Development of an Online Operator Tool to Support Real-Time Emergency Planning Based on the Use of Dynamic Event Trees and Deep Learning

Ji Hyun Lee*, Tunc Aldemir, Alper Yilmaz and Richard Denning
The Ohio State University, Columbus, US
lee.3584@osu.edu

Abstract: An initiating event that disrupts regular nuclear power plant (NPP) operation can result in numerous possible scenarios as time progresses depending on operator actions or uncertainties in NPP response. An approach that may be used to construct a real-time emergency guideline using dynamic event trees (DETs) to support the declaration of a site emergency and to guide off-site response is presented. In DET analysis, alternative scenarios or pathways are initiated by branching points in the tree as the accident progresses in time for which branching probabilities are assigned. The temporal behavior of the early stages of a severe accident as reflected by DETs and as observed from variables that can be monitored by NPP operators are used to estimate the likelihood of different levels of offsite release of radionuclides based on deep learning (DL) techniques with a training set of MELCOR/RASCAL scenarios. The ability of the resulting guideline to predict the likelihood of different levels of consequence is assessed using a separate test set of MELCOR/RASCAL scenarios.

Keywords: Dynamic PSA, Deep Learning, Convolutional Neural Network, MELCOR, Caffe

1. INTRODUCTION

In the event that an accident occurs at a nuclear power plant (NPP), the plant staff is responsible for returning the NPP to a safe stable state. The staffs are supported in taking these actions with Emergency Operating Procedures (EOPs) for which they receive extensive training in NPP simulators. If there is the potential for a significant release of radioactive material into the environment, the NPP manager must declare a level of site emergency. However, offsite response, such as evacuation, is ordered not by NPP personnel but by the State (and potentially the governor). In making offsite response decisions, the State personnel must rely heavily on the expertise of NPP personnel, their understanding of the current state of the NPP, and their projection as to the likely outcomes of the event.

All NPPs in the U.S. have performed Level 1 probabilistic risk assessments (PRAs) to estimate the frequency of events that may lead to core damage. Some PRAs have been extended to examine not only core damage frequency but also the progression of severe accident scenarios including the potential for containment failure and the release of radioactive material to the environment (Level 2 PRA) and the subsequent radiological exposure to members of the public (Level 3 PRA). These studies have been used to develop severe accident management guidelines (SAMGs) that need to be followed by the plant personnel to assist in the management of the severe accident. SAMGs, in general, are based on a very limited set of accident scenarios and do not take into account the variability of response of an event associated with modeling uncertainties and operator actions.

The dynamic event tree (DET) approach [1] was developed to improve the ability of PRA to more realistically account for the interaction of uncertainties in accident progression. The DET starts with the occurrence of the initiating event. Alternative scenarios or pathways are initiated by branching points in the tree as the accident progresses in time for which branching probabilities are assigned. If the accident is arrested prior to core damage, the radiological consequences of the scenario will be negligible. Similarly, if a severe accident is controlled prior to containment failure, the consequences will be smaller than if the containment fails.

In current practice, the declaration of a site emergency is based on the identification of the type of event sequence early in the accident. An online support tool (OST) that can assist the NPP personnel to assess an event sequence probabilistically as an Unusual Event (UE), an Alert, a Site Area Emergency (SAE) or a General Emergency (GE) as the accident evolves can reduce the level of exposure of the population, as well as reduce the negative impacts of possible evacuation.

Thus, the objective of OST would be two-fold: i) to support the declaration of site emergency, and, ii) to provide technical guidance to the State in undertaking emergency response activities associated with evacuation, sheltering in place and the distribution of the chemical KI for events with large radioactive iodine doses to prevent excessive thyroid loading. The OST presented in this paper is being developed taking advantage of continuing improvements in processing power of computers and the state-of-the-art in artificial intelligence (AI) techniques. The inputs to the tool are key observable parameters available to the operator from the plant and environmental instrumentation. The output is the projected levels of radiological exposure to members of the public and their likelihoods.

The accident model used to train the OST is based on scenarios generated in an extensive DET study performed for a station blackout (SBO) in a 3-loop pressurized water reactor (PWR) [1]. The objective of the study [1]for which the scenarios were generated was to demonstrate the seamless transition between Level 2 and Level 3 PRA (see Section 3). Clustering techniques for post-processing and unsupervised learning (see Section 2.1) were used to interpret the massive data generated from the MELCOR computer code [2] used to simulate the consequences of SBO. The analyses were limited to the range of potential outcomes from loss of offsite power scenarios followed by failure of emergency diesel generators. In the present study, deep learning (DL) [3], which is a subset of machine learning algorithms with multi-layer networks is implemented by means of the computer code Caffe [4] (also see Section 2.2) to develop the OST. The DET study that was performed in [1] was based on the use of MELCOR under the management of the Analysis of Dynamic Accident Progression Trees (ADAPT) [5] developed previously by The Ohio State University in collaboration with the Sandia National Laboratories. The results of the DET study [1] led to the generation of two sets of data. The first set of 2656 scenarios is used in this study to train, test, and validate the OST. The second set of data consisting of 775 scenarios will be used to test or validate the robustness of the OST in future work.

Two different modes can be used to analyze the set of scenarios when dealing with nuclear transients: end state, and transient analysis [6, 7]. The work presented in this paper can be categorized as transient analysis because the temporal behavior of the early stage of the accident is used to examine possible ways the NPP state can evolve and predict magnitude and likelihood of offsite radiological exposure. Section 2 overviews the features of DL relevant to this study. Section 3 describes the case study under consideration. Sections 4 and 5 present the results and conclusions of the study, respectively.

2. DEEP LEARNING: OVERVIEW

AI was introduced in 1955 by John McCarthy et al [8] as a parallel to the human decision-making process, with the term AI coined in the first AI conference in 1956 in Dartmouth Conference. The general definitions of AI can be grouped into 4 categories: thinking humanly, thinking rationally, acting humanly, and acting rationally [8]. In 1957, Allen Newell et al. programmed ‘Logic Theorist’ which was the first AI program [9] and which was designed to solve simple logic problems with traditional mathematical procedures [10].

DL is a branch of AI that mimics the way the human brain works to analyse problems. The DL technique used in this paper is applicable to temporally continuous datasets such as video, speech recognition and dynamics which makes it suitable for analysing temporal characteristics of event evolution in an NPP. DL can either be supervised or unsupervised. A simple example of unsupervised learning is illustrated in Figure 1 [11]. A computer cannot a priori recognize whether the object in the image is a person or an animal. DL selects certain parts from the image that are used as input pixels or the visible layer. Then, through a series of hidden layers, it extracts the features from the image of the previous layer and learns feature in a hierarchical manner from pixels to identify and train all layers jointly. After training with multiple examples, the resulting algorithm can be used to identify that the image is that of a person with higher probability.

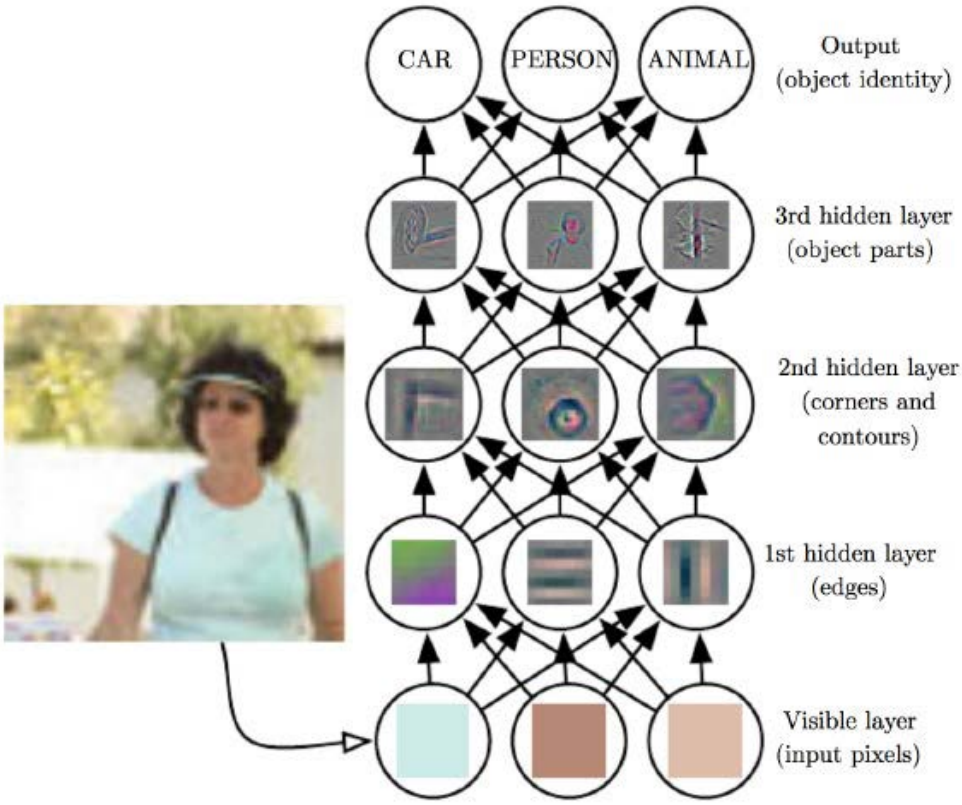
3. CASE STUDY

As indicated in Section 1, this study uses the database previously generated from [1] with the ADAPT software to manage the DET generation following the SBO. Failure/non-failure of the turbine-driven auxiliary feedwater (TDAFW) pump determines whether the event is recognized as a short-term SBO or a long-term SBO. Following the failure of AFWS, power to emergency equipment is supplied

by DC batteries whose depletion time was treated as a probability distribution. The study [1] has generated a few thousand scenarios representing different branching combinations. For each scenario, radionuclide release fractions and their probabilities are known.

The composition of the data set is described in Sections 3.1 and 3.2. The details of dataset selection are explained in Section 3.2. The DL software overview is presented in Section 3.3.

Figure 1 A machine goes through its own imaging processing procedure to derive information on the type of object in a figure [11]



3.1. Operator variables

The severe accident code MELCOR predicts the behavior of a large number of covariates associated with the thermal-hydraulic behavior of the accident. However, the operator only monitors a small number of variables. Thus, in determining the state of the NPP, the AI model must rely only on the variables that are actually monitored. A list of available observable parameters is tabulated below in Table 1 based on information provided by the PWR Owner's Group for the development of SAMGs [12].

Table 1 Critical parameters and instrumentation used for OST

Parameter	Primary purpose/information provided	Measurement Method	Alternate Method
Steam Generator (SG) Water level	• Reactor Coolant System (RCS) heat sink available • Creep rupture of SG tubes possible	• Wide range SG level*	• Narrow range SG level

	• Fission product scrubbing for faulty or leaking SG tubes		
SG Pressure	• Creep rupture of the SG tubes possible • Ability to inject into the SGs	• SG secondary pressure*	• TDAFW pump header pressure (only for select SGs)
RCS Pressure	• Ability to inject into the RCS • High Pressure Melt Ejection possible • Uncontrolled opening in the RCS	• Wide Range RCS pressure	• Pressurizer pressure • Accumulator pressure • Charging pump of LHSI pump dis-charge pressure
Core Temperature	• Transition from EOPs to SAMG • In-vessel recovery of core cooling	• Core exit Thermocouples (CETCs)*	
RCS Temperature	• Understanding earlier stages of accident progression	• CETCs* [same as the variables above]	• Resistance Temperature Detector (RTDs) [Hot Leg or Cold Leg]
RCS Water Level	• Understanding earlier stages of accident progression	• Reactor Vessel Level Indication system (RVLIS)	• Ex-core neutron detectors
Containment Water Level	• Flooding of equipment and instruments • Safety injection of spray recirculation possible • Spillover to the reactor cavity • Ability to quench dispersed core debris	• Wide range containment water level	• Refueling water storage tank level • Narrow range sump level
Containment Flammable Gas Concentration	• Containment gas flammability	• Containment hydrogen monitor	• Sampling • Calculational aids
Containment Pressure	• Containment over-pressurization	• Wide range containment pressure	• Containment temperature

*Three monitored variables each

The scenarios used in this study are characterized using the 14 monitored variables in Table 1. These variables are obtained from MELCOR for 1000 time divisions during the simulation. Each scenario is represented by the matrix, 14×1000. For each scenario, a RASCAL analysis [13] was also performed to determine the associated offsite doses. To simplify the analysis, two key radionuclides, Cs-137 and I-131 were used to characterize environmental impacts. A single pre-defined meteorology was used. In an actual application of OST, the meteorology would be known at the time of the event. The RASCAL code is used to assess the radiation dose as a function of time that would be experienced by an individual located within two miles and ten miles of the plant.

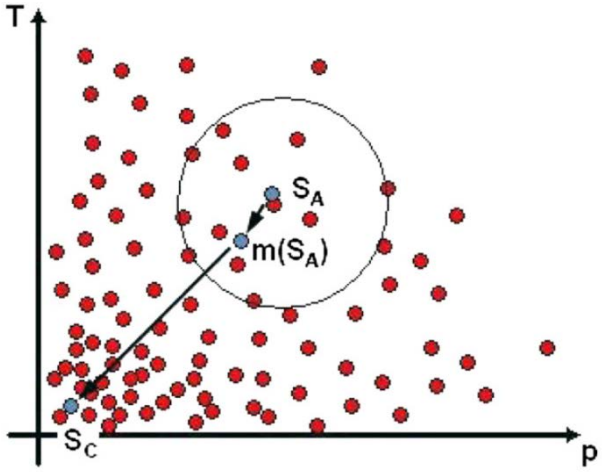
3.2. Dataset selection

For the present study, the 2656 scenarios obtained from [1] (see Section 1) were labeled as the following: i) Total Effective Dose Equivalent (TEDE) is greater than 10 rem (Bin over 10rem), and, ii) TEDE less than or equal to 10 rem (Bin 0-10rem). Using this classification scheme, 116 scenarios were found to fall in Bin 0-10rem and the remainder in Bin over 10rem. 186 scenarios were selected as the training set and 52 testing set. The rest of the data were used for validation. 2656 MELCOR realizations cover 2% of the total probability space. Given the occurrence of the initiating event, 98 percent of the

time the system would be recovered with no offsite release based on the dynamic analysis that was performed.

Instead of random sampling scenarios for training and testing, a data mining process was utilized for clustering the data in Bin over 10rem and Bin 0-10rem. Clustering techniques are particularly useful when identifying groups of scenarios with similar behavior or when classifying their characteristics. In this paper, the mean shift methodology (MSM) is adopted for clustering. The main advantage of MSM is that it is capable of handling large data sets The MSM algorithm assigns each point (in our case snapshots of scenarios in their state space at each time point) to the cluster centroid depending on the bandwidth of the chosen kernel through a set of local averaging operations [14]. This procedure is iterated until all the points in the dataset obtain the assigned clusters. Figure 2 and Eq. (1) illustrate clustering of two-dimensional data into a number of bins with radius or bandwidth h. Intuitively, larger bandwidth generates a fewer number of clusters.

Figure 2 Determination of clustering center (S_c) using Mean shift algorithm (S_A: original point, $m(S_A)$: center of mass or weighted average) in T vs. p [15]



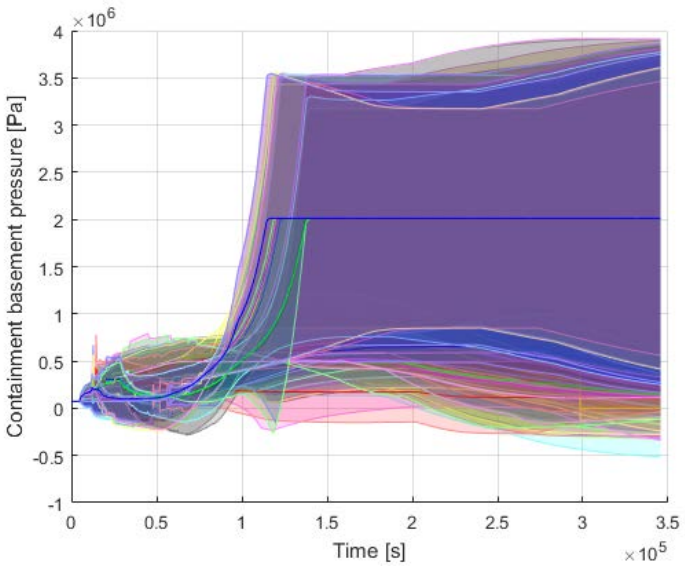
$$m(S_A) = \frac{\sum_{i=1}^{I} \overrightarrow{x_i}\, g\left(\left\|\frac{S_A - \overrightarrow{x_i}}{h}\right\|^2\right)}{\sum_{i=1}^{I} g\left(\left\|\frac{S_A - \overrightarrow{x_i}}{h}\right\|^2\right)} \quad (1)$$

In Eq. (1), x_i is a data point (scenario) of location in the space of possible scenarios and I represents the total number of scenarios. S_A is the initial estimation of location (original point). A Gaussian kernel $g(\vec{x})$ is used for weighing the distance between S_A and x_i is defined by

$$g(\vec{x}) = e^{-\|\vec{x}\|^2/h^2}. \quad (2)$$

The procedure is repeated until the centroids of clusters converge within a given error. From each cluster, the centroid and specific scenarios within that cluster can be identified. In this study, Bins over 10rem and Bin 0-10rem use the same bandwidth. Implementation of MSM on the data from [1] has led to 67 clusters for Bin over 10rem and 30 clusters for Bin 0-10rem. Figure 3 shows the response of one of the monitored variables, containment basement pressure, for all clusters in Bin over 10rem. The solid lines represent the centroid of each cluster and the different colored shades show the whole spectrum in each cluster.

Figure 3 Containment basement gauge pressure for all clusters

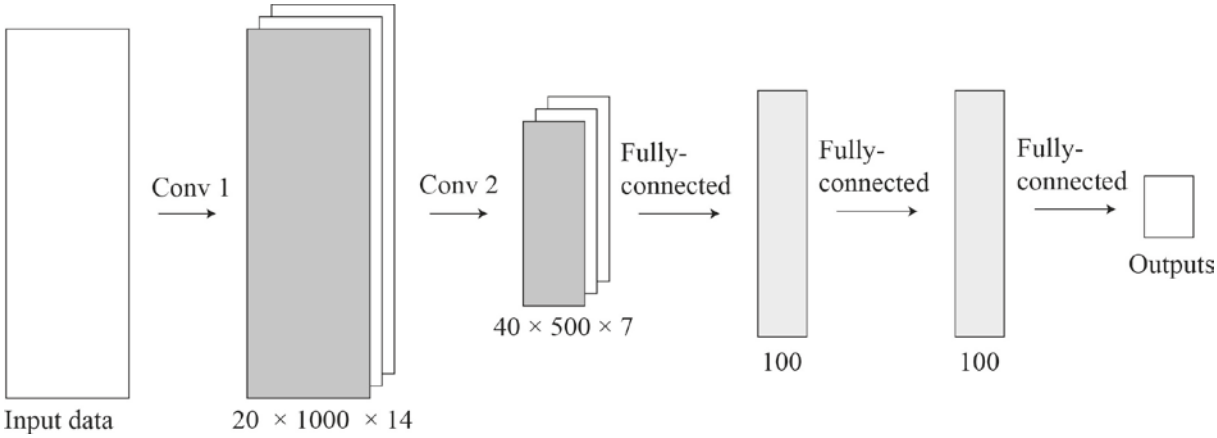


3.3. Software overview: Caffe

An open source computer code Caffe [16], which is a framework developed by Berkeley Vision and Learning Center (BVLC) with a primary focus on pattern recognition of visual objects, is used for the development of OST. Caffe is particularly attractive for a problem with a large set of data as it is capable of processing over 60 million images per day with a PC equipped with NVIDIA K40 GPU. The code is written in C++ and also developed for Matlab and Python. Expressive architecture can be easily utilized with input parameters and doesn't require hard-coding [4].

The simplest architecture of the neural network that can be generated by Caffe consists of one hidden layer between input and output layer. The convolutional neural network (CNN) is one of the well-known deep neural networks. The deep neural network is a neural network with multiple hidden layers between input and output layers [17]. A CNN can stack with one or more convolution layers and one or more fully connected layers at the ends. The main parameters of a convolution layer are referred to as filter parameters. The structure of the filter is width × height × depth. The width and height are user-defined parameters and the depth extends over the input volume. Each of the filters will produce a 2-D activation map and stacked activation maps will be the input to the next layer [11]. The fully connected layer is a type of layer which has full connections between previous and current node. The input and output can be treated each as a vector that the fully connected layer is not spatially located. The deep CNN used in this paper involves a modification to the architecture from Alexnet, which is one of the verified models from the Caffe model zoo [4, 18]. The Alexnet contains eight layers with weights, five convolutional layers, and three fully connected layers. The overall architecture of our CNN consists of five layers and it can be seen in Figure 3. The first two layers are convolutional layers and three fully connected layers are stacked aside. The last fully connected layer is producing the resultants of 2 output labels which are Bin 0-10rem and Bin over 10rem. The first convolutional layer filters the input data with 20 kernels of size 1 × 1 × 1 with a stride 1. Then, the output of the first layer is taken as an input of the second layer and the second convolutional layer filters with 40 kernels of size 5 × 5 × 500.

Figure 4 The architecture of the network



4. Results

The scenarios are defined in terms of variables listed in Table 1 whose values were selected at 1000 times during the simulation time. The set of scenarios available for this case study was partitioned into three sets, the training set (consisting of Bins over 10rem and 0-10rem), testing set (consisting of Bin over 10rem and Bin 0-10rem), and validation set (Bin over 10rem) as indicated in Section 3.2. For the set of exposures in the proximity of the plant (within two miles) the Bin over 10rem of the training set was constructed by random sampling 3% of each cluster (see Section 3.2 for the clustering scheme) and Bin 0-10rem by random sampling 90% of each cluster for balancing the number of scenarios in each bin. The remaining scenarios in Bin over 10rem constituted the validation set. Thus, the total number of scenarios in the training set is 186, with 52 for testing, and 2418 for validation (Table 2). In a similar manner for the region extending to 10 miles from the plant, Bin over 10rem of the training set was constructed by 21% of each cluster and Bin 0-10rem by 90% of each cluster. The total number of scenarios in training, testing, and validation is for this case are tabulated in Table 3.

Table 2 Data distribution of the training, testing, and validation sets for exposures within 2 miles

Train		**Test**		**Validate**	**Total**
Bin over 10rem	Bin 0-10rem	Bin over 10rem	Bin 0-10rem	Only Bin over 10rem	
96	90	26	26	2418	2656

Table 3 Data distribution of the training, testing, and validation sets for exposures within 10 miles

Train		**Test**		**Validate**	**Total**
Bin over 10rem	Bin 0-10rem	Bin over 10rem	Bin 0-10rem	Only Bin over 10rem	
450	421	69	69	1647	2656

As mentioned in Section 3.3, the CNN used in this study was based on Alexnet which competed in the ImageNet Large Scale Visual Recognition Challenge in 2012 [18]. This deep CNN consists of five convolutional layers, three max-pooling layers and three fully-connected layers with 1000 different classes. The Alexnet network has been modified to apply to our study. Our deep neural network has two convolutional layers with each max-pooling layer and three fully-connected layers with 2 different classes as shown in Figure 4. In order to account for the randomness caused by the initialization of weights and bias in Caffe, we repeated the experiments to improve accuracy.

The resulting number of false negatives (FNs) (belonging to Bin over 10rem but labeled as belonging to Bin 0-10rem), false positives (FPs) (belonging to Bin 0-10rem but labeled as belonging to Bin over 10rem), true negatives (TNs) (belonging to Bin 0-10rem and labeled as belonging to Bin 0-10rem), and true positives (TPs) (belonging to Bin over 10rem and labeled as belonging to Bin over 10rem) for testing, validation, and testing plus validation cases are presented in Table 4. In the testing set of 2 mile transport, there were 6 false positives out of a total of 2470 scenarios and the rest of data

were labeled correctly. The validation set of 2 mile transport contains only one mislabeled scenario, a FN, which was incorrectly labeled as Bin 0-10rem instead of Bin over 10rem. However, for the 10 mile transport case, the CNN is less accurate than for 2 mile transport. There were 6 FN and 6 FP out of a total 138 testing scenarios. 151 FN are in the validation set. Note that there are no FPs and TNs in the validation set for either the 2 mile or 10 mile transport cases. Figure 4 and 5 show the testing plus validation set of TPs, TNs, FPs, and FNs for 2 mile and 10 mile transport. Note that the numbers of FNs, FPs, and TNs are substantially smaller than TPs. In Figures 5 and 6, the bar graph of a contingency table (Part (a)) and the one with normalized probability as obtained from the DET analysis (Part (b)) of each category are presented. To the operator of the plant who is responsible for ordering a site emergency and advising state officials on whether or not to order an evacuation, Part (b) of Figures 5 and 6 showing the probabilities of different consequence levels is of greater relevance.

Table 4 Contingency table of testing, validation, and testing + validation set

	FN		**FP**		**TN**		**TP**	
	2 mile	10 mile	2 mile	10 mile	2 mile	10 mile	2 mile	10 mile
Testing	0	8	6	10	20	59	26	61
Validation	1	143	0	0	0	0	2417	1504
Testing + Validation	1	151	6	10	20	59	2443	1565

Figure 5 Testing + Validation set (2 miles)

(a) Graphical representation of contingency table

(b) Contingency table with normalized probability

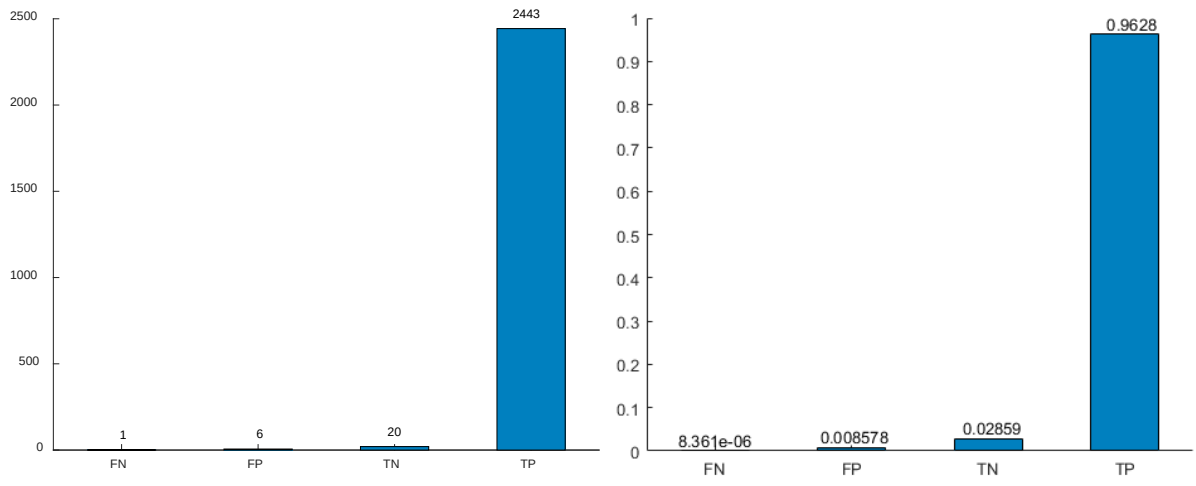
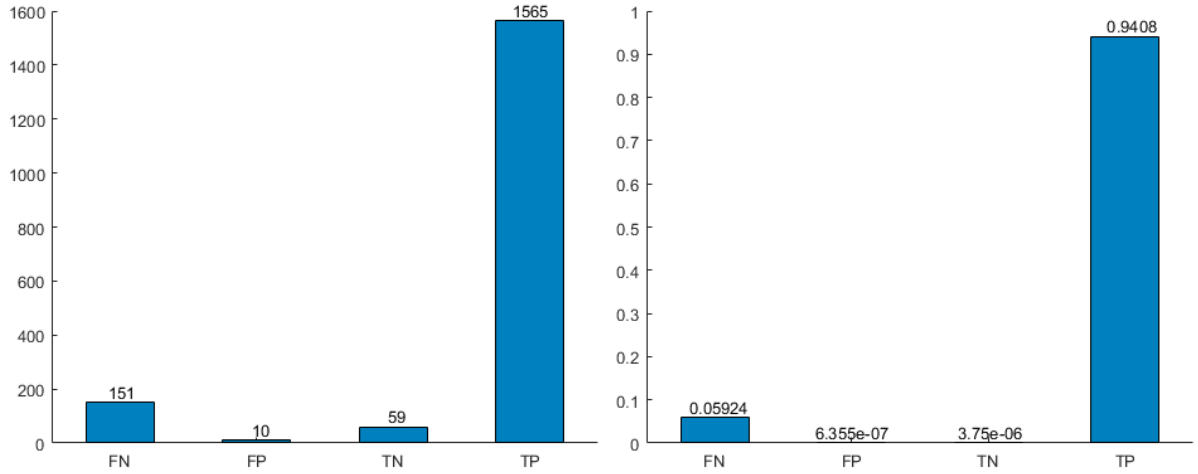


Figure 6 Testing + Validation set (10 miles)

(a) Graphical representation of contingency table

(b) Contingency table with normalized probability



We have also determined the precision, recall, f-measure, and accuracy for this case study. These three parameters are computed, respectively, using Eqs. (3) through (6) [19]:

$$\text{Precision} = \frac{\#true\ positives}{\#true\ positives + false\ positives} \tag{3}$$

$$\text{Recall} = \frac{\#true\ positives}{\#true\ positives + false\ negatives} \tag{4}$$

$$\text{f} - \text{measure} = 2 \cdot \frac{Precision \times Recall}{Precision + Recall} \tag{5}$$

$$\text{Accuracy} = \frac{\#true\ positives + \#true\ negatives}{\#\ total} \tag{6}$$

Results of Equations. (3) - (6) are tabulated in Table 5. Validation row in Table 5 shows that this experiment has resulted in over 90 % accuracy for both case studies. We also observe from Table 5 that the DL algorithm we have developed is superior in finding the scenarios that would lead to undesirable radiological impact (Bin over 10rem) rather than which do not (Bin 0-10rem). Note that recall of testing is equal to 1 because a FN does not exist. For a similar reason, the precision of validation is 1 because FPs do not exist.

Table 5 Accuracy, precision, recall, and f-measure of testing and validation set

	Accuracy		**Precision**		**Recall**		**f-measure**	
	2 mile	10 mile	2 mile	10 mile	2 mile	10 mile	2 mile	10 mile
Testing	0.8846	0.8696	0.8125	0.8592	1	0.8841	0.8966	0.8714
Validation	0.9996	0.9132	1	1	0.9996	0.9132	0.9998	0.9546
Testing + Validation	0.9972	0.9098	0.9976	0.9937	0.9996	0.9120	0.9986	0.9511

5. CONCLUSION

In this paper, we present a real-time tool to assist the NPP operators in predicting the likelihood of future states of the NPP to support the declaration of a site emergency and to assist in the emergency response. DL techniques are used to project the radiological outcomes to the public. The input data to the tool consisted of the temporal behavior of monitored data in the control room along with the training of the tool using the MELCOR/RASCAL codes. The data are from the simulation of an actual accident scenario with possible outcomes. The applied DL network was modified from Alexnet. The results show that the network achieves 99.76% of precision, 99.96% of recall and 0.9986 f-measure for the 2 mile transport case and 99.37% of precision, 91.20% of recall and 0.9511 f-measure for the 10 mile transport case which indicate that the DL network developed in this paper could produce a very credible OST.

Acknowledgments

This research has been performed using funding received from the U.S. Department of Energy Office of Nuclear Energy's Nuclear Energy University Programs. Our thanks go to Dr. Douglas Osborn of Sandia National Laboratories for allowing the use of his data for this study.

References

1. Osborn, D.M., et al. *Seamless Level 2/Level 3 Dynamic Probabilistic Risk Assessment Clustering*. in *ANS PSA 2013 International Topical Meeting on Probabilistic Safety Assessment and Analysis," Columbia, SC (Sep 2013)*. 2013.
2. Gauntt, R., et al., *MELCOR computer code manuals.* Sandia National Laboratories, NUREG/CR, 2005. **6119**.
3. LeCun, Y., Y. Bengio, and G. Hinton, *Deep learning.* Nature, 2015. **521**(7553): p. 436-444.
4. Jia, Y., et al. *Caffe:*
Convolutional architecture for fast feature embedding. in *Proceedings of the 22nd ACM international conference on Multimedia*. 2014. ACM.
5. Catalyurek, U., et al., *Development of a code-agnostic computational infrastructure for the dynamic generation of accident progression event trees.* Reliability Engineering & System Safety, 2010. **95**(3): p. 278-294.
6. Commission, N.R., *Severe accident risks: an assessment for five US nuclear power plants*. 1991, Nuclear Regulatory Commission.
7. Aldemir, T., et al., *Dynamic reliability modeling of digital instrumentation and control systems for nuclear reactor probabilistic risk assessments.* NUREG0CR-6942, US Nuclear Regulatory Commission, 2007.
8. Russell, S.J., et al., *Artificial intelligence: a modern approach*. Vol. 2. 2003: Prentice hall Upper Saddle River.
9. Crevier, D., *AI: the tumultuous history of the search for artificial intelligence*. 1993: Basic Books.
10. Newell, A., J.C. Shaw, and H.A. Simon, *Elements of a theory of human problem solving.* Psychological review, 1958. **65**(3): p. 151.
11. Goodfellow, I., Y. Gengio, and A. Courcille, *Deep Learning*. Adaptive Computation and Machine Learning. 2016: The MIT Press.
12. Rempe, J., D. Knudson, and R. Lutz, *Scoping Study Investigating PWR Instrumentation during a Severe Accident Scenario*. 2015, Idaho National Laboratory (INL), Idaho Falls, ID (United States).
13. Ramsdell, J., G. Athey, and J. Rishel, *RASCAL 4.3 User's Guide (Draft).* US Nuclear Regulatory Commission, 2013.

14. Fukunaga, K. and L. Hostetler, *The estimation of the gradient of a density function, with applications in pattern recognition.* IEEE Transactions on information theory, 1975. **21**(1): p. 32-40.
15. Mandelli, D., et al., *Scenario Clustering and Dynamic Probabilistic Risk Assessment.* Reliability Engineering and System Safety, 2013(115): p. 146-160.
16. Jia, Y. and E. Shelhamer. *Caffe*. [cited 2016 November 13]; Deep learning framework]. Available from: http://caffe.berkeleyvision.org/.
17. Bengio, Y., *Learning deep architectures for AI.* Foundations and trends® in Machine Learning, 2009. **2**(1): p. 1-127.
18. Krizhevsky, A., I. Sutskever, and G.E. Hinton. *Imagenet classification with deep convolutional neural networks*. in *Advances in neural information processing systems*. 2012.
19. Van Rijsbergen, C., *Information retrieval. dept. of computer science, university of glasgow.* URL: citeseer. ist. psu. edu/vanrijsbergen79information. html, 1979.