Exploring Geometric Feature Hyper-Space in Data to Learn Representations of Abstract Concepts

Rahul Sharma \(^1\)\(^\ddagger\), Bernardete Ribeiro \(^1\) Alexandre Miguel Pinto \(^1\)\(^\ddagger\) and F. Amílcar Cardoso \(^1\)

\(^1\) CISUC - University of Coimbra, Portugal; rahul@dei.uc.pt, bribeiro@dei.uc.pt, ampinto@dei.uc.pt, amilcar@dei.uc.pt

\(^\ddagger\) These authors contributed equally to this work.

Abstract: The term Concept has been a prominent part of investigations in psychology and neurobiology where, mostly, it is mathematically or theoretically represented. The Concepts are also studied computationally through their symbolic, distributed and hybrid representations. The majority of these approaches focused on addressing concrete concepts notion, but the view of the abstract concept is rarely explored. Moreover, most computational approaches have a predefined structure or configurations. The proposed method, Regulated Activation Network (RAN), has an evolving topology and learns representations of Abstract Concepts by exploiting the geometrical view of Concepts, without supervision. In the article, the IRIS data was used to demonstrate: the RAN’s modeling; flexibility in concept identifier choice; and deep hierarchy generation. Data from IoT’s Human Activity Recognition problem is used to show automatic identification of alike classes as abstract concepts. The evaluation of RAN with 8 UCI benchmarks and the comparisons with 5 Machine Learning models establishes the RANs credibility as a classifier. The classification operation also proved the RAN’s hypothesis of abstract concept representation. The experiments demonstrate the RANs ability to simulate psychological processes (like concept creation and learning) and carry out effective classification irrespective of training data size.

Keywords: unsupervised machine learning; hierarchical learning; computational representation; computational cognitive modeling; contextual modeling; classification; IoT data modeling

0. Introduction

Concepts are of great value to humans because they are one of the building blocks of our recognition process. They enable us to perform cognitive functions such as classification which is fundamental in decision making and also capacitate us for contextual comprehension. By definition, a concept refers to an ‘idea’ or a combination of several ideas but in the computational domain, a concept can be a feature (object or event) or set of features (objects or events). An individual concept is referred to as a concrete concept (or feature) whereas a generalized form of a set of concepts (or features) can be perceived as an abstract concept. There are several conceptual representation theoretical frameworks [1] like modality-specific, localist-distributed, experience-dependent [2].

In computational domain, the concepts are mostly represented by three broad categories i.e. symbolic (eg. ACT-R [3]), distributed (eg. ANN) and spatial (eg. Conceptual Space [4]) representations. Cognitive architecture like CLARION [5] is an example of a hybrid computational representation that combines symbolic and distributed approaches, but there is no hybrid approach that combines all the three representations. Moreover, the symbolic, distributed, spatial and hybrid (spatial+distributed) representations are mostly used on representing concrete concepts (like object detection) whereas the notion of an abstract concept is debated [1] but rarely explored.
This article proposes a computational method named Regulated Activation Network (RAN) which unifies the virtues of symbolic, distributed and spatial representations to represent concepts (both concrete and abstract). RAN has a graph-based topology hence it is distributed, every node in the graph (network) identifies an entity, therefore, it’s symbolic, and every node (or entity) is viewed in an n-dimensional feature space, hence, it’s also spatial. The spatial view of concepts as points in multidimensional geometric feature space (see Figure 1 for 6-dimensional View of Concepts) is inspired by the theory of conceptual spaces [4]. The RAN’s modeling has an evolving topology that enables it to build a model depicting a hierarchy of concepts. The geometrical associations among concepts aid in determining the Convex Abstract Concepts. Further, the representatives (nodes) of the Abstract Concepts form a new layer dynamically, where each node acts as a Convex Abstract Concept representative for the underlying category. Symbolically, the concepts at (relatively) lower level in the hierarchy are identified as concrete concepts and the concepts at (relatively) higher level are seen as abstract concepts.

The model generation process with RAN and the three cognitive functions (i.e. concept creation, learning and activation propagation) are simulated using a IRIS data. The deep hierarchy generation, automatic generic concept modeling simulations are performed using 2 UCI benchmark: IRIS data; and IoT data from smartphone sensors. The application of RAN as a classifier is reported along with the proof of concept of classification using 8 UCI benchmark datasets. The generated models were evaluated using metrics precision, recall, F1-score, accuracy and Receiver Operating Characteristic (ROC) curve analysis. The article also reports the RANs classification and feature comparison with five machine learning techniques, Multilayer Perceptron (MLP) [6], Logistic Regression (LR) [7], K Nearest Neighbors (K-NN) [8], Stochastic Gradient Descent (SGD) [9] and Restrict Boltzmann Machine [10] pipelined with Logistic Regression (RBM+).

The model generation process with RAN and the three cognitive functions (i.e. concept creation, learning and activation propagation) are simulated using a IRIS data. The deep hierarchy generation, automatic generic concept modeling simulations are performed using 2 UCI benchmark: IRIS data; and IoT data from smartphone sensors. The application of RAN as a classifier is reported along with the proof of concept of classification using 8 UCI benchmark datasets. The generated models were evaluated using metrics precision, recall, F1-score, accuracy and Receiver Operating Characteristic (ROC) curve analysis. The article also reports the RANs classification and feature comparison with five machine learning techniques, Multilayer Perceptron (MLP) [6], Logistic Regression (LR) [7], K Nearest Neighbors (K-NN) [8], Stochastic Gradient Descent (SGD) [9] and Restrict Boltzmann Machine [10] pipelined with Logistic Regression (RBM+).

The article is organized in the following order; Section 1 puts forward the work closely related to Abstract Concept representation and models with evolving topology. Section 2 describes the background associated with principles, theories, and motivations for RAN’s modeling. RANs methodology is detailed using a IRIS data in Section 3. Section 4 shows the experiments with two datasets acquired from UCI machine learning repository to exhibit (1) flexibility in choosing a suitable concept identifier, (2) building a deep hierarchy of Abstract Concepts, (3) automatic association of
input-labels to their respective Abstract Concept nodes. Section 5 provides RANs comparisons with five classifiers and proof of concept with eight benchmark datasets. At last, Section 6 summarizes and concludes the article with remarks over ongoing and future work.

1. Related Work

Abstract Concepts are of immense value because they help in developing unique abilities in humans such as relative recognition and effective decision-making. In medical science, there have been significant efforts to study Abstract Concepts with the help of technology. One such example is MRI\(^1\), which is being used to inspect the sections of the brain involved in Abstract Concept identification [11, 12]. Research in psychology has also reported investigations over Abstract Concepts, like probing the role of emotional content in processing and representing Abstract Concepts [13].

There has been a notable contribution from cognitive, and psycholinguists in studying languages through Abstract Concept modeling and representations. Internally representing Abstract Concepts via amodal symbols like a feature list, and frames [14,15] is among the preliminary research work in linguistics. The association and context were also established, to relating Abstract and Concrete words [14]. Some research reveals that we internally recognize metaphors as Abstract Concepts [16]. Besides theoretical methods, computational approaches are playing a vital role in comprehending and representing Abstract Concepts. Research in NLP addresses computational learning, comprehension and processing of human understandable language, and its components. An interesting article published a work about the representation of Abstract, and Concrete Concepts in daily written Language using a text-based multimodal architecture of NLP [17]. Other than NLP, semantic networks are also used to study semantic similarity among Abstract, and Concrete nouns (of Greek, and English) [18] with the aid of network-based Distributed Semantic Model [19].

Though the aforementioned computational approaches contribute toward Abstract Concept modeling and representation, they have a fixed topology (i.e., the modeling process begins with a fixed structure and configuration). In connectionist computational modeling, there have been efforts to develop models that evolve. ANNA ELEONORA (standing for Artificial Neural Networks Adaptation: Evolutionary LEarning Of Neural Optimal Running Abilities) [20] demonstrated a way to grow neural networks with the aid of parallel genetic algorithms. NEAT (NeuroEvolution of Augmenting Topologies) [21] is another work that reported evolving neural network modeling, showing how nodes and weights are added to the model when new features emerge as part of the existing population and CoDeepNEAT [22] is the most recent member of such evolving models. Markov Brains [23] also belong to the family of evolving neural networks which uses binary variables and arbitrary logic to implement deterministic or probabilistic finite state machines. They have been used to investigate behaviors, character recognition and game theory.

This article communicates an approach which is not only hybrid but also has an evolving topology. The RANs modeling learns the representation of the Convex Abstract Concepts dynamically, hence makes it an evolving topology. RANs approach is connectionist, and each newly created node corresponds to an Abstract Concept symbolically, thus portraying its hybrid characteristics.

2. Background

This section provides information about the principles and methodologies related to RANs modeling. It highlights the significance of each approach, along with their applicability in RANs modeling.

---

\(^1\) Magnetic Resonance Imaging
2.1. Principles of Regulated Activation Networks

The tenets of RANS modeling, presented in [24], states model should be topologically a connectionist and intends to represent and simulate the dynamic cognitive state of an agent. In the first version RANs [24] the authors implemented a single-layer version of the model where each node had a lateral connection to its same-layer companions. It had a simple learning and reasoning mechanisms, but these showed to be sufficient to simulate several known cognitive phenomena such as the Priming [25], the False Memory [26,27].

Two principles of Regulated Activation Networks inspired our proposal. First, the model should be dynamic, and this is achieved by dynamically creating layers (deep representations) of Concepts. Second, the model must be capable of learning and creating an Abstract representation of Concepts. This is obtained by viewing associations among the Concepts (at the same level) in n-dimensional geometric space, and learning relationship between the newly created Abstract Concepts, and input level Concepts.

2.2. Conceptual Spaces

Conceptual Spaces theory [4] is one of the cognitive approaches that form the basis of RANs modeling. This theory views the Concepts as regions within a multi-dimensional space, with the data features representing the dimensions. The similarity among the Concepts can be identified based upon the geometrical distance between the objects. The Conceptual Spaces, thus, serves as a natural way or tool to capture the similarity relationships among Concepts, or Objects. Under this setting, one data instance corresponds to a single point in the space. Formally we can say, the Quality Dimensions, i.e., a set of $D_1, ..., D_n$, forms the Conceptual Space $S$. A point in $S$ is represented by a vector $v=\langle d_1, ..., d_n \rangle$, where $\{1, ..., n\}$ are the indexes of the dimensions. Atomic Concepts are Convex Regions –a Convex Region $C$ having point $x$ that falls between points $x_1 \in C$ and $x_2 \in C$ also belongs to $C$. The quality dimension is the basic requirement for Conceptual Spaces [28]. An example is a color space with the dimensions Hue, Saturation, and Brightness. Each quality dimension has a geometrical structure. For example, Hue is circular, whereas brightness and saturation correspond with finite linear scales (see Figure 2).

![Figure 2. The color space [29]](image)

The theory of Conceptual Spaces also addresses prototype theory of categorization [30–32]. The main idea of prototype theory is that within a category of objects, like those instantiating a Concept, certain members are judged to be more representative of the group than others. For example, robins are judged to be more representative of the category “bird” than are ravens, penguins, and emus. If Convex Regions of Conceptual Space describes Concepts, then prototype effect is, indeed, expected, i.e., the most likely central position of a Convex Region describes an Abstract Concept. For example, if color Concepts in a Convex region identified as subsets of the color space, then the central points of these regions would be the most prototypical examples of the color.

Clustering is a suitable way of identifying and learning atomic Convex Concepts in conceptual spaces. There are several clustering techniques, like hierarchical clustering, subspace clustering [33], partitioning relocation clustering, density-based clustering, grid-based clustering and many more.
Table 1. Notations

| Notation | Description |
|----------|-------------|
| $W$      | Inter-Layer weight matrix |
| $A$      | Output Activation |
| $a$      | Input Activation |
| $n_a$    | Number of elements in input vector at Layer $l$ |
| $n_A$    | Number of elements in output vector at Layer $l + 1$ |
| $l$      | $l$'th Layer representative |
| $d$      | Normalized Euclidean distance |
| $C$      | Cluster center or Centroids |
| $i, j, k$ | Variables to represent node index for input-level, abstract-level, and arbitrary node index in either of the levels, respectively |
| $t$      | Iterator variable |
| $f(x)$   | Transfer function to obtain similarity relation |

Many are frequently used in the statistical and scientific analysis of data [34,35], and in machine learning for the identification of Concepts/features [36]. On the other hand, the creation of a hierarchy of sub/super-Concepts is a way to represent more Abstract Concepts and their taxonomic-like relations. Deep learning techniques [37–40,41] found in the literature can also be used to create deep hierarchical representations, but usually do not interpret data as points in Conceptual Spaces. In the proposed approach, the clustering techniques enable us to identify categories of Concepts in a Conceptual Space thus laying the foundation to form a layer of Abstract representation of Concepts.

2.3. Spreading Activation

Spreading Activation is a theory of memory [42] based on Collins and Quillian’s computer model [43] which has been widely used for the cognitive modeling of human associative memory and in other domains such as information retrieval [44]. It intends to capture the information representation and how it is processing. According to the theory, long-term Memory is represented by nodes and associative links between them, forming a semantic network of Concepts. The links characterized by a weight denotes the associative or semantic relation between the Concepts. The model assumes activating one Concept implies the spreading of activation to related nodes, making those memory areas more available for further cognitive processing. This activation decays over time as it spreads, which can occur through multiple levels [45], and the further it gets the weaker it becomes. That is usually modeled using a decaying factor for activation. The method of spreading activation has been central in many cognitive models due to its tractability and resemblance of interrelated groups of neurons in the human brain [46]. This theory of Spreading Activation inspires the activation propagation mechanism in our proposal to propagate (spread) activation in the upward direction, i.e., from the input-to-abstract layer in the network. The method has its significance, i.e., in the creation of the network, and in understanding the created Abstract Concepts.

3. Abstract Concept Modeling with RANs

The data value used with RANs modeling should be between “0” and “1” (both inclusive). This limitation has its inspiration from biological neurons, a value “0” indicates neuron (or node) is inactive, whereas “1” shows the neuron is highly active. An additional header is also needed for modeling with RAN. The size of the header is the same as the dimension of the input data vector, and each header element holds the largest value of their corresponding input data attribute. See Section A.1 for elaboration. RANs works with multivariate datasets except image because pictures are not ideal.
candidates to be interpreted as points in conceptual spaces, (discussed in Section 2.2). For this reason, our approach will, most probably, underperform on image processing tasks against other models that are, individually, designed for this kind of data, such as deep representations built with Convolutional Networks [40,47,48]; our technique is preferably suitable for understanding and simulating cognitive processes like Abstract Concept Identification.

The proposed approach models Convex Abstract Concepts through four core steps (i.e., Concept Identification, Concept Creation, Interlayer Learning and Upward Activation Propagation), along with one optional step (i.e., Abstract Concept Labeling). The RAN’s methodology is explained using benchmark IRIS dataset. Figure 3 shows the parallel coordinate plot of IRIS data normalized between $[0, 1]$ using min-max technique. The plot also shows the Cluster Representative Data Points (CRDPs) for all three classes of IRIS data (the importance of CRDP is detailed in 3.1). The objective of this experiment is to show how RANs build a hierarchical representation dynamically and simulate cognitive process of concept creation, learning, and activation propagation. For this experiment, it was hypothesized that the created abstract concepts symbolically represents the three classes of IRIS data. Classification operations were performed to prove the hypothesis which are reported at the end of this section.

3.1. Step 1: Concept Identification (CI) Process

The concept identification is the process of identifying convex groups in the input data. This is realized by categorizing the input data based upon their geometrical relationship, i.e., distance, conforming to the theory of conceptual spaces (see Section 2.2). The quality-dimension (i.e. SepalLength, SepalWidth, PetalLength and PetalWidth attributes of input data) symbolically represents input nodes (i.e. $S_1$, $S_2$, $S_3$ and $S_4$ see Figure 4). In this experiment, K-means [49] clustering method is used a concept identifier and applied to determine the convex groups in the IRIS data. The K-means was configured to determine the 3 classes (i.e. Iris-setosa, Iris-virginica, and Iris-Versicolor) of IRIS data. The clustering operation also determines the three cluster centers as Cluster Representative Data Points (CRDPs). According to the theory of prototype (see Section 2.2) these three CRDPs are the most probable representative of the three convex groups respectively, therefore are of great importance in learning relationship among concepts in two adjacent layers (see Section 3.3).

Any clustering algorithm can act as a Concept Identifier in RANs modeling if it suffices two basic requirements. First, the algorithm is able to determining Convex categories based upon their geometric relationship among the data instances. Second, the algorithm recognizes CRDPs of all the...
identified clusters. This flexibility of choosing a suitable method for Concept Identification process in RANs modeling is demonstrated by a separate experiment using Affinity propagation [50] clustering algorithm, in Section 4.1.

3.2. Step 2: Concept Creation (CC) Process

Concept creation is a cognitive process to create representation of a newly identified concept. In RANs this cognitive process is simulated by creating a new layer of concepts dynamically. Each constituent node in the new layer symbolically acts as an abstract representative of their respective categories identified in the CI process. Step-2 in Figure 4 shows the newly created layer (Layer-1), that has 3 nodes (N₁, N₂ and N₃), corresponding to 3 classes (i.e. Iris-setosa, Iris-virginica, and Iris-Versicolor) of IRIS data (see Figure 3), identified in CI operation.

3.3. Step 3: Inter-Layer Learning (ILL) Process

Learning is an important cognitive process it acts as a relationship to associate concepts. In RANs modeling, learning is simulated by an assignment operation. As aforestated in Section 3.2 that each node in the new layer is an Abstract representative of categories identified in CI process, thus we learn association among the two-layer such that it substantiates the Abstract representation by the nodes at the new layer. Since CRDPs (see Section 3.1) are the most apparent choice as an Abstract representative of a cluster (and adhere to the inspiration from prototype theory); consequently, the CRDPs assigned as an association between the two layers.

Equation 1 shows the general learning in the form of a matrix, where \( W \) is the learned Inter-Layer Weight (ILW) between node \( j \) at new layer (i.e., Layer-1 in Figure 4) and node \( i \) at input layer (i.e., Layer-0). The set of ILWs, from one node \( j \) at new layer to all input nodes \( i \), are the values of CRDP of \( j^{th} \) cluster center (i.e., \( C_j \)) identified in CI process. For instance, cluster center \( C_1 \) (see Figure 3) forms the weight vector \( [W_1,1, W_1,2, W_1,3 \text{ and } W_1,4] \) (ILWs shown by 4 yellow lines in Step 3 Figure 4) between the node \( N_1 \) at Layer-1 and all four input nodes \( S_1, S_2, S_3 \text{ and } S_4 \) at Layer-0.

\[
W = \begin{bmatrix}
W_{1,1}, W_{1,2}, \ldots, W_{1,n_a}, \\
\vdots \\
W_{k,1}, W_{k,2}, \ldots, W_{k,n_a}, \\
\vdots \\
W_{n_A,1}, W_{n_A,2}, \ldots, W_{n_A,n_a}
\end{bmatrix} = \begin{bmatrix}
C_1 \\
\vdots \\
C_k \\
\vdots \\
C_{n_A}
\end{bmatrix}
\]

(1)

Where \( j=1, 2, \ldots, n_A \), and \( i=1, 2, \ldots, n_a \).
3.4. Step 4: Upwards Activation Propagation (UAP) Process

This upward activation propagation is a geometric reasoning operation, i.e., a non-linear projection of an $i$-dimensional input data vector $a_i$ into a $j$-dimensional output vector $A_j$ (see Step 4 in Figure 4). The UAP operation is carried out in two stages, in the first stage the geometric distance operation takes place, and in the second stage, geometric distance is translated to establish a similarity relation.

3.4.1. Geometric Distance Function (GDF)- Stage 1

In the first phase of the UAP mechanism we determine the geometrical distance between the learned weight vectors (see Equation 1) and an input instance $a_i$. The numerator of Equation 2 shows a function to calculate the Euclidean distance between the $j$th weight vector and input vector $a_i$. The denominator of Equation 2 shows the relation that normalizes the calculated distance between $[0, 1]$.

$$d_j = \sqrt{\sum_{i=1}^{n} (W_{ij} - a_i)^2}$$

And consequently, $j$ normalized Euclidean distances $d_j$ are obtained between all $j$ weight vectors and input instance $a_i$.

3.4.2. Similarity Translation Function (STF)- Stage 2

In the second phase the calculated normalized distance is transformed to obtain a similarity relation such that following requirements are fulfilled:

- $f(d = 0) = 1$, i.e. when distance is 0 similarity is 100%.
- $f(d = 1) = 0$ i.e. when distance is 1 similarity is 0%.
- $f(d = x)$ is continuous, monotonous, and differentiable in the $[0, 1]$ interval.

$$f(x) = (1 - \sqrt{x})^2$$

In RANs modeling Equation 3 is used as the Similarity Translation Function to determine the similarity relation of the previously calculated distance. The non-linearity of STF is depicted in Figure 5, indicating that the similarity value reduces drastically when the normalized Euclidean distance is larger than 0.05 (or 5% dissimilar).

![Figure 5. Plot of Similarity Translation Function with respect to varying input values in range $[0, 1]$](image)

---

2 In RANs modeling the activation values are, by definition, real values in the $[0, 1]$ interval – in an $n$-dimensional space the maximal possible euclidean distance between any two points is $\sqrt{\sum_{i=1}^{n} (a_i - 0)^2} = \sqrt{n}$, where $a_i=1$. 
Algorithm 1: Upwards Activation Propagation algorithm

**Input:** Vector \([a_1, a_2, ..., a_n]\) as input at layer \(l\).

**Output:** New activation vector \([A_1, A_2, ..., A_n]\) in layer \(l + 1\).

**for** Each node \(A_j\) in layer \(l + 1\) **do**

1. Calculate Normalized Euclidean Distance:
   \[
   d_j = \sqrt{\frac{\sum_{i=1}^n (W_{j,i} - a_i)^2}{n}}
   \]
2. Transform \(d_j\) through STF Equation 3:
   \[
   A_j = f(d_j^2)
   \]

**end for**

Where:

- \(i = [1, 2, ..., n]\).
- \(j = [1, 2, ..., n_A]\).
- \(W_{j,i}\) is ILW see Equation 1.

The first three steps generate the RANs model (see Figure 4), later, in the fourth step, this model is used via UAP operation by propagating the input activation \((a_i)\) upward and obtaining activation \((A_j)\) at Convex Abstract Concept layer (inspired by the theory of spreading activation see Section 2.3).

Algorithm 1 describes the Upward Activation Propagation operation, showing how the inputs and interlayer learning weights \(W\) are used to calculate similarity relation to generating output activation at each Abstract Concept representative nodes. The activation \(A_j\) in newly created nodes \(N_j\) also indicate the degree of confidence (DoC) of the identification of a class by its representative node in the new layer (for a given input data instance). For instance, in Figure 4, Step-2, at Layer-0 input vector is \([0.1, 0.21, 0.12, 0.5]\) it signifies that the dimensions \(S_1, S_2, S_3\) and \(S_4\) has activation 0.1, 0.21, 0.12, and 0.5 respectively. For the, aforementioned, input vector, \([0.13, 0.32, 0.89]\) vector of activation is observed at all nodes (\(N_1, N_2\) and \(N_3\)) respectively, at Layer-1. The observed activation vector itself describes that the input data belongs to Class-3 (Versicolor) with a DoC of 89%.

3.5. RANs Proof of Hypothesis

In the beginning of this Section 3 it was hypothesized that nodes in the newly created layer symbolically represents abstract concepts of the 3 classes (Iris-setosa, Iris-virginica and Iris-versicolor) of Iris data. This hypothesis can be proven through classification operation using the RAN model generated with IRIS data. The classification experiment setup consists of 30 iterations of an experiment. Each experiment consist of 9 Research Design (RD) (see Table A3 in Section A.2), where, in every RD a 10-fold cross-validation procedure was applied. To carry out the evaluation operation **True-labels**, and **Test-labels** are determined via Abstract Concept Labeling (ACL) operation of RANs (see Section A.4 for ACL’s description). Further, these labels were used to form a multi-class confusion matrix for the 3 classes of IRIS data. and with the aid of this confusion matrix 4 metrics (i.e. Precision, Recall, F1-Score, and Accuracy) were calculated.

Multi-class Receiver Operating Characteristics (ROC) curves were also plotted for the 3 classes to support the classification experiment with IRIS data. The binary labels corresponding to the True-labels (obtained via ACL operation) were obtained using the method node-wise binary transformation of input True-label (see Section A.3). Further, the confidence scores for the binary vectors were calculated using the node-wise confidence-score calculation method (described in Section A.3).

The Table 2 not only shows the RAN’s comparison with other 5 classifiers but also that RAN indeed preformed well in the classification process with a performance of 95% (ca.) for all classification
metrics. The ROC curve analysis also observed an Area Under Curve (AUC) of 99.07\% (ca.), 99.40\% (ca.) and 98.75\% (ca.) for IRIS Setosa, Virginica and Versicolor classes respectively. These results show the ability of RAN’s modeling to identify the abstract concept where the three nodes ($N_1$, $N_2$ and $N_3$) in Layer-1 symbolically represents the classes IRIS Setosa, Virginica and Versicolor, respectively, as abstract concepts, hence proves the hypothesis.

4. Behavioral Demonstration of RANs

This section exhibits two distinct aspects of RANs modeling via separate experiments. Both investigations present a different view of RANs methodology, highlighting the capabilities of the RANs approach.

4.1. Experiment with IRIS dataset

There are two objectives of this probe, first is to demonstrate flexibility in choosing an appropriate methodology for Concept Identification operation in RANs modeling (see Section 3.1). Second is to show how RANs modeling can be used to build a deep hierarchy of Convex Abstract Concepts dynamically. This experiment uses Affinity propagation [50] clustering algorithm as a Concept Identifier to support the claim of independence in selecting a suitable clustering method for CI process in RANs modeling. Unlike the K-means algorithm (used to describe the RANs methodology in Section 3), with the Affinity Propagation algorithm, the number of clusters within the data need not be known beforehand. Furthermore, Affinity Propagation conforms to the basic requirements (see Section 3.1) for being a Concept Identifier in RANs modeling.

The second prospect of this experiment is to illustrate the dynamic topology of RANs approach where the network grows to form several layers representing Convex Abstract Concepts. For this demonstration, an algorithm is developed, named Concept Hierarchy Creation (CHC) algorithm (see 2). The CHC algorithm streamlines all four steps of RANs modeling (i.e., CI, CC, ILL and UAP).
and uses these steps iteratively to build a hierarchy of Convex Abstract Concepts as described through Algorithm 2. This experiment was also conducted using the IRIS dataset obtained from the UCI machine learning repository [51]. In the CHC algorithm the Affinity propagation clustering algorithm was initialized with the following parameters: (1) damping_factor (DF) = 0.94 for layers below level 3, DF = 0.9679 for the layers at level 3 and above; (2) convergence_iteration=15; (3) max_iteration=1000.

![Diagram of the model generated with 90% stratified IRIS data using Concept Hierarchy Creation Algorithm.](image)

**Figure 7.** The model generated with 90% stratified IRIS data using Concept Hierarchy Creation Algorithm. Layer-0 is created while initializing the CHC algorithm. The algorithm grew to a Desired-depth of six Layers (including input Layer-0), and in each iteration of CHC algorithm a new layer is created dynamically and the Interlayer weights (ILW) are learned between the existing layer and a newly created layer above it.

Input layer-0 was created, with four nodes (equal to the dimension of IRIS data), and the RANs hierarchy generation is carried out according to Algorithm 2. The model obtained from CHC process is depicted by Figure 7, the model was initialized to grow six layers deep. Therefore, hierarchy augmentation terminates at Layer-5, with Layer-5 identified as most Abstract layer consisting of three nodes acting as Abstract representatives of three categories of flowers of IRIS dataset. To evaluate the obtained RANs model, True-labels, and Test-labels were retrieved using an Abstract Concept labeling procedure (see Section A.4). A confusion matrix (see Figure 8) was generated using the True and Test labels. With the aid of the confusion matrix, Precision, Recall, F1-Score and Accuracy were calculated to evaluate the model. The model performed quite decently with an observed accuracy of 93.33 (ca.), the results of precision, recall and F1-Score are reported in Table 3. The ROC curve analysis of the RANs model, as shown in Figure 9, displays the various operating characteristic and the observed Area Under Curve for all the classes of IRIS data. In this experiment, it is worth mentioning the application of RANs modeling for data dimension transformation and data visualization. In Figure 7 we can observe that the dimension of Layer-0 is four, whereas the size of the other layers either expands or reduces when the network grows. This dimension transformation operation is helpful in addressing...
Algorithm 2 Concept Hierarchy Creation Algorithm

**Input:** Multi-variate data with values between [0,1].

**Output:** Set of layers of Concepts – concept hierarchy.

**Initialization:** Create input layer layer-0 having dimension equal to that of input data.

Set Current-layer-size $CLS = i$, dimension of input-data vector.

Set Layer-count $L = 0$.

Set Desired-depth= 6.

Select Clustering algorithm and initialize.

Set current-data = input-data.

repeat

Run clustering algorithm on current-data to identify set of cluster centers $C$.

Create a new-layer above current-layer, with no nodes.

for each cluster center $C_j \in C$ do

Create new node $j$ in new layer $l+1$.

for each node $i$ in current-layer do

Create a new weighted connection $W_{c_j,i}$ between $c_j$ and $i$ such that $W_{c_j,i}$ is the coordinate of $c$ along the $i$ dimension.

end for

end for

Set new-data = empty data set.

for each datum in current-data do

Inject datum in current-layer

Propagate activation from current-layer to new-layer using algorithm 1.

Add activation pattern produced in new-layer to new-data.

end for

Set $L = L + 1$.

Set $CLS = number of clusters in current-layer$.

Set current-data = new-data.

Set current-layer = new-layer.

until $CLS=1$ OR Desired-depth= $L$.

**Figure 8.** Confusion Matrix generated to validate RANs model with IRIS data (having 9 : 1 train, and test data ratio) for Class-0 (Setosa), Class-1 (Verisicolour), and Class-2 (Virginica).

...the issue of the cures of dimensionality. Besides, the transformed data can be plotted to extract useful information from the data.

4.2. Experiment with Human Activity Recognition Data

This experiment aims to show the ability of the RANs approach to build the representation of generic Concepts. The experiment uses UCIHAR [52] dataset for home activity recognition using the smartphone, obtained from the UCI machine learning repository. The data captured six activities Walking, Walking_upstairs, Walking_downstairs, Sitting, Standing, and Laying. The hypothesis of this experiment is that the labels Walking, Walking_upstairs, Walking_downstairs are identified by an abstract concept (say) Mobile and the other 3 labels Sitting, Standing, and Laying by abstract concept (say) Immobile. In this experiment also classification operation can be used to prove the hypothesis.

The UCIHAR dataset was normalized and a header was attached. In CHC algorithm K-means is chosen as concept identifier and the parameter Desired-depth was set to 1 so that model has only...
Table 3. Evaluation of RANs Model generated through IRIS data

| Class     | Precision (%) | Recall (%) | F1-Score (%) | Support |
|-----------|---------------|------------|--------------|---------|
| Setosa    | 100           | 100        | 100          | 5       |
| Versicolour | 83.33       | 100        | 90.91        | 5       |
| Virginica | 100           | 80         | 88.89        | 5       |
| Avg/Total | 94.44         | 93.33      | 93.26        | 15      |

Figure 9. ROC curve analysis with IRIS dataset (having 9 : 1 train, and test data ratio), for Class-0 (Setosa), Class-1 (Versicolour), and Class-2 (Virginica)

Figure 10. Model generated with RANs approach. Nodes $N_1$ and $N_2$ at Layer-1 represents either of the two Abstract Concepts, i.e. Mobile and Immobile. Each node at Layer-0 represents individual dimensions of input data vector.
Table 4. RAN's Comparative Study for UCIHAR dataset

| Model  | Precision (%) | Recall (%) | F1-Score (%) | Accuracy (%) |
|--------|---------------|------------|--------------|--------------|
| RBM    | 99.68 ± 0.14  | 99.68 ± 0.14 | 99.68 ± 0.14 | 99.68 ± 0.14 |
| K-NN   | 99.96 ± 0.02  | 99.96 ± 0.02 | 99.96 ± 0.02 | 99.96 ± 0.02 |
| LR     | 99.97 ± 0.02  | 99.97 ± 0.02 | 99.97 ± 0.02 | 99.97 ± 0.02 |
| MLP    | 99.96 ± 0.02  | 99.96 ± 0.02 | 99.96 ± 0.02 | 99.96 ± 0.02 |
| RANs   | 99.85 ± 0.01  | 99.85 ± 0.01 | 99.85 ± 0.01 | 99.85 ± 0.01 |
| SGD    | 99.98 ± 0.01  | 99.98 ± 0.01 | 99.98 ± 0.01 | 99.98 ± 0.01 |

for ACL process elaboration), it was observed that Walking, Walking_upstairs, Walking_downstairs classes were mapped to one node of Layer-1. Whereas, the labels Sitting, Standing and Laying traced to the other node in Layer-1. Interestingly, this outcome commensurate with the expectations from this experiment and shows the RANs capability to identify Abstract Concepts in an unsupervised manner naturally. The True-label and Test-label obtained through ACL operation were used to form the confusion matrix, which is later referred to calculate Precision, Recall, F1-Score, and Accuracy for evaluating the generated model. Node-wise binary labels and confidence scores were determined (as described in Section A.3) for both Abstract nodes at Layer-1. Figure 11 shows the Area Under Curve (AUC) observed during the ROC curve analysis of all 10-Folds in different Research Designs. With both these evaluations it is deduced that, apart from building the representation of Abstract Concepts, the model generated with RANs performed satisfactorily.

The RANs modeling was compared with five different types of approaches based upon their classification operation. To carry out the comparative study it was essential to transform the six Labels into binary Labels, because RANs modeling was identifying two Abstract Concept, and its performance was measured based upon them. Thus, with these five approaches, the Labels of the dataset were merged to form two groups, i.e., Walking, Walking_upstairs, Walking_downstairs in Class-1, and Sitting, Standing, and Laying in Class-2. Later the modeling was performed followed by validation and evaluation. Table 4 displays the comparison of all five approaches with RANs modeling. It is observed that RANs approach is competent to these five techniques, with an added advantage of being an unsupervised approach, and ability to build representations of Abstract Concepts.

5. RANs Applicability and Observations

This section highlights the scope of RANs modeling as a classifier w.r.t. distinct domains. To support this ambit of RANs usability, experimental results are reported using eight datasets concerning...
(a) RANs performance with eight different datasets depicting RANs appositeness with data belonging to distinct domains.

(b) Observed Area Under Curve (AUC) while performing ROC curve analysis for RANs model generated with eight different datasets.

Figure 12. RANs performance with eight datasets using Precision, Recall, F1-Score and Accuracy along with ROC-AUC analysis with Eight benchmark datasets [Mice Protein (MP), Breast Cancer 669 (BC1), Breast Cancer 569 (BC2), Credit Approval (CA), IRIS data (ID), Mamographic Mass (MM), Wine Recognition (WR) and Glass Identification (GI)]. The graph 12b shows the plot of percentage AUC for classes 1 to 8. For each dataset class labels of the graph is serially mapped as: Mice protein (c-CS-s [Class-1], c-CS-m [Class-2], c-SC-s [Class-3], c-SC-m [Class-4], t-CS-s [Class-5], t-CS-m [Class-6], t-SC-s [Class-7] and t-SC-m [Class-8]); Mammographic Mass (Benign [Class-1] and Malignant [Class-2]); Credit Approval (Positive [Class-1] and Negative [Class-2]); IRIS (Setosa [Class-1], Versicolor [Class-2] and Virginica [Class-3]); Breast Cancer 569 (Benign [Class-1] and Malignant [Class-2]); Breast Cancer 669 (Benign [Class-1] and Malignant [Class-2]), Wine Recognition (Class-1, Class-2 and Class-3) Glass Identification (Window Glass [Class-1] and Non-Window Glass [Class-2]).
with different areas. A comparative study was also carried out using these datasets to match RANs classification ability with five different classifiers.

Among the eight datasets, the Mouse Protein [53], Mammographic Mass [54], Breast Cancer 569 & 669 [55,56] data pertain to the medical field, Glass Identification [57] data representing forensic science, Credit Approval [58] represents economic data, Iris [59] is a botanical data, and Wine Recognition [60] is a data for chemical composition analysis. The experiments performed with these datasets were akin to the investigations done with Toy-data (in Section 3), and UCIHAR data (in Section 4.2), i.e., K-means algorithm used as concept identifier, where ‘K’ is the number of class labels of each dataset, the hierarchy is set to have a depth of two layers (one Input and one Abstract Concept layer). For every dataset, models were generated using thirty iterations in nine Research Designs (RD) (refer the Table A3 in Section A.2). In every RD 10-Fold cross-validation was applied to determine the performance of the models. An aggregate of Precision, Recall, F1-Score, and Accuracy of all folds in all RDs was calculated for all the datasets, as shown in Figure 12a. From the Figure 12a it can be observed that with Mouse Protein data RANs scores 99.99% (ca.) for all evaluation metric, whereas for Iris, Glass Identification, Breast and Wine Recognitions the observations were convincing, i.e., above 89.00% (ca.). In all the folds of nine RDs ROC curves were also plotted for each class label of the eight datasets, the mean AUC for each of the datasets is shown in Figure 12b. The evaluation metrics and ROC-AUC analysis (Figure 12a & 12b respectively) displays the RANs capability in machine learning tasks with different kind of datasets.

The same procedure was applied to obtain average Precision, Recall, F1-Score and Accuracy for all the datasets with five other classifiers (i.e. RBM+, KNN, LR, MLP, and SGD). Table 5 shows the overall comparison. It is worth noting that being dynamic and unsupervised RANs modeling performed quite satisfactorily especially with Mouse Protein data, where it outperformed SGD and RBM+, was found competent with LR, KNN and MLP classifiers. Figure 13 shows four graphs depicting RANs performance with different benchmark data sets. These graphs display an important aspect of RANs modeling and its performance behavior when evaluated to different research design 13. The Precision, Recall, F1-Score, and Accuracy trajectories of Human Activity Recognition (HAR), Breast Cancer 669 (BC1), Toy-data (TD) and Mouse Protein (MP) Data is almost straight. The evaluation plots of Glass Identification (GI), Wine Recognition (WR), Mammographic Mass (MM), Breast cancer 569 (BC2) and Mouse Protein (MP) datasets show a minimal decline in observations w.r.t RD-1 and RD-9 Research Design. On the contrary, results from IRIS Data (ID) and Credit Approval (CA) dataset depicted a higher value while comparing the evaluation of RD-1 with RD-9 Research Designs of these data

### Table 5. RANs comparison with eight datasets belonging to different domains

| Data                      | Algo     | Precision (%) | Recall (%) | F1-Score (%) | Accuracy (%) |
|---------------------------|----------|---------------|------------|--------------|--------------|
| Breast Cancer             | RBM+     | 99.36 ± 0.09  | 99.31 ± 0.27 | 99.46 ± 2.30 | 99.31 ± 2.77  |
|                          | KNN      | 99.80 ± 0.97  | 99.79 ± 0.66 | 99.85 ± 1.82 | 99.79 ± 1.82  |
|                          | LR       | 99.80 ± 0.97  | 99.80 ± 0.97 | 99.80 ± 0.97 | 99.80 ± 0.97  |
|                          | MLP      | 99.80 ± 0.97  | 99.80 ± 0.97 | 99.80 ± 0.97 | 99.80 ± 0.97  |
|                          | SGD      | 99.80 ± 0.97  | 99.80 ± 0.97 | 99.80 ± 0.97 | 99.80 ± 0.97  |
|                          | RBM+     | 99.36 ± 0.09  | 99.31 ± 0.27 | 99.46 ± 2.30 | 99.31 ± 2.77  |
|                          | KNN      | 99.80 ± 0.97  | 99.79 ± 0.66 | 99.85 ± 1.82 | 99.79 ± 1.82  |
|                          | LR       | 99.80 ± 0.97  | 99.80 ± 0.97 | 99.80 ± 0.97 | 99.80 ± 0.97  |
|                          | MLP      | 99.80 ± 0.97  | 99.80 ± 0.97 | 99.80 ± 0.97 | 99.80 ± 0.97  |
|                          | SGD      | 99.80 ± 0.97  | 99.80 ± 0.97 | 99.80 ± 0.97 | 99.80 ± 0.97  |
|                          | KNN      | 99.80 ± 0.97  | 99.80 ± 0.97 | 99.80 ± 0.97 | 99.80 ± 0.97  |
|                          | LR       | 99.80 ± 0.97  | 99.80 ± 0.97 | 99.80 ± 0.97 | 99.80 ± 0.97  |
|                          | MLP      | 99.80 ± 0.97  | 99.80 ± 0.97 | 99.80 ± 0.97 | 99.80 ± 0.97  |
|                          | SGD      | 99.80 ± 0.97  | 99.80 ± 0.97 | 99.80 ± 0.97 | 99.80 ± 0.97  |

RAN- Regulated Activation Network; SGD- Stochastic Gradient Descent

---

**Preprints** (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 January 2020

377

360

388

387

385

384

380

379

372

367

366

365

364

361

359

369

368

367

366

365

364

363

362

361

360

359

368

367

366

365

364

363

362

361

360

359

368

367

366

365

364

363

362

361

360

359
Table 6. Feature based comparative study of RANs with 5 modeling techniques

| Features | Models | RBM | K-NN | LR | MLP | RANs | SGD |
|----------|--------|-----|------|----|-----|------|-----|
| Graph-Based | Yes | No | No | Yes | Yes | No |
| Dynamic Topology | No | No | No | No | Yes | No |
| Dimension Reduction | Yes | Yes | No | Yes | No |
| Dimension Expansion | May be | No | No | May be | Yes | No |
| Unisupervised | Yes | No | No | No | Yes | No |
| Supports Classification | Yes | Yes | Yes | Yes | Yes | Yes |
| Bio-inspired | Yes | No | No | Yes | No |

sets. Principally, the results of all four metrics of evaluation obtained similar results (with marginal variation) irrespective of the Test and Train data ratio. This is a notable observation because it shows that RANs approach obtains a satisfactory result even when trained with a small amount of data.

Besides classification comparison, the RAN’s modeling is compared with the 5 classifiers based upon 7 features: (1) Whether the modeling in graph based; (2) whether the modeling has a dynamic topology; (3) and (4) whether modeling can reduce or expand the dimension of the data; (5) whether modeling can perform classification; and (7) whether modeling is biologically inspired or not. Tabel 6 details this comparative study. It can be observed from this table that RAN is closely related to the models that are biologically inspired i.e. RBM and MLP.

6. Conclusions and Future work

To comprehend and reasoning for emotions, ideas, etc., it is evident to understand Abstract Concepts because they are perceived differently from Concrete Concepts. There have been notable efforts to study Concrete Concepts (features like walking or ingredients), but progress in investigating Abstract Concepts (generic features such as is-moving or recipe) is relatively less. This article proposes an unsupervised computational modeling approach, named Regulated Activation Networks (RANs), that has an evolving topology and learns a representation of Abstract Concepts. The RAN’s methodology was exemplified through a UCI’s IRIS dataset, yielding a satisfactory performance evaluation of 95% (ca.) for Precision, Recall, F1-Score and Accuracy metrics, along with an average AUC of 99% (ca.) for all the three classes in the dataset. These evaluation result not only showed the classification capability of RANs but also proved the hypothesis of the experiment i.e. the 3 newly created nodes in the Layer-1 symbolically represent the 3 classes of IRIS data as abstract concepts.

Another experiment with IRIS data displayed the characteristic of RAN’s deep hierarchy generation and independence in choosing Concept Identifier. With the aid of Concept Hierarchy Creation algorithm (proposed in Section 4.1), evolving nature of RAN’s modeling is shown using Affinity Propagation clustering algorithm (as an alternate Concept Identifier instead of the K-means algorithm as used in modeling with Toy-data problem). With the generated model it was shown that the model dynamically grew to a depth of six layers and performed with Precision of 94.44% (ca.), Recall of 93.33% (ca.), F1-Score of 93.26% (ca.) and Accuracy of 93.33% (ca.), along with an observed AUC of 100% (ca.), 92% (ca.) and 94% (ca.) for the three classes of data. This experiment also highlights the application of RANs modeling in data dimension transformation and data visualization.

Modeling with UCI’s IoT based Home Activity Recognition (UCIHAR) smartphone sensor dataset exhibited the RAN’s behavior of natural identification of generic Concepts. The experiment hypothesize that six data labels (activity of Walking, Walking_upstairs, Walking_downstairs, Sitting, Standing and Laying) of the dataset are to be identified as Mobile (Walking, Walking_upstairs and Walking_downstairs) and Immobile (Sitting, Standing and Laying) Abstract Concepts. This hypothesis was also proven using classification operation, where, the evaluation of the model shown a performance of 99.85% (ca.) for all four metrics and AUC of 99.9% (ca.) for both Abstract Concepts. The experiment also demonstrates how RAN can be used to model the data from IoT domain in an unsupervised manner.
Figure 13. RANs evaluation metric (Precision, Recall, F1-Score and Accuracy) value behavior w.r.t. varying test and train data ratio over ten datasets [Mice Protein (MP), Breast Cancer 669 (BC1), Breast Cancer 569 (BC2), Credit Approval (CA), IRIS data (ID), Mamographic Mass (MM), Human Activity Recognition (HAR), Toy-data(TD), Wine Recognition (WR) and Glass Identification (GI)].
Table 7. Acronyms used in the Article

| Acronym | Description | Acronym | Description | Acronym | Description |
|---------|-------------|---------|-------------|---------|-------------|
| ACL     | Abstract Concept Labeling | DoC     | Degree of Confidence | MM      | Mammography Mass Dataset |
| AUC     | Area Under Curve | GDF     | Geometric Distance Function | MP      | Mice Protein Dataset |
| BC1     | Breast Cancer 669 Dataset | GI      | Glass Identification Dataset | MRI     | Magnetic Resonance Imaging |
| BC2     | Breast Cancer 569 Dataset | HAR     | Human Activity Recognition Data | RANs    | Regulated Activation Networks |
| CA      | Credit Approval Dataset | ID      | IRIS Dataset | RBM     | Restricted Boltzmann Machine |
| CC      | Concept Creation | ILL     | Inter Layer Learning | RD      | Research Design |
| CHC     | Concept Hierarchy Creation | ILW     | Inter Layer Weights | ROC     | Receiver Operating Characteristic |
| CI      | Concept Identification | K-NN    | K Nearest Neighbor | SGD     | Stochastic Gradient Descent |
| CLS     | Current Layer Size | LR      | Logistic Regression | STF     | Similarity Translation Function |
| CRDP    | Cluster Representative Data Point | MLP     | Multilayer Perceptron | UAP     | Upward Activation Propagation |

The proof of concept of RAN’s modeling as a Machine Learning classifier was also provided with eight UCI benchmarks. It was identified that RAN’s approach performed satisfactorily displaying the best outcome of 98.9% (ca.) with Mice Protein dataset (for all metrics). The comparison of RAN’s modeling with five classifiers substantiated the effectiveness of the proposed methodology. We also observed that the RAN’s performance remained similar irrespective of the size of train data. RAN was also compared with the 5 classifiers based upon its features and it was observed that RAN was similar to bio-inspired models. During the simulations, a non-convexity was observed in several datasets. As future work, we intend to improve RAN’s modeling that can capture the non-convexity in the data and enhance the modeling to build non-convex abstract concepts.

**Funding:** “The work presented in this paper was partially carried out in the scope of the SOCIALITE Project (PTDC/EEI-SCR/2072/2014), co-financed by COMPETE 2020, Portugal 2020 - Operational Program for Competitiveness and Internationalization (POCI), European Union’s ERDF (European Regional Development Fund), and the Portuguese Foundation for Science and Technology (FCT).

**Conflicts of Interest:** “The authors declare no conflict of interest.”

**Abbreviations**

The abbreviations are used in this manuscript are listed in Table 7:

**Appendix A Supplementary Materials**

**Appendix A.1 Data & Scripts**

This section provides links to download the data and python script used to perform RANs modeling experiments, mentioned in this article. The data and script folders can be downloaded from the web URL mentioned in Table A1. The data folder contains many files and the direct path to the files are provided in the Table A1. Similarly, the script folder RAN_V2.0 also contains many folders where Folder RAN consist of the python scripts. The folder Observations is for storing the outcome of the experiments, at the beginning of each experiment the empty folder in directory empty_passes_for_Experiment_Observations must be copied into the Observation directory. The python script related to RANs modeling is in folder RAN, the description is mentioned in the Table A1.

The implemented RANs modeling tool in python takes input data in a specific format (shown in Table A2). Besides the data, the inputs require a header as the first row stacked over the original data. Each header element, \([H - 1, H - 2, \ldots, H - n]\), is the Maximum value possible for their respective column (feature, or dimension). It is assumed that the minimum value of the column is zero, if it is not then the data must be transformed between zero and the maximum positive value as described in Section 3.
Table A1. Data and Python Script of RANs modeling

| Type                  | Description                                      | File-path                                                                 |
|-----------------------|--------------------------------------------------|---------------------------------------------------------------------------|
| Download link         | Data                                              | https://www.dropbox.com/sh/3410ozeru300qpm/AAA2kaA,tUS1r+xHkp%ky9ka/drl=0  |
| Iris Data             | data/iris_with_label.csv                          |                                                                           |
| Mice Protein data     | data/Data_cortex_Nuclear/mice_with_class_label.csv|                                                                           |
| Glass Identification data | data/newDataToExplore/new/GlassIdentificationDatabase/RANsform.csv       |                                                                           |
| Wine Recognition data | data/newDataToExplore/new/WineRecognitionData/RANsform.csv                 |                                                                           |
| Breast cancer 669 data | data/newDataToExplore/new/BreastCancerDatabases/669RansForm.csv            |                                                                           |
| Breast Cancer 559 data | data/newDataToExplore/new/BreastCancerDatabases/559RansForm.csv            |                                                                           |
| UCI HAR data          | data/UCI_HAR_Dataset.csv                          |                                                                           |
| Mamographic Mass data | data/newDataToExplore/new/MamographicMassData/RansForm1                     |                                                                           |
| Credit Approval data  | data/newDataToExplore/new/CreditApproval/RansForm.csv                          |                                                                           |
| Toy data data         | data/toydata5clustersRAN.csv                      |                                                                           |
| RANs classes and methods | RAN_V2-0/RAN/RAN_kfold.py                           |                                                                           |
| Methods               | RAN_V2-0/RAN/Layer.py                              |                                                                           |
| Utilities like Labeling and plotting | RAN_V2-0/RAN/UtilsRAN.py                          |                                                                           |
| Python Script for using RANs | RAN_V2-0/RAN/RAN_input_T1.py                     |                                                                           |

Table A2. Input Data Format for implemented RANs Modeling

| Header  | H-1 | H-2 | .......... | H-n |
|---------|-----|-----|----------|-----|
| Data    | Instances | D-1 | D-2 | .......... | D-n |
|         |         | D-1 | D-2 | .......... | D-n |
|         |         | .   | .   | .......... | .   |
|         |         | .   | .   | .......... | .   |
|         |         | D-1 | D-2 | .......... | D-n |

Appendix A.2 Model Configurations and Research Design

Various experiments, reported in this article, were conducted with several datasets, using six modeling techniques including the proposed methodology i.e. RANs modeling. Table A4 in Section A.2 shows configurations of all the models for all the experiments. The experiments were carried out using python programing language, and implementations of Restricted Boltzmann Machine pipelined with Logistic Regression (RBM+), Logistic Regression (LR), K-Nearest Neighbor (K-NN), Multilayer Perceptron (MLP), and Stochastic Gradient Descent (SGD) models of Scikit-learn library [? ]. It is to be noted that experiments with RBM were carried out, pipelined with the LR algorithm using the default configuration of its implementation in scikit-learn library. The Table A3 lists the nine Research Designs (RD) used in the experiments of this article. In every RD the ratio of the Train and Test data is varied to capture the ability of the classifier being inspected. The Table 7 lists the acronyms used in this article.

Appendix A.3 Multi-class ROC analysis with RANs Modeling

This study is carried out by two processes, first the input true-labels are transformed into a separate vector of binary labels, individually for all Abstract nodes (i.e. 1 for class c1, 0 for all other classes), second, calculating the confidence score for each instance of the input data (or test-data). Both processes are described as follows:

1. Node-wise binary transformation of True-Labels: For example, suppose there are three classes (c1, c2, c3) represented by three abstract nodes (n1, n2, and n3) in RANs model at Layer-1, and

Table A3. Train & Test data distributions in nine Research Designs (RD)

| RD-1 | RD-2 | RD-3 | RD-4 | RD-5 |
|------|------|------|------|------|
| Train | Test | Test | Train | Test | Train | Test |
| 90%  | 10%  | 80%  | 20%  | 70%  | 30%  | 60%  | 40%  | 50%  | 50%  |
| RD-1 | RD-7 | RD-8 | RD-9 |
| Train | Test | Test | Train | Test | Train | Test |
| 40%  | 60%  | 30%  | 70%  | 20%  | 80%  | 10%  | 90%  | 90%  | 90%  |
Table A4. Dataset specific configuration details of models

| Data          | Algo  | Configurations                  | Data          | Algo  | Configurations                  |
|---------------|-------|---------------------------------|---------------|-------|---------------------------------|
| Toy-data      | RBM + LR | Lr=0.00001, iter=500, comp=20 max_iter=30, C=70 | Breast Cancer 669 | RBM + LR | Lr=0.006, iter=100, comp=10 max_iter=30, C=1 |
|               | K-NN  | n_neighbors=30                  |               | K-NN  | n_neighbors=30                  |
|               | LR    | max_iter=10, C=1                |               | LR    | max_iter=10, C=1                |
|               | MLP   | Rs=1, hls=10, iter=250          |               | MLP   | Rs=1, hls=10, iter=250          |
|               | RANs  | CLS=5, Desired_depth=1          |               | RANs  | CLS=2, Desired_depth=1          |
|               | SGD   | alpha=0.0001, n_iter=5, epsilon=0.25 |               | SGD   | alpha=0.0001, n_iter=5, epsilon=0.25 |
|               | RBM + LR | Lr=0.1, iter=500, comp=20 max_iter=30, C=30 |               | RBM + LR | Lr=0.006, iter=100, comp=10 max_iter=30, C=1 |
|               | K-NN  | n_neighbors=15                  |               | K-NN  | n_neighbors=30                  |
|               | LR    | max_iter=4, C=0.00001           |               | LR    | max_iter=10, C=0.001            |
|               | MLP   | Rs=1, hls=10, iter=300          |               | MLP   | Rs=1, hls=10, iter=200          |
|               | RANs  | CLS=8, Desired_depth=1          |               | RANs  | CLS=2, Desired_depth=1          |
|               | SGD   | alpha=0.1, n_iter=10, epsilon=0.25 |               | SGD   | alpha=0.0001, n_iter=5, epsilon=0.25 |
| Mouse Protein | RBM + LR | Lr=0.001, iter=100, comp=10 max_iter=30, C=1 | Mammographic  | RBM + LR | Lr=0.006, iter=100, comp=10 max_iter=30, C=1 |
|               | K-NN  | n_neighbors=10                  |               | K-NN  | n_neighbors=30                  |
|               | LR    | max_iter=10, C=0.001            |               | LR    | max_iter=10, C=0.001            |
|               | MLP   | Rs=1, hls=10, iter=200          |               | MLP   | Rs=1, hls=10, iter=200          |
|               | RANs  | CLS=2, Desired_depth=1          |               | RANs  | CLS=2, Desired_depth=1          |
|               | SGD   | alpha=0.0001, n_iter=5, epsilon=0.25 |               | SGD   | alpha=0.0001, n_iter=5, epsilon=0.25 |
| Breast Cancer 669 | RBM + LR | Lr=0.001, iter=100, comp=10 max_iter=30, C=5 | Mamographic  | RBM + LR | Lr=0.006, iter=100, comp=10 max_iter=30, C=5 |
|               | K-NN  | n_neighbors=15                  |               | K-NN  | n_neighbors=30                  |
|               | LR    | max_iter=5, C=0.00001           |               | LR    | max_iter=5, C=1                 |
|               | MLP   | Rs=1, hls=10, iter=200          |               | MLP   | Rs=1, hls=10, iter=250          |
|               | RANs  | CLS=2, Desired_depth=1          |               | RANs  | CLS=2, Desired_depth=1          |
|               | SGD   | alpha=0.0001, n_iter=5, epsilon=0.25 |               | SGD   | alpha=0.0001, n_iter=5, epsilon=0.25 |
| Glass Identification | RBM + LR | Lr=0.01, iter=1000, comp=20 max_iter=30, C=5 | Wine Recognition | RBM + LR | Lr=0.01, iter=500, comp=20 max_iter=30, C=50 |
|               | K-NN  | n_neighbors=15                  |               | K-NN  | n_neighbors=15                  |
|               | LR    | max_iter=10, C=1                |               | LR    | max_iter=10, C=0.01             |
|               | MLP   | Rs=1, hls=10, iter=400          |               | MLP   | Rs=1, hls=10, iter=300          |
|               | RANs  | CLS=3, Desired_depth=1          |               | RANs  | CLS=3, Desired_depth=1          |
|               | SGD   | alpha=0.01, n_iter=10, epsilon=0.25 |               | SGD   | alpha=0.01, n_iter=10, epsilon=0.25 |

Lr-Learning Rate; iter-Iterations; comp-Number of Hidden Components of RBM; RS-Random State
hls=Hidden Layer Sizes; CLS-Number of clusters at the input layer of RANs

let true-label be \([c_1, c_2, c_2, c_1, c_2, c_3, c_3]\) for 7 test instances, then for node n1 label will be \([1, 0, 0, 1, 0, 0, 0]\) where 1 represents class c1, and 0 depicts others (i.e. c2, and c3).

2 Node-wise confidence-score calculation: This is calculated by averaging activation-value and confidence-indicator of activation for an input instance at an Abstract node. Activation-value is an individual activation of an activation vector obtained by propagating up the data using UAP mechanism of RANs whereas, confidence-indicator is calculated by min-max normalization operation of activation vector. For example, after UAP operation each node (n1, n2, and n3) receives activation \([0.89, 0.34, 0.11]\) (a vector of activation), and confidence-indicator is \(\min\)-\(\max\) \(\frac{[0.89, 0.34, 0.11]}{[0.89, 0.34, 0.11]} = [1.0, 0.29, 0.0]\). and the confidence-score for nodes n1= \((0.89 + 1.0)/2.0 = 0.95, n2= (0.34 + 0.29)/2.0 = 0.32, and n3= (0.11 + 0.11)/2.0 = 0.05).

Appendix A.4 Abstract Concept Labeling (ACL)

This method is optional and useful when the input data is labeled. With this mechanism, we associate an identifier to every Abstract Concept node \(N_j\). Having generated the RANs model with CI, then through CC, ILL, input data is sorted label-wise, and perform UAP operation. The propagated data is inspected class-wise, and label node \(N_j\) with a class-name for which it got the maximum count of.
the highest activation. For example, suppose input data for class-X has 100 instances, after inspecting
the propagated data, it is observed that node \( N_1 \) received highest activation 74-times, whereas, with
remaining 26 cases other nodes experienced maximum activation, therefore, we recognize node \( N_1 \) as
representative of class-X. True-Labels are identified by mapping each class of the input instance directly
to its respective node representative Observed-Labels are obtained by propagating every test-instance
through UAP operation, inspecting which Abstract node received the highest activation for that
data-unit, and label it with the class represented by that node. True-Labels and Observed-Labels are
used to validate the model’s performance.

References

1. Kiefer, M.; Pulvermüller, F. Conceptual representations in mind and brain: theoretical developments,
current evidence and future directions. Cortex 2012, 48, 805–825.
2. Xiao, P.; Toivonen, H.; Gross, O.; Cardoso, A.; Correia, J.a.; Machado, P.; Martins, P.; Oliveira, H.G.; Sharma,
R.; Pinto, A.M.; Díaz, A.; Francisco, V.; Gervás, P.; Hervás, R.; León, C.; Forth, J.; Purver, M.; Wiggins, G.A.;
Miljković, D.; Podpečan, V.; Poliak, S.; Kralj, J.; Žnidaršič, M.; Bohanec, M.; Lavrač, N.; Urbančič, T.; Velde,
F.V.D.; Batterby, S. Conceptual Representations for Computational Concept Creation. ACM Computing
Survey 2019, 52, 1–33.
3. Anderson, J.R.; Matessa, M.; Lebiere, C. ACT-R: A theory of higher level cognition and its relation to visual
attention. Human-Computer Interaction 1997, 12, 439–462.
4. Gärdenfors, P. Conceptual spaces: The geometry of thought; MIT press, 2004.
5. Sun, R.; Peterson, T. Learning in reactive sequential decision tasks: The CLARION model. IEEE
International Conference on Neural Networks. IEEE, 1996, Vol. 2, pp. 1073–1078.
6. Rumelhart, D.E.; Hinton, G.E.; Williams, R.J. Learning internal representations by error propagation.
Technical report, California Univ San Diego La Jolla Inst for Cognitive Science, 1985.
7. Freedman, D.A. Statistical models: theory and practice; Cambridge University Press, 2009; chapter 7.
8. Altman, N.S. An introduction to kernel and nearest-neighbor nonparametric regression. The American
Statistician 1992, 46, 175–185.
9. Zhang, T. Solving large scale linear prediction problems using stochastic gradient descent algorithms.
Proceedings of the twenty-first international conference on Machine learning. ACM, 2004, p. 116.
10. Hinton, G. A practical guide to training Restricted Boltzmann Machines. Momentum 2010, 9, 926.
11. Binder, J.R.; Westbury, C.F.; McKiernan, K.A.; Possing, E.T.; Medler, D.A. Distinct brain systems for
processing concrete and abstract concepts. Journal of Cognitive Neuroscience 2005, 17, 905–917.
12. Gao, C.; Baucom, L.B.; Kim, J.; Wang, J.; Wedell, D.H.; Shinkareva, S.V. Distinguishing Abstract from
Concrete concepts in Supramodal Brain Regions. Neuropsychologia 2019, 131, 102 – 110.
13. Kousa, S.T.; Vigliocco, G.; Vinson, D.P.; Andrews, M.; Del Campo, E. The representation of abstract words:
Why emotion matters. Journal of Experimental Psychology: General 2011, 140, 14.
14. Barsalou, L.W.; Wiemer-Hastings, K. Situating abstract concepts. Grounding cognition: The role of perception
and action in memory, language, and thought 2005, pp. 129–163.
15. Lühr, G. Embodied cognition and abstract concepts: Do concept empiricists leave anything out?
Philosophical Psychology 2019, 32, 161–185.
16. Gibbs Jr, R.W. Why many concepts are metaphorical. Cognition 1996, 61, 309–319.
17. Hill, F.; Korhonen, A. Learning Abstract Concept Embeddings from Multi-Modal Data: Since You Probably
Can’t See What I Mean. Empirical Methods in Natural Language Processing, 2014, pp. 255–265.
18. Iosif, E.; Potamianos, A.; Giannoudaki, M.; Zervanou, K. Semantic similarity computation for abstract and
concrete nouns using network-based distributional semantic models. Proceedings of the 10th International
Conference on Computational Semantics. Potsdam, Germany-[sn], 2013, pp. 328–334.
19. Iosif, E. Network-based distributional semantic models. PhD thesis, Technical University of Crete, Chania,
Greece, 2013.
20. Maniezzo, V. Genetic evolution of the topology and weight distribution of neural networks. IEEE
Transactions on Neural Networks 1994, 5, 39–53.
21. Stanley, K.O.; Miikkulainen, R. Evolving neural networks through augmenting topologies. Evolutionary
computation 2002, 10, 99–127.
22. Miikkulainen, R.; Liang, J.; Meyerson, E.; Rawal, A.; Fink, D.; Francon, O.; Raju, B.; Shahrzad, H.; Navruzyan, A.; Duffy, N.; others. Evolving deep neural networks. In Artificial Intelligence in the Age of Neural Networks and Brain Computing; Elsevier, 2019; pp. 293–312.

23. Hintze, A.; Edlund, J.A.; Olson, R.S.; Knoester, D.B.; Schossau, J.; Albantakis, L.; Tehrani-Saleh, A.; Kvam, P.; Sheneman, L.; Goldsby, H.; others. Markov Brains: A technical introduction. arXiv preprint arXiv:1709.05601 2017.

24. Pinto, A.M.; Barroso, L. Principles of Regulated Activation Networks. In Graph-Based Representation and Reasoning; Springer, 2014; pp. 231–244.

25. Jacoby, L. Perceptual enhancement: persistent effects of an experience. Journal of Experimental Psychology. Learning, Memory and Cognition 1983, 9, 21–38.

26. Roediger, H.; Blaxton, T. Effects of varying modality, surface features, and retention interval on priming in word-fragment completion. Memory & Cognition 1987, 15, 379–388.

27. Roediger, H.; Mcdermott, K. Creating false memories: Remembering words not presented in lists. Journal of Experimental Psychology. Learning, Memory, and Cognition 1995, 21, 803–814.

28. Gärdenfors, P. Conceptual spaces as a framework for knowledge representation. Mind and Matter 2004, 2, 9–27.

29. Sivik, L.; Taft, C. Color naming: A mapping in the IMCS of common color terms. Scandinavian Journal of Psychology 1994, 35, 144–164.

30. Rosch, E. Cognitive representations of semantic categories. Journal of Experimental Psychology: General 1975, 104, 192.

31. Mervis, C.B.; Rosch, E. Categorization of natural objects. Annual Review of Psychology 1981, 32, 89–115.

32. Rosch, E. Prototype classification and logical classification: The two systems. New trends in conceptual representation: Challenges to Piaget’s theory 1983, pp. 73–86.

33. Parsons, E. Evaluating subspace clustering algorithms. in Workshop on Clustering High Dimensional Data and its Applications, SIAM International Conference on Data Mining, 2004, pp. 48–56.

34. Livingstone, D. Unsupervised Learning; John Wiley & Sons, Ltd, 2009; pp. 119–144.

35. Yoon, J.; Raghavan, V.; Chakilam, V. BitCube: clustering and statistical analysis for XML documents. Journal of Intelligent Information Systems 2001.

36. Van Deursen, A.; Kuipers, T. Identifying Objects Using Cluster and Concept Analysis. Proceedings of the 21st International Conference on Software Engineering. ACM, 1999, pp. 246–255.

37. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. Nature 2015, 521, 436–444.

38. Schmidhuber, J. Deep Learning in Neural Networks: An Overview. Neural Networks 2015, 61, 85–117. Published online 2014; based on TR arXiv:1404.7828 [cs.NE].

39. Bengio, Y. Learning Deep Architectures for AI. Foundations and Trends® in Machine Learning 2009, 2, 1–127.

40. Eigen, D.; Rolfe, J.; Fergus, R.; LeCun, Y. Understanding Deep Architectures using a Recursive Convolutional Network. International Conference on Learning Representations. CBLS, 2014.

41. Bengio, Y.; Courville, A.; Vincent, P. Representation Learning: A Review and New Perspectives. IEEE Transactions on Pattern Analysis and Machine Intelligence 2013, 35, 1798–1828.

42. Anderson, J. A spreading activation theory of memory. Journal of Verbal Learning and Verbal Behavior 1983, 22, 261–295.

43. Collins, A.; Quillian, M. Retrieval time from semantic memory. Journal of Verbal Learning and Verbal Behavior 1969, 8, 240–247.

44. Crestani, F. Application of Spreading Activation Techniques in Information Retrieval. Artificial Intelligence Review 1997, 11, 453–482.

45. McNamara, T.; Altarriba, J. Depth of spreading activation revisited: Semantic mediated priming occurs in lexical decisions. Journal of Memory and Language 1988, 27, 545 – 559.

46. Roediger, H.; Balota, D.; Watson, J. Spreading activation and arousal of false memories. The nature of remembering: Essays in honor of Robert G. Crowder 2001, pp. 95–115.

47. Kavukcuoglu, K.; Serenat, P.; Boureau, Y.; Gregor, K.; Mathieu, M.; LeCun, Y. Learning Convolutional Feature Hierarchies for Visual Recognition. Advances in Neural Information Processing Systems (NIPS 2010), 2010, Vol. 23.
48. Sermanet, P.; Eigen, D.; Zhang, X.; Mathieu, M.; Fergus, R.; LeCun, Y. OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks. International Conference on Learning Representations. CBLS, 2014.

49. Hartigan, J.A.; Wong, M.A. Algorithm AS 136: A K-means Clustering Algorithm. Journal of the Royal Statistical Society. Series C (Applied Statistics) 1979, 28, 100–108.

50. Frey, B.J.; Dueck, D. Clustering by Passing Messages Between Data Points. Science 2007, 315, 972–976.

51. Lichman, M. UCI Machine Learning Repository, 2013.

52. Anguita, D.; Ghio, A.; Oneto, L.; Parra, X.; Reyes-Ortiz, J.L. A Public Domain Dataset for Human Activity Recognition using Smartphones. Empirical Methods in Natural Language Processing, 2013.

53. Higuera, C.; Gardiner, K.J.; Cios, K.J. Self-organizing feature maps identify proteins critical to learning in a mouse model of down syndrome. PLOS ONE 2015, 10, e0129126.

54. Elter, M.; Schulz-Wendtland, R.; Wittenberg, T. The prediction of breast cancer biopsy outcomes using two CAD approaches that both emphasize an intelligible decision process. Medical Physics 2007, 34, 4164–4172.

55. Street, W.N.; Wolberg, W.H.; Mangasarian, O.L. Nuclear feature extraction for breast tumor diagnosis. Biomedical Image Processing and Biomedical Visualization. International Society for Optics and Photonics, 1993, Vol. 1905, pp. 861–871.

56. Bennett, K.P.; Mangasarian, O.L. Robust linear programming discrimination of two linearly inseparable sets. Optimization methods and software 1992, 1, 23–34.

57. Evett, I.W.; Spiehler, E.J. Rule induction in forensic science. Technical report, Central Research Establishment, Home Office Forensic Science Service, 1987.

58. Quinlan, J.R. Simplifying decision trees. International Journal of Human-Computer Studies 1999, 51, 497–510.

59. Fisher, R.A. The use of multiple measurements in taxonomic problems. Annals of Human Genetics 1936, 7, 179–188.

60. Forina, M.; Leardi, R.; Armanino, C.; Lanteri, S.; Conti, P.; Princi, P. PARVUS: An extendable package of programs for data exploration, classification and correlation. Journal of Chemometrics 1988, 4, 191–193.