opinion about humans may improve the context for behavioral analysis [22].

These discussions indicate that as data increases multiple patterns, multiple unknowns need to be found for more accurate results. The computation power is not a problem now, but analysis and knowledge building should be more domain-specific for efficient processing. Most of the time, algorithms written are task-specific, complex problem needs more complex solutions. Artificial Intelligence and Machine Learning are used to solve such problems. If contextual inputs are provided for many such machine learning applications, more accuracy is provided. Such accurate results are needed for dynamic scenarios like gaming and sporting events [16, 22].

Most gaming or sports players need a fast response in dynamic scenarios where multiple inputs are uncertain. If we take a simple example of chess; the moves are dependent on the opponent’s response, previous own moves, and possible empty spots for pawns and their predefined movement patterns. All such inputs to a system create difficulties while playing the game which is nothing but uncertainty in the selection of moves. This problem internally has multiple test cases like detection of player type, the strategy behind each move, and available resources. Henceforth, gaming and sports are more complex and need uncertainty modeled with an understanding of the current move context [11, 15].

In this paper, uncertainty is modeled using Bayesian Network-based Uncertainty Modeling (BNUM) which is input to deep learning which results in more contextual weight for generating moves sequence. In general, deep learning learn under historical input sequences given, which does not open for multiple possibilities available that are not included in the learning set. Here, the complete Bayesian Network considered possible moves from the current sequence of moves with probabilistic weights. With the use of Context-Li weights and moves are updated in dynamic scenarios which intern improves deep learning.

2. Related Work

2.1. Uncertainty and Creativity

Uncertainty is defined as “Unable to determine the truth value of logical statement as true or false.” The condition of the statement is not always deterministic in a real-life situation. Raining tomorrow, this condition is uncertain but by using the Bayes theorem probability of rainfall can be predicted. Uncertainty is defined by Shannon in the form of information gain and entropy. Probability theory is used to identify information gain and entropy [8, 17]. In the case of decision-making, optimal uncertainty is needed, for accurate decision-making. These decision-making systems try to follow equations or patterns to give predictions or decisions for the input sequences. Learning like supervised learning helps to incubate domain knowledge for deciding the next move [7, 10]. As per Gibbs, principal uncertainty can be observed in different situations, for example, catching a ball thrown in the air has different uncertainties like surface, gravitational force, wind, etc. Each such attribute can add different types of uncertainties, hence new uncertainties give birth to new creativities [8].

2.2. Game Design Concepts

For every game, there is a different game designing motto. These mottos are very much necessary while designing a game and its gameplay. For example, racing games need focused attention for every single second, while strategic games like chess need less attention but more strategies [1, 14, 27]. While sports like cricket and soccer need strategy and attention with surprises.

2.3. Uncertainty in Gaming

There is an association between game design and uncertainty [5]. Based on the association uncertainty can be modeled. In this section, a couple of case studies and their association to model uncertainty is described.

- Chess: Chess is an abstract strategy game with a mind-to-mind battle. This battle can be strategic or completely random. Randomness in playing is the only source of uncertainty. If both players are strategic, then they can be easily modeled with the sequence of moves. This will make prediction easy. This loses interest in the long run as well as achievement feeling for the winner also. In short, uncertainty is needed to keep players motivated to play, as well as for achievement. Few players do not want to lose a single piece, this represents the mindset and expertise towards mastery, while some of the players reach Mastery, they create their uncertainty to keep themselves motivated like multiplayer or multilevel chess. Such a population of different types of players creates new moves or creates new playing styles. So new moves or strategies generation is dependent on the player type, game design, situational awareness, etc. which should be included before the machine learning model is trained.
- Monopoly: A two-player game of telling truth is the primitive requirement of the monopoly, as it may offer the training and other scenarios. This game is a complex, strategic and multiplayer game.

This game cannot be strategically modeled to win. You win only when other players are bankrupt. Here influencing other players by making an impressive decision is key to winning the game. There is another uncertainty involved with luck. Dynamic numbers observed on dice for every turn decide the winner of the player. This creates great uncertainty and keeps the player motivated. Online monopoly is played for achievements. In this game, uncertainty with dice and cards drawn decides to win, but purchasing the cards is only based on the initial cash of each player. Here to keep the people motivated, design and winning are important.

To keep player motivated and interested in playing, game designer needs to consider uncertainty. These uncertainties can be of different types such as [5].

- Performative.
- Randomness.
- Surprises.
- Schedule.
- Player perception.
- Solver analytical.
- Narrative anticipation.
- Developer anticipation.
- Hidden information.

2.4. Uncertainty Modeling

The focus of this research work is to model uncertainty; the “Context-Li” model is already being proposed to explore the possibility of information gain which is similar to regularization. The context-Li model provides a way of extracting relevant contextual information. It also helps in the optimization of information to reduce overfitting problems in machine learning. In the case of gaming based on the shortest path finding problem, the Context-Li model explores all possible edges/paths before the machine learning model is learned. It builds a contextual graph that should contain the shortest path before learning. This intern reduces the error of missing edges [13]. Lack of information creates uncertainty in decision-making for next move selection in case of gaming. Most of the games select the next move based on probability which is used in modeling uncertainty. Shannon defined the uncertainty based on the entropy value [2].

Tossing a fair coin has a probability value of 0.5 for getting a head or tail. But if a coin is not fair, this information changes the probability value [21]. Bayesian Network is used for uncertainty modeling [18, 20] and related tasks like uncertainty used in an accident [25, 26], network congestion as gaming [3], and intrusion detection [23]. Uncertainty is modeled using various mathematical concepts like probability and entropy, transition state, and decision tree [4, 6, 12]. Uncertainty is modeled and managed in the following ways:

- Understanding Uncertainty.
- Modeling Uncertainty.
- Limiting Uncertainty.
- Propagation of uncertainty.

3. A Bayesian Network-Based Uncertainty Modeling (BNUM)

There is the use of conditional probabilities in the Bayesian network based on the Bayes theorem. It is represented with random variables and their conditional probabilities. For example, a full toss ball is delivered to a batsman. We consider a full toss ball to reduce uncertainty created by pitch. And we kept the fielding formulation the same, then the Bayesian Network-based probability described for shot selection is shown in Figure 1.

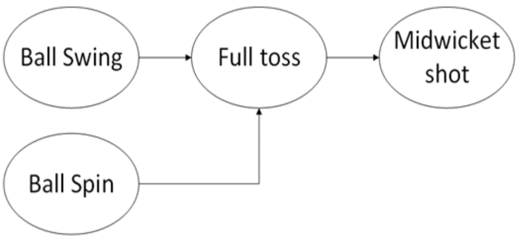

Figure 1. Sample bayesian network for cricket moves.

The probability of the system is given by:

$$P(SW, SP, F, M) = P(SW) * P(SP) * P(F \vee SW, SP) * P(M \vee F) \quad (1)$$

The probability of a shot played on midwicket is given by

$$P(M/F)\alpha P\,(SW, SP, F, M)/P\,(F) \quad (2)$$

As shown above, there are different such networks created and based on current game scenarios of a match or a population of inputs. The next move could be predicted based on the probability values available in historical data. This does not consider all possible uncertainties as a single match or a population of moves would not be enough for prediction. Hence these Bayesian networks should be updated to have all possible moves while predicting.

As represented in Figure 2 multiple Bayesian Networks are integrated for a single instance from different matches or populations of moves based on the current situation of the game. This aggregation and finding of contextual similar situations of the match are provided by the Context-Li model. For example, a couple of cricket matches may have a similar situation in particular instances; the decisions made might be different even if the situation or context of a game is similar. Here the aggregation of such multiple Bayesian Networks helps to reduce error by keeping all the possible outcomes from the current situation of the game. The proposed model is shown in Figures 2, and 3.

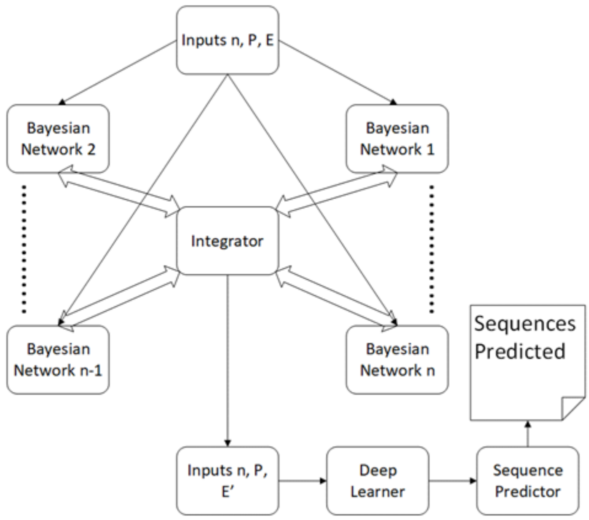

Figure 2. Bayesian networks model for uncertainty.

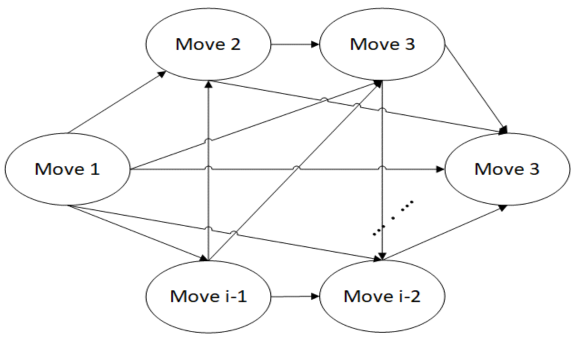

Figure 3. bayesian networks for all possible moves from any given sequences.

Based on probability values uncertainty could be incorporated into learning. Each Bayesian Network for each sequence of actions is combined based on possible moves from the current scenario, the example in cricket game format is T-20 and the full toss ball is bowled and the player type is defensive. He/she will try to hit for boundary rather than six but here other possibilities are needed to be included in shot selection, shot placement, fielding placement of another team, bowler speed, and bowler's line. Hence, it creates a completely connected graph of dependent variables for the current scenario and its propagation. All such dependent networks are integrated for finding more common dependencies and integrated. Here Context-Li model limits uncertainty and integrated Bayesian Network models, manage the propagation of uncertainty too. Abstract mathematical representation for BNUM is represented in the next part.

Let us consider there are 'n' variables that creates uncertainty which is called uncertainty variables,

$$n \in N \tag{3}$$

Integrated Bayesian network with P states in the form of the graph as

$$G(P, E), P \geq n, E \in [0,1] \tag{4}$$

Here edges represent the probability value for state transition. It is most of the time conditional probability. E' is an updated probability value

$$E' \geq E \tag{5}$$

$$G = G1 \cup G2 \cup \ldots \cup Gk \tag{6}$$

$G_i(P_1, E_1)$ is an individual Bayesian Network graph, which is a connected graph representing edges as the probability of Bayesian for a micro event like bowling. Edges are modified based on edges updating and depending on more possible moves or states. Most sequence generation uses conditional probability to represent as shown in the above equations. An algorithm of Bayesian Network building using Context-Li is represented as:

INPUTS
M [ip]: ith move from game sequence from pth population
U[M[i]] : uncertainty associated with move M[i]

OUTPUT
BN [: jth Bayesian Network consisting of context graph after M[j-1] th moves in the form of context graph
BN: aggregated Bayesian Network for the entire game.

Algorithm 1: Bayesian Network-Based Uncertainty Modeling(Bnum)

```
# From n population iterate to update moves and their weighs in the form of probability value for subsequent selection
for p to n do
# Loop for all moves from same population
    for i to t do
        M [ip ] = Context-Li(M [i ],p)
    Context-Li (M[i], p)
    for p to n do
        for i to t do
            U[M[ip]] = 1-probability(M[i],p)
# sorting will help in understanding which move is needed to be analyzed for more associated subsequent move from other population
    SU[M[ip]] = Sort(U[M[ip]])
    M [ip ] = Find(SU[M [ip ]])
# This execution makes sure no moves and no sequence left from each population and next step is integrate all moves in single Bayesian Network
for j to u do
    BN [ ] = M [ip ]
    BN = Merge(BN [ j ])
```

4. Case Studies

4.1. Tic Tac Toe

BNUM is applied for exploring a game and a Network graph with probability values is shown. Here, the game intends to win for both the players. 3*3 cell grid is the standard tic tac toe design considered for the case study.

If randomly selected player strategies and Bayesian network graph are generated. K9 is a completely connected graph, with random probabilities increasing from 1/9 to 1 probability values in sequences as sample space is reduced with each cell getting filled. But as players are trying to win, this context of the process limits uncertainty using Context-Li Model as given in Table 1 and we will generate a graph for one sequence of an event using the Bayesian Network. Figure 4 shows Context Graph for Tic Tac Toe

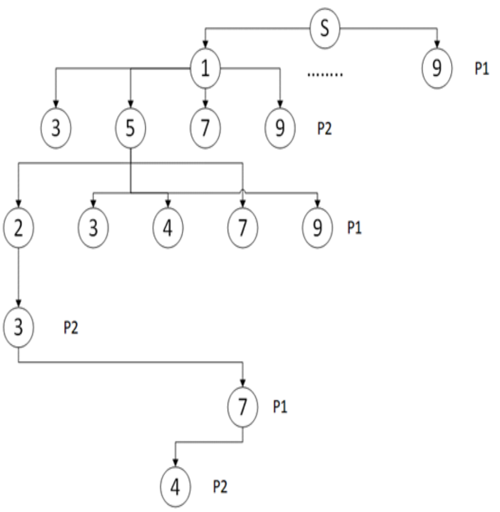

Figure 4. Context graph for tic tac toe.

Table 1. Context graph generated based on uncertainties for moves (Steps=A, Player= B, Move Probability value= C, Context-Li limited moves= D, Probability Reduced= E, Remark= F).

A	B	C	D	E	F
1.	P1	1/9	{1, 5, 9, 3, 7}	1/5	• Corner cell selection, group theory • Choose the remaining cell from selection
2.	P2	1/8	{1, 5, 9, 3, 7} (-1)	¼	
3.	P1	1/7	{1, 5, 9, 3,7} (-2)/ {2, 4, 6, 8}	1/3+1/4	
4.	P2	1/6	{1, 5, 9, 3,7} (-3)/ {2, 4, 6, 8} (-1)	1/2+1/4 or 1/3+1/3	
5.	P1	1/5	{1, 5, 9, 3,7} (-4)/ {2, 4, 6, 8} (-2)	1+1/4 or 1/3+1/2	
6.	P2	1/4	{1, 5, 9, 3,7} (-5)/ {2, 4, 6, 8} (-3)	1/4 or 1/3+1	
7.	P1	1/3			
8.	P2	½			
9.	P1	1			

4.2. Musical Node Sequences

In Indian Classical music, some Ragas define the composition of music and inside Raga, multiple songs can be composed. While creating a composition for the song, there are rules followed to maintain the feel of the selected Raga. These rules already limit uncertainty like the Context-Li model, but for sequence generation, musical nodes are composed based on one cycle of the base. Bhup Raga is considered for music sequence generation. Details of Bhup is given in Table 2. Now in cycles Vadi and Savadi, nodes have higher probability values. That means the sequence from Pakad and Chalan already represents it.

Hence, while generating musical nodes probability of sequence generation can be visualized in a diagram. Here probability of sequence generation containing G is more, later D but because of G and D, P is an intermediate node hence its probability of occurrences in generation also increased. Figure 5 shows Context Graph for Bhup

Table 2. Details of raag bhup.

Raag Bhup	
Arohana	S R G P D S'
Avarohana	S' D P G R S
Pakad	• S R G R S D1 S R G S R G R S D1 S R G P G D P G R S • G R P G G R S R D1 S G R S D1 S R G R P G D P G R S
Chalan	• S R G R S D1 S R G • S R G R S D1 P1 • P1 D1 S R G R G • S R P G • G R S R G P • G P D P D D S’ P G P D P D S’ R’ G’ R’ G’ • G’ R’ S’ D P G R S
Vadi	G
Samavadi	D

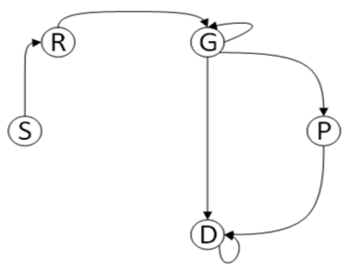

Figure 5. Context graph for raag bhup.

5. Results and Discussion

Chess Data Set from Kaggle (https://www.kaggle.com/datasets/datasnaek/chess) is being used for experimentation and Table 3 represents the comparative results. As deep learning itself has the capabilities to incorporate multiple sequences, there is not much variation available in results. The accuracy is improved as G(P, E) graphs and subgraphs are directed graphs, they provide more knowledge base for each step to predict the next move. With the help of Context-Li model G(P, E) from multiple games updated to the direction of their weight. It also limits the number of possible moves based on the current state of the game. Also, each game has a different game design, if some dynamic game or sport can go back to previous steps, there are more variations in prediction. In the case of chess, going back to the previous position is an open option. These deep learning neural networks are applied to BNUM generated from the sequences to predict the next move. The BNUM model is applied before inputs are added to neural networks. With context-graph generated by BNUM has all possible contextual moves information and weight updated. This intern help the neural network predict accurate results for the next move. Here small improvement is observed as these results are updated when BNUM is updated for a couple of populations of chess matches. It may further improve as the number of times BNUM is applied with multiple populations. Overall accuracy is improved by 1.03%. it is observed that Convolutional Neural Network (CNN) provides better accuracy as the moves are not just sequences of single pawns but it is also using move sequences from the combination of multiple pawns. Due to this more complex context graph is generated similar to Figure 5 and CNN predicts the next accurate move from the current position.

Table 3. Comparative result.

Method	Base Accuracy Results without BNUM	Accuracy Results with BNUM
Deep Chess (CNN)	63%	64.3%
RCNN	61%	62.1%
LSTM	64%	64.7%

6. Conclusions

Experimentation and observations of results indicate that by adding surprises to learning, new learning directions are introduced. This will improve learning itself and has many more sub-paths available. As Context-Li and BNUM both helped to incorporate uncertainty in a limited way for learning, it will also include adaptive behavior in learning. The BNUM helps in creating a complete network of subsequences and weights that are updated from one to multiple populations. We propose if uncertainty is not known or present in a limited amount, then the creation and addition of deliberate uncertainty will improve the performance of learning. This will also add a new flavor to creativity.

References

[1] Adams E. and Dormans J., *Game Mechanics: Advanced Game Design*, New Riders, 2012.

[2] Ayyub B. and Klir G., *Uncertainty Modeling and Analysis in Engineering and the Sciences*, CRC Press, 2006.

[3] Castiglioni M., Celli A., Marchesi A., and Gatti N., “Signaling in Bayesian Network Congestion Games: the Subtle Power of Symmetr,” *in*

Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, no. 6, pp. 5252-5259, 2021.

[4] Cárdenas I., "On the Use of Bayesian Networks As A Meta-Modelling Approach to Analyse Uncertainties in Slope Stability Analysis," *Georisk: Assessment and Management of Risk for Engineered Systems and Geohazards*, vol. 13, no. 1, pp. 53-65, 2019.

[5] Costikyan G., *Uncertainty in Games*, Mit Press, 2013.

[6] Dai J. and Deng Y., "A New Method to Predict the Interference Effect in Quantum-Like Bayesian Networks," *Soft Computing*, vol. 24, no. 14, pp. 10287-10294, 2020.

[7] Ghadim A., Pannell D., and Burton M., "Risk, Uncertainty, and Learning in Adoption of A Crop Innovation" *Agricultural Economics*, vol. 33, no. 1, pp. 1-9, 2005.

[8] Gibb A., "Creating Conducive Environments for Learning and Entrepreneurship: Living with, Dealing with, Creating and Enjoying Uncertainty and Complexity," *Industry and Higher Education*, vol. 16, no. 3, pp. 135-148, 2002.

[9] Han J., Pei J., and Kamber M., *Data Mining: Concepts and Techniques*, Elsevier, 2011.

[10] Hoskisson R. and Busenitz L., "Market Uncertainty and Learning Distance in Corporate Entrepreneurship Entry Mode Choice" *Strategic Entrepreneurship: Creating A New Mindset*, pp. 151-172, 2002.

[11] Howard-Jones P. and Demetriou S., "Uncertainty and Engagement with Learning Games" *Instructional Science*, vol. 37, no. 6, pp. 519-536 , 2009.

[12] Huang Z., Yang L., and Jiang W., "Uncertainty Measurement with Belief Entropy on the Interference Effect in the Quantum-Like Bayesian Networks," *Applied Mathematics and Computation*, vol. 347, pp. 417-428, 2019.

[13] Jagtap V. and Kulkarni P., "Contextual High-Level Uncertainty Modeling Reducing Surprises in Decision Making" *in Proceedings of International Conference on Electrical, Computer and Communication Technologies*, Coimbatore, pp. 1-4, 2019.

[14] Koster R., *Theory of Fun for Game Design*, O'Reilly Media, Inc, 2013.

[15] Kulkarni P., *Reverse Hypothesis Machine Learning*, Springer International Publishing, 2017.

[16] Li Y., "Deep Reinforcement Learning: An Overview" *arXiv preprint arXiv:1701.07274*, 2017.

[17] MacKay D. and Mac Kay D., *Information Theory, Inference and Learning Algorithms*, Cambridge University Press, 2003.

[18] Marcot B. and Penman T., "Advances in Bayesian Network Modelling: Integration of Modelling Technologies," *Environmental Modelling and Software*, vol. 111, pp. 386-393, 2019.

[19] Phillips-Wren G., Ichalkaranje N., and Jain L., *Intelligent Decision Making: an AI-Based Approach*, Springer Science and Business Media, 2008.

[20] Scanagatta M., Salmerón A., and Stella F., "A Survey on Bayesian Network Structure Learning from Data," *Progress in Artificial Intelligence*, vol. 8, no. 4, pp. 425-439, 2019.

[21] Soundappan P., Nikolaidis E., Haftka R., Grandhi R., and Canfield R., "Comparison of Evidence Theory and Bayesian Theory for Uncertainty Modeling" *Reliability Engineering and System Safety*, vol. 85, no. 1-3, pp. 295-311, 2004.

[22] Sung H., Hwang G., and Yen Y., "Development of A Contextual Decision-Making Game for Improving Students' Learning Performance In A Health Education Course," *Computers and Education*, vol. 82, pp. 179-190, 2015.

[23] Tabash M., Abd-Allah M., and Tawfik B., "Intrusion Detection Model Using Naive Bayes And Deep Learning Technique," *The International Arab Journal of Information Technology*, vol. 17, no. 2, pp. 215-224, 2020.

[24] Tzeng G. and Huang J., *Multiple Attribute Decision Making: Methods and Applications*, CRC press, 2011.

[25] Wang Q., Cai M., and Wei G., "A Scenario Analysis Under Epistemic Uncertainty in Natech Accidents: Imprecise Probability Reasoning in Bayesian Network," *Environmental Research Communications*, vol. 4, no. 1, pp. 015008, 2022.

[26] Zhang G., Thai V., Yuen K., Loh H., and Zhou Q., "Addressing the Epistemic Uncertainty in Maritime Accidents Modelling Using Bayesian Network with Interval Probabilities," *Safety Science*, vol. 102, pp. 211-225, 2018.

[27] Zichermann G. and Cunningham C., *Gamification by Design: Implementing Game Mechanics in Web and Mobile Apps*, O'Reilly Media, 2011.

Vinayak Jagtap is a passionate researcher in the field of computer science. His research areas include data science, machine learning, and data security. Vinayak holds a Bachelor of Engineering from Walchand College of Engineering, a Master in Engineering from Pune Institute of Computer Technology, and currently pursuing his Ph.D. from the College of Engineering Pune. He has a couple of patents filed along with dozen publications.

Parag Kulkarni is an entrepreneur, Machine Learning researcher, and author of best-selling Innovation strategy, ML, and Data science books. An avid reader, Parag holds Bachelors from Walchand College of Engineering Sangli (1990), a Ph.D. from IIT Kharagpur (2001), management education from IIM Kolkata, and was conferred a higher doctorate DSc by UGSM monarch, Switzerland (2010). Parag helped underperforming professionals, start-ups, and students to transform into happy and passionate warriors. Fellow of the IET, IETE, and senior member IEEE, Parag is the recipient of the Oriental Foundation Scholarship, distinguished alumnus award WCE - Sangli, and was nominated for the prestigious Bhatnagar award in 2013 and 2014. He was also awarded IETE-KR Phadke award for innovative entrepreneurship and research in 2019. Parag has published over 300+ research papers and articles in peer-reviewed journals and renowned conferences. He invented over a dozen patents and authored 14 books (with the world's best technical and business publishers like Bloomsbury, IEEE, Wiley, Prentice Hall, Springer, Oxford University Press, etc.).