Chapter 2: Literature Review

Nate Bachmeier

DIS-9902: Dissertation Proposal

April 24, 2022

Northcentral University

# Chapter 2: Literature Review

The problem to be addressed in this study is implementing a quality assurance process for an autonomous assistant for elderly and special needs care. This research theorizes that using computer vision (CV) can provide a consistent experience across a diverse global audience. Building autonomous assistants is challenging due to requiring multiple domain specializations like computer networking, embedded technologies, AI/ML, and distributed computing (Tun, Madanian, & Mirza, 2021). Beyond technical constraints, potential privacy and safety from video monitoring create barriers to locating volunteer patients. Furthermore, those difficulties limit other researchers from reproducing the results. These factors slow down innovation and restrict the value researchers can contribute to the body of knowledge.

This constructive research design study aims to propose a research process that divorces privacy and safety concerns from investigating autonomous assistants in elderly and special needs care. It aims to deliver this capability by utilizing humanoid constructs within a realistic physics simulation process. Next, positioning virtual cameras, instruments, and devices within the virtual world enables researchers to collect their experimentation data. Lastly, the automation can modify the environment using programmable interfaces such as raising the alarm or applying other mitigations.

## Chapter overview

*Provide an overview of the subheadings in the literature that will discuss. This section should be around a 1-page and help frame what the reader will learn from this 50-page document. Otherwise, it risks being very confusing and not meeting their expectations.*

Bringing together evidence across studies first requires a search of the literature; this is one of the early steps in what is sometimes called a “systematic review.” With a set of studies in hand, the next step in the review may (or may not) be a statistical approach to analyzing the published findings, usually called “metaanalysis” or a variant thereof. Different disciplines emphasize different aspects of this process, and afford the process different levels of scholarly prominence (Ozier, 2021).

## Search Method and Resources

This literature review used the Northcentral University Library (NCUL) to identify relevant peer-reviewed articles and books published from 2019 to 2022. It also includes foundational papers for historical context and generally accepted process standards outside this period. Students use NCUL’s Roadrunner search to aggregate results from industry-standard sources like the Institute of Electrical and Electronics Engineers (IEEE), Association for Computer Machinery (ACM), Springer Publishing, John Wiley & Sons ProQuest, among others.

A breath-first search scanned for surveys, challenges, and opportunities on the constructive research project’s core concepts (see Table 1). The breath-first search uncovered several themes that drove depth-first investigations. For instance, researchers are approaching hyper-scale ML training with custom hardware acceleration and continuous learning-at-the-edge methods (Plus Company Updates, 2021; Prapas et al., 2021). In other cases, themes like *Using Convolutional-Graph Neural Networks (C-GNN) for HAR* necessitate a sequential breadth-first search to contextualize supporting concepts. This search process continued until finding fifty unique documents. Next, bibliographical reviews for each document extracted themes. Those sorted themes are available in the proceeding conceptual frame section, which attempts to present each topic’s current state and direction from Table 1.

Table 1: Survey search terms

|  |  |
| --- | --- |
| Concept | Example search queries |
| Elderly and special needs industry state | * (elderly care or special needs) and industry * (global or internal) and (disabled or medical) |
| Computer vision (CV) | * computer vision or CV * computer vision and (surveys or opportunities) |
| Human Activity Recognition (HAR) | * (human activity recognition or HAR) and (computer vision or CV) * HAR (state-of-the-art or challenges) |
| Machine Learning (ML) Training | * (ML or machine learning) training and scale * distributed ML training |
| Physics simulation | * (Unity or ROS or robotic operating system) and (process or environment) simulation * (dynamic or synthetic or virtual) environment testing |

## Conceptual Framework

A conceptual framework is a blueprint that communicates a natural progression of the phenomenon to be studied (Dickson, Emad, & Adu-Agyum, 2018). It is essential for quality research as it outlines a methodical structure of definitions, concepts, and relationships.

There are four approaches to studying a business use case or phenomena (see Table 2). This study’s blueprint derives from a constructive design science research (DSR) methodology.

DSR is one of the most common research methods for information systems and technology (Silvestrini & Sammito, 2012). These studies identify a problem, build artifacts, and communicate the implementation’s unique value (Hevner et al., 2004). In addition, many researchers follow this process to build proof-of-concept and execute case studies. This methodology is appropriate for examining elderly and special needs care solutions. After creating the system, it can support a targeted case study that measures its ability to deliver value.

Table 2: Example Research Strategies for Classifying Movement in Video

|  |  |  |
| --- | --- | --- |
| Approach | Description | Study Example |
| Quantitative | Studies the magnitude of a phenomena | Measure the resources necessary to classify movement with embedded systems |
| Qualitative | Explores a concept without a numerical basis | Exploration of reasons movement classification fails |
| Mixed-Method | Combines exploration and studying the magnitude of these issues | What preparation steps reduce the costs of movement classification |
| Constructive | Produce artifacts to study a scenario | Create an algorithm for classifying movements |

### Fundamental Approach

Constructive research practitioners gravitate toward either Design Science Research (DSR) or the Constructive Research Approach (CRA). One of the critical differences between them is that DSR relies more heavily on existing theories, versus CRA does not explicitly require a base theory (Piirainen & Gonzalez, 2013). More recently, Iivari (2020) criticizes the debate stating that constructive research must first and foremost produce high-quality artifacts. She advocates for “less theory, but better design theory (pg. 504),” especially within rapidly evolving industries like Information Technology and Communication. Zeller (2014) would agree with this position, adding success criteria that the artifacts are “challenging, elegant and useful.” This research project aligns with these requirements by focusing on connecting artifacts with patient needs and challenges.

### Central concepts and relationships

Here, the study presumes that CV and HAR can improve the livelihood of elderly and special needs patients. However, implementing those custom models is prohibitively expensive, and any research conclusions will be challenging to reproduce. This research project aims to mitigate these challenges by demonstrating CV and HAR methodology with simulated humanoids. It is beyond the scope of this dissertation to prove those methods are superior to existing and more laborious strategies.

There are multiple core concepts necessary to delivering this outcome. First, a literature review must examine the challenges and opportunities for elderly and special needs care. Quality research starts from customer challenges and works backward to find technological solutions (Bryar & Carr, 2021). In contrast, technology-first methodologies come with a higher risk of not producing valuable outcomes.

Second, the central premise relies on exhibiting CV and HAR methodologies for predicting actions and behaviors. Computer vision (CV) is a process for extracting data from image sources. Next, human activity recognition (HAR) processes must classify that data into distinct actions and behaviors, such as the person is sitting or falling. Those requirements raise several questions within the literature review context. For example, what mechanisms are being built or deprecated? This research study does not aim to create a novel solution and plans to reuse existing methods.

Third, the research topic needs to train the ML model using a simulated environment with humanoid characters and virtual instruments. This situation raises implementation questions such as trade-offs between industry-standard tooling, design patterns, and configuration nuances. The literature review must identify strategies that are likely to produce high-quality results. It is beyond the scope of this dissertation to implement proof-of-concept (POC) solutions for every potential combination.

Fourth, the artifacts must be high-quality and functional within a noisy environment. Meeting these expectations raises questions regarding ML training strategies. For example, do other researchers remove (or add) randomness to their DNN architectures? Are there specific situations that are more applicable for improving model quality? This constructive research study aims to incorporate these recommendations but stops short of directly comparing algorithms or methods.

Fifth, the literature review must uncover strategies for scaling the ML training and inference to production scale. This sub-topic is crucial for bringing ML capabilities to public markets. However, it is also sufficiently complicated to populate multiple separate dissertations. Therefore, this dissertation only discusses literature trends for high-performance hardware, low-power hardware, and edge processing.

### Implementations and alternative framework

The proposed framework establishes capabilities that align them with the business challenge of improving elderly care and special needs. It also uses generalizable virtual camera instruments for CV and HAR experimentation with humanoid agents. Aspects of this framework appear in other publications (Gu et al., 2021; Banjarey et al., 2021). However, utilizing humanoid subjects with HAR and CV is not a mainstream topic in surveyed literature.

An evaluation of alternative conceptual frameworks also took place. First, would an alternative virtual instrument be more appropriate? Instead, this study could predict HAR with accelerometers and gyroscopes (Gu et al., 2021). There are several advantages to wearable sensors, such as they follow the patient within the world. Researchers have also demonstrated applying these sensors to predicting diverse action spaces (Nugroho et al., 2018). However, CV-based agents can extract more context from those same behaviors. For instance, a biosensor might predict that a patient is eating, but not the food type. Instead of directly competing technologies, future solutions must integrate these heterogeneous sources.

Second, an argument might exist that using humanoids is nonsensical and advocate for training the HAR models with public video repositories (e.g., YouTube). This approach has several benefits, such as realistic action depictions and freely available labeled data. However, it might be more challenging to train models on this real-world basis versus the controlled and sterile simulation process. Furthermore, researchers can dynamically scale humanoid properties (e.g., weight) to assess model performance across parameter gradients. Future research could combine the frameworks with the video repositories validating the laboratory environment’s usefulness.

## Challenges and opportunities for care providers

*This section is a placeholder for compiling notes from the Industry state section. It attempts to frame the business environment and limitations that create the need for additional research.*

## What is the role of data mining

The four data-mining categories are association rule mining, clustering, classification, and regression modeling (Barua & Mondal, 2019). Association rules are patterns like if *X then Y,* such as a person buying bread (X) is likely also to purchase butter (Y). Clustering and classification are related strategies that attempt to group similar items into buckets. The critical difference is that classification knows the bucket labels ahead of time (supervised) while clustering does not (unsupervised). For instance, a teacher gives their class a quiz and then maps them into groups by their assessment score (e.g., A, B) is a classification problem. Suppose they mapped the students on their favorite color. In that case, the groups are not deterministic, and it is a clustering scenario. Regression modeling tries to find a mathematical equation that explains the observations. A classic example estimates housing prices by considering square footage, house age, and room count, among other features.

Across these high-level categories, numerous scenario-specific algorithms are available for different data sets. For instance, Apriori-based algorithms rely on the concept that subsets of frequent itemsets must also be frequent itemsets to prune the search space and timely report recommendations (Mejia, Quintero, & Builes, 2017). Another use case comes from Self-Organizing Maps that cluster or categorize arbitrary data for anomaly detection (Sonmez et al., 2018). Then consider Ant Colony Optimization and Genetic Algorithms, which combine random guessing and regression modeling to iterate toward optimal solutions (Mirjaili et al., 2018; Leios, 2017). Other strategies exist to handle countless other challenges like dimension reduction (e.g., Principal Component Analysis) and brute force discovery (e.g., Parameter Sweeping) (Starmer, 2017).

### Organizational examples of data mining

Many financial investment firms rely on different automated strategies to filter the sea of market data into a manageable number of options. For example, Fonskea and Liyange (2008) propose a data mining strategy that tracks related companies' correlation (e.g., FedEx and UPS) and profits from deviations. In this case, both shipping companies are likely to experience similar political and economic headwinds. Bhoopathi and Rama (2017) propose an Apriori-like algorithm that attempts to derive trading signals based on implicit associations between instruments (e.g., X and Y are inversely correlated). Hargreaves and Yi (2012) use a decision tree model to filter the Australian index on fundamental data (e.g., return on equity) from 2000 companies down to a high-quality basket of the top six. Finally, George and Changat (2017) assess the market interdependencies by transforming daily quotes into connected graphs.

### Explain challenges experienced using data mining

There is a joke that ‘70% of all statistics are made-up,’ inferring that the model is unlikely to work in practice without properly evaluating correlation versus causation. Carver (2007) touches on this point with guidance that researchers focus on relevance, not “just seeing what we want to see.” Snee (2015) echos this point that high-quality models are practical and explainable. Both Fonseka & Liyanage’s and George & Changat’s did not account for the contextually sensitive results of the Great Recession occurring in parallel. Bhoopathi and Rama’s association rules discovered tight relationships between Intuit (creator of TurboTax) and International Fragrance—with no economic justification. Aside from Hargreave and Yi, none of these approaches even had a basis in modern market theory. For instance, correlations between price movements did not account for volume. The authors also limited their asset analysis to only primary assets instead of expanding into secondary assets. George & Changat determined that banks were the most critical aspect of their network but did not investigate interest rates, GDP, or consumer credit statistics. Bhoopathi and Rama could have transformed the data with a moving average to smooth out noise, decreasing false-positive rules.

### Enabling Machine Learning

Data mining enables transforming data into information. Researchers can build statistical models that predict outcomes when that information represents evidence for questions. Enabling this symbiosis requires well-defined objectives or machine learning algorithms will lead to inaccurate solutions.

## What exactly is artificial intelligence

Dreams of artificial intelligence can trace back to philosophical debates in ancient Greece. Prometheus would mold handfuls of clay into images of the gods and later gave life. The sprouting of ideas came from mathematics, biology, and computer science before eventually producing modern artificial intelligence. While these different domains have unique perspectives, they collectively land in four categories of intelligent systems (Lukac, Milic, & Nikolic, 2018). The first divide asks if the system *thinks* or *acts*, or more precisely, can reason about the problem. These top-level categories contain subcategories of applications that mimic *humans* versus *rational* actors.

### Description of Technology

There are three high-level categories of artificial intelligence, specifically rules and heuristics, machine learning, and deep understanding (Buchanan, 2005).

1. Before 1962, applications would rely on practical techniques for reducing the trial-and-error search space. This heuristic-centric approach is helpful for chess and other video game engines. Despite criticism for being naïve, many LOB (Line of Business) applications continue to leverage this technique successfully.
2. In 1963, Edward Feigenbaum and Julian Feldman’s *Computers and Thought* centralized many ideas across the computing industry. Their literature and new programming paradigms, such as McCarthy’s LISP, lay the foundation that became machine learning. Researchers use these tools to build statistical models that represent a situation. For instance, if a customer purchases bread, what else could you recommend? Perhaps butter, jam, and deli meat.
3. In 1949, neural scientists found that the human brain transmits signals between a weighted graph of neurons (Lukac, Milic, & Nikolic, 2018). Despite unlocking the biological key to mimicking cognitive learning, the processing power was unavailable until the early 2000s. Researchers use neural networks to extract patterns to nebulous problems that meet or exceed human capacities.

### Purpose and Function

Traditional software follows the model of *data* plus *rules* equals *outcomes.* In contrast, intelligent systems use data and outcomes to derive rules. This distinction can be valuable when the *rules* are fuzzy or not entirely understood. After extracting those rules into a model, researchers and engineering teams can predict actions across mechanical, thinking, and feeling tasks (Huang, Rust, & Maksimovic, 2019).

* Mechanical tasks are actions that are highly repetitive and benefit from automation. These are operations like turning on lights or assembly-line construction.
* Thinking tasks are operations that require analysis and rationalization. For instance, “does this picture contain a hotdog,” or “is this sentence grammatically correct?”
* Feeling tasks, emulate interpersonal experiences, and express empathy toward the users. These autonomous systems might replace a call center or control support chatbots.

### Evolution of the problem

Numerous organizations begin their journey into intelligent systems with statistical modeling and variance analysis. These approaches work for many linear models but break down non-parametric functions (Waal & Toit, 2011). For example, a business wants to appraise houses given a collection of features about the home. Houses come in all shapes and sizes, making it challenging to compare those features directly. Instead, the appraiser must approximate a function that considers these characteristics and their weighted importance. Meanwhile, another company needs to classify handwritten digits, which requires mapping a 32x32 pixel image to its numeric value. Both scenarios and countless more require a mechanism to translate these non-parametric functions into parametric approximations.

### Nature’s solution

In biology, animal brains accomplish these tasks through meshes of neurons that transmit signals across connected synaptic (transforming) and activation (filtering) links (Keller, Liu, & Fogel, 2016). Later, that animal sees an object, and its brain encodes the image into a feature map. These features traverse the brain’s neural pathways and output a collection of responses, such as “the object is food and ten feet away.” Over time, the creature *learns* if those responses are correct and revise network weights to encourage or avoid similar situations. Data scientists and mathematicians replicate these ideas by calibrating edge weights, through backpropagation, on connected graphs called *neural networks*.

## How does computer vision work

Modern CV-based methods emulate primate biology across three distinct subsystems called neural dynamics, embodiment, and awareness (Ballard & Zhang, 2021). Researchers expand on these different subsystems to implement their specific use cases.

### Neural dynamics

Primates use retinotopy to map visual input from their retina to neurons. This process incorporates a random sampling and batching procedure to activate those neurons, with different combinations producing unique classifications. Marr (1982) proposed that machines could emulate this behavior to extract intrinsic images and functional constraints. His research shows that for every point within an image , it is possible to calculate its information level. Then applying a smoothing function can remove the noise and produce object detection masks! Marr’s paradigm remains foundational to modern CV methodologies even forty years later.

### Embodiment

Bajcsy (1988) proposed that vision is an active process, and a hierarchy of decisions must occur. Each layer within the hierarchical map must encode the likelihood of a prediction within the context of the previous layer. For instance, when a person sees a cat, their brain uses different neuron groups to identify edges-to-shapes, shapes-to-labels, and aggregate labels to object names. Today, AI/ML practitioners call this construct the “hidden layers” within neural network architectures.

### Awareness

Around the mid-90s, researchers began exploring the notion of gaze control and fixation (Ballard & Zhang, 2021). They discovered that at least six separate systems stabilize objects within primate vision and implement a sophisticated data inventory system. For instance, when primates search for *a blue ball in the image*, their brains cache metadata to accelerate the gaze. Another critical service called fixation only persists memories that have an associated high reward. For example, people safely drive to work without recognizing the preceding events because the routine operation did not produce new information.

In 1996, Kaelbling et al. proposed encoding these systems as policy maps that activate through an abstract reward function. Their notion of *reinforcement learning* explains how primates program their brain using visual information. Researchers have since formalized this approach into a multi-process model where “reinforcement threads” combine to produce sophisticated composite decisions. Consider the problem of “should I eat this food?” In this situation, parallel threads predict it is a hotdog, hunger level, and availability of mustard. Their aggregate response invokes an appropriate behavior based on the visual information.

## What’s the role of Markov chains

A core challenge to applying basic statistics to real-world data is assuming that each action is independent. However, many scenarios contain a conditional state transition probability dependent on the current state. If the stock market falls 5%, should an investor buy? The binary question requires a contextually sensitive answer that considers their net position (short the market), outlook (2008 financial crisis versus 2017 Trump bump), and similar factors. Markov chains provide the mathematical basis for making statistical models that incorporate these dependencies (Kahn Academy, 2014). Creating the hypothetical purchasing model (see Figure 1) begins with a state diagram representing the different actions available. Then Monte Carlo solutions can approximate each edge’s weight by random sampling and recording the decisions. At the same time, multiple use-cases can follow the same model, the scenario-specific decision weights. For instance, consider the differences between investing in (a) 401k retirement account that only adds index funds versus (b) delta-neutral (directionless) options trader. This trait is similar to other algorithms where efficient training requires relevant facts to specific questions.

Figure : Purchasing Model

Diagram, schematic

Description automatically generated

### Markov Experiment

Many online tutorials recommend exploring Markov chains as a solution to predict the next token in a sequence. Mason (2020) maintains an open-source repository of Shakespeare plays, which is easy to mine for different related sentences. An experiment began with downloading each script and normalizing the text into a corpus of lowercase words. Next, an iterator constructs a word\_dictionary that maps n-gram tuples to a word bag of immediately following values. Then traversal of the Markov model chooses a random starting point, then selects a random next word, iterating until a stop condition. Across the test iterations, tests of different n-gram sizes (degrees of freedom) ranged from one to six. The higher the count, the more natural the sentences sound, primarily due to overfitting. Even at low n-gram terms, a frequent challenge arose from many unique words causing long sequences of static choices.

Figure n-gram Examples

Graphical user interface, text, application

Description automatically generated

### Neural Networks

A Multi-Layer Perceptron (MLP) algorithm aims to map a non-parametric set of inputs to a parametric set of outputs by approximating an intermediary mapping function (the hidden layer). A fully connected graph can represent this structure. All inputs connect to the hidden layer, which connects to all outputs. Next, an iterative process forward-feeds examples through the network. Lastly, backpropagation updates the network weights and performs error corrections concerning the expected value (Ng, 2016). According to Fridman (2017), backpropagation is a recursive process of taking the partial derivative of two logic gates and applying a weighted update. He expands on the idea of these connected graphs with an example of image classification passing through several three layers (extracting edges, corners, object parts, and object identities). While the mathematical basis and engineering steps are somewhat procedural, the efficient design of the network architecture requires both art and science.

Perhaps the artfulness comes from a lack of planning or awareness of how the *ensemble* of distinct training subsystems combines. There is no reason to assume that every node is fully connected or has an edge weight above zero (see Figure 3). A logical representation might consider feature ‘x1’ connected to N neurons that regress one output, with feature ‘x2’ implementing some classification pattern. These network segments produce collaborative signals to provide a more productive inference about the broader topology. These network segment microstructures extrapolate and remain present in more complex architectures. The solutions by both BellKor (2007) and Li et al. (2019) suggest that this assumption is generally accurate.

Figure GANN Architecture (de Waal & du Toit, 2011, p. 399)

Diagram

Description automatically generated

### Neural Network Experiment

Consider the scenario of mapping 28x28 images of clothing to ten categorical labels (e.g., hats versus coats). The number of input features (neurons) is 784, and there will be ten output neurons—how many neurons should exist in the middle? Rosebrock (2019) provides an example solution (see Figure 4) to Fashion MNIST that begins with feature reduction through two max-pooling hidden layers and batch normalization. After cleaning, the solution uses a single 512-neuron hidden layer to predict one of ten output categories (with softmax). Reducing the hidden layer's size to 128 or 256 has minimal impact on the cross-validation scores, though really low values of 5 to 16 negatively impact accuracy. In this specific example, changing the activation functions (e.g., softmax to tan-h) creates more performance fluctuation than any other knob, with model accuracy ranging from 20 to 85%.

Figure : Fashion MNIST

Diagram, schematic

Description automatically generated

### Observations

The first and most critical step in any data mining exercise is determining the question and discovering supporting evidence. Until this action occurs, the business is unlikely to have a successful deliverable and will spend excessive resources investigating irrelevant materials. After clearly articulating the business value, the engineer teams can perform broad filtration of data sources based on their ability to address those questions. During filtration, having a logical framework can improve the search process through partition pruning of the relevant data stores. For instance, if the business operates in Michigan, there is potentially minimal value in exploring Texas-specific data. After coalescing the supporting facts into a central location, cleaning and curation processes must confirm that the data is complete and pristine. Perfect information needs to be both the right size and volume, or it might be incompatible with the analysis algorithms. For example, an instance learning algorithm expects individual records, not aggregate counts.

Markov Chains and Neural Networks are two strategies for making predictions on data through graph-like structures. Unlike basis statistics, Markov removes the need for actions to be independent and instead expresses them as weighted state machines. These state machines can improve accuracy in workflows, such as guessing the next word in a sentence. Neural Networks and related MLP algorithms rely on weighted graphs and backpropagation to make predictions. While there is some artfulness, an alternative perspective asks if these are ensembles of small network segments. Evidence towards this interpretation exists in multiple advanced papers and helps to demystify the “machine learning black box.” It also means that several related concepts, patterns, and practices of data processing networks should also appear within more advanced neural network architectures.

## How are neural networks evolving

Frank Rosenblatt (1958) proposed the Mark I Perception as the first neural network architecture. This construct attempts to explain animals’ biological networks to *perceive* the world around them. The network consists of a collection of weighted sensors that converge into *one* learning circuit. Mechanical devices can replicate that implementation and train the connected weights to emulate animal intelligence with noisy data. Since this seminal paper, researchers have expanded deep neural networks to incorporate hundreds to thousands of connectivity layers. However, the specific configuration remains more art than science, with researchers manually iterating through trial and error experimentation (Ünal & Başçiftçi, 2021). The research field has two eras, with the first being evolution of the artificial neural network (ANN) from 1989 to 2015. Then, the deep neural network (DNN) evolution became the primary focus from 2015 to the present. The industry is moving so fast that information beyond two years is becoming outdated.

### Artificial neural networks era

Perceptron was revolutionary with its weighted signals triggering an activation function. This construct was insufficient for many scenarios and led to Multi-Layer Perceptron, which links a series of activation functions. Semantically, researchers can encode Boolean logic into these gates to derive more sophisticated insights. For instance, a network might contain two gates representing a person’s hunger level and food availability. Distinct signals can activate with each predicate to determine the overall scenario probability. That aggregate threshold can trigger an alarm or notification for the overarching decision to eat the food.

There are numerous activation functions, and a subset of the most common ones are available in Table 3. Originally researchers began with Sigmoid functions, which exponentially become a positive or negative one-value. However, this calculation is complex and slows down model convergence. A simple performance improvement came from using the tanh(x) function, similar to Sigmoid (Meta AI, n.a.). Now, researchers choose Rectified Linear Unit (ReLU) as the most preferred industry-standard algorithm (Ünal & Başçiftçi, 2021). Several scenario-specific variations like Leaky ReLU aim to scale and retain negative values versus truncating them entirely.

Table 3: Activation Functions

|  |  |  |
| --- | --- | --- |
| Activation Function | Formula | Description |
| Sigmoid |  | Mathematical function having an S-shaped curve with asymptotes at -1 and 1 |
| Tanh |  | A hyperbolic function that’s a ratio of sinh and cosh |
| ReLU |  | The most popular activation function |
| Leaky ReLU |  | An enhanced ReLU for incorporating scaled negative values |

### Architecture generalization challenge

Simple networks have poor learning abilities and are challenging to generalize to more sophisticated scenarios. Meanwhile, deep neural networks can learn intricate and subtle patterns but require more data before converging (Ünal & Başçiftçi, 2021). This trade-off causes many researchers to follow the principles of Occam’s Razor, which “promotes minimizing complexity and defending reductionism where possible (Oxford, 2022).” Calculating the most efficient and minimal network is an open problem, so researchers approximate with genetic algorithms. These algorithms aim to converge to a decent local optimum, not necessarily the global optimum. Genetic programming is an essential tool and recipient of significant scientific investment. Multiple dissertations could cover this topic, which is full of open problems.

*Insert a sequence diagram or something that demonstrates knowledge of this idea for interviews. That goes here.*

Modern network architectures aim to simultaneously solve multiple objectives regarding weight and structural parameters to maximize fitness with minimal design (Ünal & Başçiftçi, 2021). Researchers can optimize various dimensions of the problem concurrently using ensemble methods, provided those subtasks have similar but not overlapping objectives (Kim & Cho, 2008). These subtasks typically mutate the network architecture through additive and pruning strategies until convergence, as illustrated in Figure 1.

Figure : Multi-dimensional convergence (Kim & Cho, 2008, p. 1605)

Chart

Description automatically generated

### Deep learning era

Object detection and labeling tasks were some of the first problems in leveraging deep neural networks. Notably, in 2006, separate work by Hinton and Li led to the creation of ImageNet, a CV model for detecting twenty thousand labels based on fourteen million images (Ünal & Başçiftçi, 2021). In 2012, AlexNet incorporated graphic processing units (GPUs), reducing the error rate by 50% over previous CV architectures. Today, using GPUs over CPUs is table stakes and has opened the door to training across big data sets.

DNN architectures contain multiple kernels, regularization, and hyperparameters, and this variability makes them particularly challenging to optimize. Training hyperparameters control the model’s initial weights, learning rates, momentum factor, generalization, and the amount of training data (Jaisswal & Naik, 2021). These options influence several critical aspects of the final model, such as its sensitivity and degree of overfitting. Additionally, incorrect values can negatively impact training performance and defer model convergence.

Practitioners typically choose genetic programming or reinforcement learning (R.L.) for this procedure (Ünal & Başçiftçi, 2021). Data scientists can represent multiple expert systems as a connected mesh of R.L. models that search for ensemble methods like Kim & Cho (2008) articulate. This mesh approach is standard for state-of-the-art architecture competitions like ILSVRC (ImageNet Large Scale Visual Recognition Challenge).

## How does neural network training work

The objective of model training is to estimate the weights and connectivity structure for mapping a set of inputs to prediction outputs. This process requires optimizing a cost function using a series of forward-feeding data operations followed by backpropagation (Fridman, 2020). Backpropagation is a mathematical procedure that compares the expected versus actual outputs. Next, it calculates each parameter's partial derivatives and cascades updates to adjust the input weights accordingly (Lee & Yoo, 2021). Finally, after sufficient iterations, the network converges, which means that the difference between expected and actual outputs is within an acceptable margin of error.

### How are they scaling to millions of parameters

Recently, titans of the industry like Google Brain Team, OpenAI, and Uber Labs are scaling their evolutionary algorithms to millions of parameters without human intervention. Discovering the optimal architecture for CV and NLP problems at this scale is improbable due to its combinatorial nature and high costs per iteration (Lui, K, Fernando, & Kavukcuoglu, 2018). Instead, automation must represent the network as a traditional graph G consisting of nodes and weighted edges. Next, multi-level motifs augment the connectivity hierarchy to discover the impact of broad and narrow changes. For instance, these mutations might inject high-level filters or split the N-th hidden layer. In many ways, this is analogous to traditional fuzz testing, which exploits software through bit flipping.

Google DeepMind (2018) implements its search algorithm based on tournament selection. Each round begins by selecting the top 5% of the population and applying permutations from a discrete action space. After mutating the networks, only the superior offspring survive to the next round. This random search process is a genetic algorithm with an R.L. model steering its evolution to deliver the best of the breed through a simple search model.

### Modern scalability challenges

The sheer volume of matrix-based computations introduces challenges in scaling the training systems. Around 2012, researchers began using Graphical Processing Units (GPUs) over Central Processing Units (CPUs) to expedite model convergence significantly (Krizhevshy, Sutskever, & Hinton, 2012). According to some estimates, this hardware change leaped the state-of-the-art Deep Learning a decade (Ünal & Başçiftçi, 2021). However, researchers continue to require more extensive and sophisticated models. For example, GPT-3, an autoregressive language model, has over 175 billion parameters, a 10x increase over GPT-2 (Brown et al., 2020). More recently, Google Brain demonstrated an NLP translation model with over a trillion parameters (Fedus, Zoph, & Shazeer, 2022). This exponential parameter growth is likely to continue into the foreseeable future.

There are several practical limitations to *scaling up* an individual server to support training a billion parameters. Instead, researchers turn to distributed systems patterns for horizontally *scaling-out* across hundreds to thousands of commodity servers (Langer, He, Rahayu, & Xue, 2020). The orchestrator, parameter server, and the worker node are three essential roles within these compute clusters. Worker nodes communicate with the orchestration process to determine which training operation is next in the queue. Then, it fetches the shared parameter state and performs its work. Eventually, the operation completes, and the calculated gradients push to the parameter server. The parameter server is responsible for merging worker node updates into the shared state. This cycle repeats until the orchestrator determines that no more work is necessary.

### Fault-tolerant design requirements

The exponential growth in parameters and data volumes forces the training process to execute for days or weeks. That process must be highly reliable to withstand random errors during this period.

#### Influence of Hierarchy

Generally speaking, there are two mechanisms for modeling distributed systems, lists and trees. A list can efficiently manage small groups of related nodes; however, it can become cumbersome with more massive sets. Trees allow for more expansive designs as the system can hierarchically describe the problem through multiple levels of control. Consider the difference between Domain Name Services (DNS, tree) and NetBIOS (list). NetBIOS can easily manage a small branch office, not the Internet, because its simple flat list structure is *globalized*. In contrast, DNS has multiple subdomains, with each subdomain owned by heterogeneous service providers. Since each subdomain holds a specific set of children, read and write operations can be *localized*.

#### Influence of Partitioning

Localized designs are inherently more performant and fault-tolerant because of the containment of both scale and blast radius (Vosshall, 2018). Imagine a scientific dataset that has grown to several petabytes in size. The storage network would need to decompose this logical file system into multiple blocks and replicate it across multiple physical servers. These physical servers will run into mechanical failures, such as disk corruption or power outages.

When these outages occur, other nodes need to Setup, Challenge, and Repair (SCR) the missing data efficiently (Chen & Curtmola, 2017). The time necessary to perform that repair operation is proportional to the size of each block and the system’s ability to scale the reconstruction over multiple peers horizontally. Assume that 1TB of the dataset has entered a failed state and needs to recover across a 10GB/s network (see Table 1). If only one virtual peer has a copy of the data, the system will heal in 102.4 seconds. Then contrast that with the smaller block size of 128GB and which can economically be sprawled across many servers, reaching an MTTR of under a second.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Repair 1TB of Data | | |
| Block Size | **Virtual Peers** | **Num Blocks** | **MTTR (s)** |
| 1024 GB | 1 | 1024 | 102.4 |
| 8 | 1024 | 12.8 |
| 16 | 1024 | 6.4 |
| 512 | 2 | 2048 | 51.2 |
| 16 | 2048 | 6.4 |
| 32 | 2048 | 3.2 |
| 256 | 4 | 4096 | 25.6 |
| 32 | 4096 | 3.2 |
| 64 | 4096 | 1.6 |
| 128 | 8 | 8192 | 12.8 |
| 64 | 8192 | 1.6 |
| 128 | 8192 | 0.8 |

Table : Mean Time to Recover

#### Influence of Fail-Over Groups

Proxy servers and similar brokers operate on ephemeral requests and need fault tolerance to come from a different source. One strategy is maintaining a target group of service instances and monitoring their availability (see Figure 1). The monitoring can come from at least three reference points: (1) the network operating system, (2) the observed traffic of the broker itself, and (3) a local health agent on the service instance. The broker can use the Observed Health State Store (OHSS) to select the most appropriate receiver as new requests arrive.

A recovery policy could also exist to manage any Service Level Objectives (SLO) of the backend application. For instance, if the backend application needs to be highly available, the broker could be augmented to trap specific exceptions and automatically route to another node. Other systems need to optimize across different metrics, such as more consistent response time, and choose completely different behaviors.

Diagram

Description automatically generated

Figure : Broker Fail-Over

#### Influence of Geo-Redundancy

Cloud Service Providers (CSP) allow fault tolerance across multiple physical regions, so that entire data centers can fail without impacting applications uptime (see Figure 2). The scheme starts with deploying the service stack into two or more locations like Seattle and New York. Next, data store replication enables the sites to be kept in sync. Finally, the user can discover the most performant service stack instance from a location-aware Canolical Naming Service (CNAME). That system can consider latency and other metrics, similar to the proposed Fail-Over Group solution.

A picture containing text, map, table, indoor

Description automatically generated

Figure : Multi-Region Deployment

#### Influence of Consensus

The physical distance between the sites forces the need for eventual consistency protocols that range in complexity from (a) the latest timestamp wins, (b) Paxos algorithms, and (c) Byzantine General’s solutions (Zhao, 2014). The latest timestamp wins, is easy to understand but needs to rely on highly reliable distributed clock synchronization, an open research problem in itself (Ting, Chun-Yang, Di, Xiao-ming, & Heng, 2014). Under Paxos (see Figure 3), multiple rounds of preparation, acceptance, and learning phases take place to gain consensus. This elegant protocol can efficiently reconcile a single systems image, provided none of the nodes are malicious. If malicious or erroneous nodes exist, 3f +1 cross-validations need to occur (Zhao, 2014).

Diagram

Description automatically generated

Figure : Paxos Consensus (Zhao, 2014, p. 196)

#### Influence of Protocol

Message passing between components can either use reliable or unreliable communication. Unreliable handoff can be helpful for best-effort or performance-critical systems, such as real-time video or sampled telemetry reporting. Reliable handoff is crucial for scenarios that mandate full and consistent accounting, such as user data or financial records. These fault tolerance decisions are not limited to the low-level transport protocol differences between User Datagram Protocol (UDP) and Transmission Control Protocol (TCP). They also appear at higher application levels (see Figure 4). The actor can notify the Alice service directly; however, the message could become lost due to a network failure. Instead, they can first place the payload into a command queue and remove it only after the server-side acknowledgment. When Alice accepts the event, she needs to receive confirmation from Bob and Charlie before returning success. Bob chooses to store the event in a durable command queue, whereas Charlie executes it directly. In either scenario, the client can reliably infer that handoff has occurred.

Diagram

Description automatically generated

Figure : Handoff Protocol

### Scheduling-specific design requirements

The training process must also choose an asynchronous or synchronous scheduling architecture (Langer, He, Rahayu, & Xue, 2020). Synchronous systems partition the training iteration (epoch) across the worker nodes and merge all responses before starting the next round. This approach has several advantages, such as simplicity regarding the design and maintenance. However, as the total worker nodes increase, so do the idle cycles from waiting on the slowest tasks to complete. Asynchronous systems aim to eliminate these idle cycles by starting the next training round prematurely. Since each worker node owns an isolated task, this technique is highly effective. However, the parameter server and orchestration processes must assume more complexity for this time-optimization. For instance, what should the parameter server do when task outputs (gradients) are late or never arrive?

Langer et al. (2020) propose a collection of boundary conditions for asynchronous architectures. For example, the parameter server system should disregard any gradient exceeding K-epochs latency or changing an individual parameter by X-threshold delta. They also advise using priority queuing over FIFO queuing for the incoming responses. Another solution is to use multiple parameter servers and periodically reconcile the differences between their local states.

### Continuous learning techniques

Countless researchers and product teams periodically retrain their models from scratch as they iteratively add more example data. This approach is intuitive to design but wasteful and could become impractical as model parameter counts continue exponentially growing. Instead, researchers are proposing training architectures that support *continuous learning*.

#### On-device training architectures

On-device training architectures aim to fine-tune generalized machine learning models using local sensor data (Lee & Yoo, 2021). This design begins with a cloud training process that consumes a public data set to produce a shared model. Next, a second training process will download the shared dataset and fine-tune it to include personal data (e.g., a smartphone’s photo album). Lastly, an inference process uses the private model to deliver a personalized experience.

There are several essential advantages to this design pattern. First, the cloud-based training process does most calculations and has significantly more resources than an individual device. Next, the device-based training process can maintain data privacy and residency requirements because the personal data doesn’t need to be in the public data set. This characteristic could help overcome specific privacy concerns in regulated industries like healthcare. Third, the decoupled training processes enable the refreshing of both models at a higher frequency.

Diagram

Description automatically generated

#### Stream-based training architectures

## What is autoencoding

In its simplest form, “an autoencoder learns the representation or code by trying to copy the input to the output by encoding the input’s distribution into a low-dimensional vector (Atienza, 2018, p. 78).” Figure 2 contains an example architecture illustration with three features compressed into one parameter before expanding into three new features. The precise connectivity graph depends on the specific situation. For instance, researchers can use this process for scenarios such as colorizing images, denoising, replicating artistic styles, and intrusion detection, among other conditions.

Figure 2: Autoencoding architecture

*Diagram

Description automatically generated*

### Example usages

*Include a summary of the art stylizer paper*

*Include the summary of DACNN here from the intrusion detection paper.*

## How does sequence analysis work

Natural Language Processing (NLP) sits at the intersection of artificial intelligence, human language, and computer science.

### Language Parsing

NLP systems typically begin with sentence normalization, combining and annotating tokens, and finally performing custom business logic (see Figure 1) (Edureka, 2018). Using Lemmatiziation and Stemming strategies enables the parsers to reduce the variability between sentences, such as removing verb-tensing. Next, annotations are associated with the words by subsystems like Named Entity Recognition (NER) that discover the sentence’s critical components. After chunking related tokens together, the scenario-specific business logic can operate on a semantic representation of the text. Depending on the use case, these steps could be massive subsystems or single lines of code.

Figure 1: NLP Process

Diagram

Description automatically generated with medium confidence

### Deep Learning

NLP appears across various use cases like language translation, speech-to-text, and sentiment analysis. In biology, animal brains accomplish these tasks through meshes of neurons that transmit signals across connected synaptic (transforming) and activation (filtering) links (Keller, Liu, & Fogel, 2016). Computer scientists mimic this behavior with Deep Learning on Neural Networks, essentially weighted graphs. Generally, NLP architectures use Recurrent Neural Network (RNN) structures containing connectivity loops to previous layers (see Figure 2). More advanced designs include subnets for memory retention (see Table 1), encoding and decoding segments, and greater parallelization from attention vectors (Fu, 2019). Researchers and engineers can add or remove these subsystems to optimize a specific use case.

Table 1: Example progressions of N.N. architecture complexity

|  |  |
| --- | --- |
| Algorithm | Description |
| seq2seq | Simple Recurrent N.N. (RNN) for a token sequence to sequence prediction. These systems are easy to implement but lack memory |
| Long Short Term Memory | Extends the seq2seq by including a “long term” cache to hold context information |
| Transformers | State of the art solution for massively parallel NLP through attention vectors and position encoding |

Figure 2: Abstract Diagram of Differences

Diagram

Description automatically generated

### Feature Extraction Process

The first steps to any business intelligence problem are identifying the specific questions and locating facts to support answers (Snee, 2015). When researchers ignore this preparation, it produces garbage-in/garbage-out results. For instance, Alsudias et al. (2014) built an NLP system for predicting where the user was during the submission (e.g., restaurant or nightclub). Their approach extracts keywords from Yelp reviews (using term frequency), business metadata (e.g., name and location), and tweet metadata (e.g., timestamp). These features flow into a random forest classifier that determines the user’s location with a 74% accuracy. However, using only the business metadata produces an 88% accuracy, indicating that these additional details provide negative value.

NPAC has specific requirements to model social media users’ speech patterns and create new content in their voices. The Feature Extraction Process must therefore consider the user’s metadata (e.g., age and locale), the online community properties (e.g., forum name), the posted content, and any quality ratings (e.g., Facebook Likes). There are several considerations to augment this process. For instance, adding a filtration step to remove comments with negative ratings might create more well-liked personalities. However, it could also be advantageous to generate trolls that argue an alternative position, reinforcing NPAC’s position that the other side is illegitimate or less sophisticated.

### Training Process

Around 2014, GAN (Generative Adversarial) Networks became the state-of-the-art approach to producing high-quality fabricated content (Fridman, Deep Learning State of the Art, 2020). These systems utilize a feedback loop between a Generative N.N. (GNN) and Discriminator N.N. (DNN). Each iteration outputs a ‘Deep-Fake’ asset and assesses its likelihood of being legitimate. This process enables both systems to learn from one another, continuously improving. According to Fridman (2020), detecting Deep Fakes is an arms race because advances in DNN naturally improve GNN results. NPAC leverages this methodology for self-teaching its systems to deliver more accurate content (see Figure 4). The organization’s solution uses the NLP transformer to improve parallelization over LSTM and a second RNN classification network. Periodic snapshots archive the content and model state for offline troubleshooting use cases during the training process.

Figure 4: Training Configuration

Diagram, schematic

Description automatically generated

## How does human activity recognition work

HAR is a critical component for health care

### Traditional methods

*There are currently no documents for this section. Need it?*

### Deep learning methods

* Learning models for HAR (2021)
* Survey on HAR using Sensors (2021)
* Survey on DNN for HAR (2021)

### Graph CNN methods

## Recognizing human activities

One critical application of CV is to detect human activities from photos, images, and video streams. This capability is essential for personalizing systems across the healthcare, smart home, and safety industries (Gu et al., 2021). Adapting traditional ML tactics to human activity recognition (HAR) is laborious, error-prone, and challenging. Researchers mitigate these issues with deep learning models (Gu et al., 2021; Banjarey et al., 2021). There are several algorithm families used to model these predictions.

### Restricted Boltzmann Machine (RBM)

The first HAR implementations used Deep Belief Networks as their prediction basis. Training this solution was extraordinarily challenging and deprecated (Gu et al., 2021).

### Convolutional Neural Networks (CNN)

A preceding section examined the biological constructs that enable primates’ vision and nature’s solution.

A neural network consists of three building blocks' input, hidden, and output layers. For instance, an animal image classification system might assign 64x64 pixel images into ten predetermined categories. This example requires an input layer with 4096 neurons, an output layer of ten neurons, and some hidden layers in the middle. Adding more hidden layers enables extracting more details from the image, similar to object edges (layer-1), ears (layer-2), cat’s ears (layer-3), and a tiger’s ears (layer-4) (Fridman, MIT 6.S094: Introduction to Deep Learning and Self-Driving Cars, 2017). While more complex networks can extract more insights, it comes with the cost of needing exponentially more data to train the model.

Experts suggest that a fully trained model requires at least ten observations per parameter (Snee, 2015). Depending on the connectivity configuration, this can become too expensive and require model compression strategies (Cheng, Wang, Zhou, & Zhang, 2018). For instance, the input layer could feed into a series of pooling transforms that downgrade the resolution by averaging every 2x2 pixels. Another tactic might focus on connecting and evaluating local segments of neurons before outputting into global join constructs and prognostication output (see Figure 5). Meanwhile, other situations like estimating housing prices perform better with fully connected shallow pipelines. While standard architectures exist for many classes of predictions, some experimentation is necessary.

Figure 5: Network Structure

Diagram

Description automatically generated

### Recurrent Nural Networks (RNN)

### Generative Adversarial Networks (GAN)

### Graph Convolutional Neural Networks (G-CNN)

## Intelligent Agent Modeling

Engineers consistently find that maintaining monolithic technologies requires substantial overhead. Alternatively, using microsystem architectures enables them to build and replace components rapidly in isolation. A similar idea exists with simulations with a decomposition of the environment into multiple intelligent agents (see Table 3).

Table 3: Principal Components

|  |  |
| --- | --- |
| Aspect | Definition |
| Intelligence | The ability to reason about a problem |
| Simulation | An experiment that produces a statistical model |
| Environment | The universe contains the agents |
| Agent | An automaton that follows a predefined script |
| Objective | The goal of the agent |
| Tasks | The steps necessary to complete the objective |
| Notification | A collaborative or competing message between agents |
| Swarm | A group of agents |
| Choice | The random decision of an agent within its action space |
| Aggregate Choice | The net effect of multiple independent agent decisions |

A simulation experiment first identifies the environment, participants, and one or more objectives. Each participant, called an agent, attempts to complete their aim under guiding rules and principles. For instance, NetLogo’s BeeSmart environment contains multiple bees that try to maximize food production from various honey pots within a given scene (Wilensky, 2014). Initially, the swarm fumbles around until it discovers a food source. After some time, the colony will divide across multiple honey pots and compare site values with neighboring peers. Eventually, the bees converge to the optimal configuration that provides the maximum food for the hive.

Figure 1: BeeSmart Simulation (Wilensky, 2014)

A screenshot of a computer

Description automatically generated with medium confidence

While no individual agent (bee) understands the ideal distribution across the environment, the aggregate of independent decisions enables analysts to extract sophisticated observations about the broader objectives. It is also possible to quickly expand upon this simulation by designing expert agents, such as communication specialists, that propagate messages twice as fast. After defining the role and its local rules, the existing simulation can immediately incorporate those customizations.

### Genetic Algorithms (G.A.)

The Traveling Salesman is a classical graph puzzle that attempts to find the most efficient route through N-cities. Even with ubiquitous access to cloud computing, enumerating through an exhaustive search is not practical due to the combinations growing at (Keller et al., 2016). As the simulation continues to scale out, it requires a mechanism to prune that search space and quickly discover the optimal answers. The Theory of Evolution states that biology weeds out inferior strains through the Natural Selection Process (Darwin, 1859). Computers can replicate this model through Genetic Algorithms to converge on optimal configurations (see Figure 2).

Figure 2: Genetic Algorithm

Diagram

Description automatically generated

The solution begins by modeling a potential answer as a vector of classification features. First, hundreds to thousands of randomly initialized instances run through the simulation to compute a per-instance score. Then a TOP-N ranking keeps the best instances, discarding the remainder. Next, a cross-breeding and mutation process mixes features from the fittest combinations to produce the offspring. Those offspring cycle through this system thousands of times until only superior specimens remain.

### Multi-Level Agent-Based Modeling (ML-ABM)

After decomposing complex models into individual agents, a mechanism must aggregate the independent decisions into more macro observations. Cellular Automata (C.A.) paints this picture by grouping related swarms into “a hierarchical series of discrete systems (Makarenko & Osaulenko, 2018).” Through multiple levels of aggregation, agents can feed into swarms and those individual swarms into swarm networks.

For instance, a financial market environment has individual buy-and-sell participants who react to supply-and-demand fluctuations (see Figure 3). This specific example simulation contains thousands of personal portfolio accounts (agents) that frequently make rational transactions. An analysis could apply C.A. across these portfolios by aggregating the multitude of data points to improve the data’s usability for professional traders. However, an inefficiency exists within this design because some individual portfolios (agent states) are nearly identical. Like the risk-free rate, other aspects do not require the fidelity that swarms of agents produce. These situations can rely on ML-ABM to approximate irrelevant details (e.g., with caches) and enable fine-grained influence over critical decisions (e.g., with swarms of agents) (Hijorth et al., 2020).

## How do dynamic environment simulations work

### Methods

### Unity-based

## Computer vision and autonomous driving

*Include transition here. CV is already demonstrating value in improving passenger safety. Let's look at how they approach the problem.*

Machine learning can enhance every aspect of the drive, from extending the physical parts’ lifespan to increasing the driver’s overall satisfaction. Figure 1 contains a non-exhaustive taxonomy of uses-cases for artificial intelligence in motor vehicles, such as reducing wear and tear and object detection. The central application topic is the primary deciding factor for item assignment within the tree. Many items, such as Voice Assistance (V.A.), could arguably live under a different pillar, safety. However, safety systems could exist in the same capacity using more traditional input interfaces, making this example fall under convenience.

Figure 1: Taxonomy of Example Use-Cases

### Data collection process

Since covering each use case in full detail would fill multiple books, this section reviews these user scenarios’ commonality. The lifeblood of these systems is data, and only through synthesizing information into knowledge can they be more adaptive. These processes require collecting telemetry, mining data, and modeling the interactions (see Figure 2). As simulations run across that model, statistical distributions form, leading to predictive capabilities. When the model’s complexity grows or the required accuracy increases, the learning system also needs more examples to cover each scenario.

Consider the analogy of building an All-Wheel-Drive (AWD) feature that only knows about Florida’s flat tropical roads. Despite the engineers ' best efforts, the vehicle will encounter challenges on Colorado’s ice-covered mountainous climbs. Similar behaviors exist across the autonomous vehicle supply chain, demanding either (a) more data or (b) more erroneous assumptions. This trade-off introduces acceptable feature risks in specific situations (e.g., entertainment modules) and undesirable consequences for others (e.g., safety modules).

Figure 2: System Design

A picture containing text, map, indoor

Description automatically generated

Around 2014, GAN (Generative Adversarial Networks) became the state-of-the-art approach for constructing high-quality detectors and fabricated content (Fridman, Deep Learning State of the Art, 2020). These systems utilize a feedback loop between a Generative Neural Network (GNN) and Discriminator Neural Network (DNN). Each iteration outputs a ‘Deep-Fake’ asset and assesses its validity (see Figure 3). Under this process, both systems learn from one another, continuously improving their expertise.

Figure 3: Training Configuration

Diagram

Description automatically generated

Automotive companies like Formula One use this methodology to synthesize more efficient race cars that can safely operate at high speeds (Smedley, 2019). First, a collection of features (e.g., car shape and weather data) load into the GNN and its simulation environment. Next, the output flows into a DNN, assessing the solution’s feasibility (e.g., wind drag and safety requirements). Then, the GNN modifies the solution’s parameters to search for a higher score (e.g., faster car). After executing thousands of cycles, both networks converge with optimal solutions to deliver innovation (GNN) or detect problems (DNN).

### Safety Control Systems

Annually, 32,000 Americans die from automotive accidents, and another 2 million are injured (CDC, 2016). These statistics are unacceptably high and require innovations that increase all participants’ safety on the road (see Figure 2). Artificial intelligence can assist in these scenarios by collecting sensor data and predicting risks and opportunities. However, numerous open problems exist throughout the safety domain. These challenges should not discourage investments in these areas as they are essential to public safety.

Figure 2: Taxonomy of Participants and Example Challenges

For example, several manufacturers, like Subaru and Lexus, include audible collision alerts during lane changes or reversing. While these capabilities exist today, they are often incomplete models due to the high volume of edge cases, such as children fetching a ball from the street. Several open problems span ethical and philosophical debate even after detecting the example child. Lex (2017) asksif avoiding the pedestrian requires killing the driver— what calculus dictates that autonomous decision? These situations might play out in fractions of a second, limiting the value of human intuition. Since concrete answers do not exist, machines must resort to static guardrails (e.g., slamming on the breaks or swerving) that could risk a multi-vehicle accident.

### Convenience Systems

There are often long monotonous segments necessary to arrive at the destination during a road trip. This requirement forces the driver to expel significant amounts of concentration relative to the mundane task. Instead, auto-pilot systems can take the wheel and allow the motorist to relax and participate in leisure activities (e.g., conversing with passengers). Some manufacturers tackle these needs with adaptive cruise control technologies. This approach is helpful in static environments (e.g., open highways) but encounters limitations in more dynamic environments (e.g., urban cities). Researchers are closing this gap by including more sensors that feed into sophisticated reinforcement learning algorithms (Fridman, 2020). However, continuing to scale these monolithic expert systems is challenging.

Figure 3: Example Microservice Architecture

Diagram

Description automatically generated

V-TORCS (Virtual The Open Racing Car Simulation) and other modern architectures address these issues using ensemble and multi-task learning methods (Li et al., 2019). Consider a decision process that feeds camera frames into an image classification Convolutional Neural Network (CNN) to extract objects and contextualize the environment (see Figure 3). CNN algorithms mimic an eye’s biological structures by normalizing neighboring pixel blocks to derive structure (Keller et al., 2016). These results flow into various expert subsystems that control the car, such as turning the wheel or accelerating. An ensemble of subsystem observations merges into a broader system-wide decision model that contains one or more I/O (Input/Output) controllers. Like other microsystem architectures, each subsystem’s implementation can evolve independently of peer components—enabling greater agility and innovation.

### Optimization Systems

Modern personal vehicles have a lifespan of over 200,000 miles and often travel 24,000 miles per year (Ford, 2012). Assuming a driver purchases a $25,000 car and keeps it that entire usable period, they will likely spend at least that much on fuel and repairs (see Table 1). Data scientists can improve this situation by introducing micro-optimization systems across the automobile. For instance, traditional cruise control maintains a specific speed (e.g., 70mph) without considering any environmental context. Meanwhile, a more intelligent system can factor in the road’s incline, the driver’s profile, and metrics about the trip to create a dynamic profile ranging from, e.g., 65-70mph. As this idea expands outward, it collects micro-optimizers that monitor all aspects of the driving experience, potentially saving hundreds of dollars in costs.

Table 1: Ongoing Fees

|  |  |
| --- | --- |
| Line Item | Total |
| 200,000 miles @ 30 miles/gallon | 6,667 gallons |
| x $2.50/gallon | $16,700 |
| + Typical Repairs | $10,000 |
|  |  |
| Total Costs | $26,700 |

When the driver has advance notice that a component is likely to fail, they can schedule the maintenance and minimize costs. Preventative Maintenance Systems (PMS) provides this capability by collecting component-level telemetry and looking for anomalous metrics (see Figure 4). For instance, an engine monitoring solution might observe the RPMs (Revolutions per Minute) and the electrical output. Sensor time series data are noisy and require a curation process (e.g., Kalman filter) to derive a stable moving average signal (Jackson & Rege, 2019). Next, the curated call flows into a Recurrent Neural Network (RNN), which uses sequences of previous tokens to predict future values (Keller et al., 2016). An anomaly exists when new observations deviate from these predictions and needs to surface in a decision control process.

Figure 4: Preventative Maintenance System

Diagram, schematic

Description automatically generated

### Smart City Integration

The future evolution of city planning makes urban areas highly connected with fast wireless networking and intelligent machines emitting enormous telemetry data volumes (Balduccini et al., 2018). The autonomous vehicle is a central participant in this ecosystem, as it transmits metrics to infrastructure (V2I) and other vehicles (V2V). These metadata feeds will enable more efficient traffic shaping, alertness to potential risks, and more insights into driver patterns (Tong et al., 2019). For instance, each driver inputs their GPS destination and follows the route in an isolated silo. Tomorrow, aspects of those routing decisions can become centralized, resulting in less route congestion. While ideas of this ideal state are already coming to life in major cities like Seattle, Boston, and New York, the large-scale implementation is still years out (Cohen, 2013). Completely modernizing these areas will require significant infrastructure investments, consensus on V2X communication protocols, and machines that implement those standards. Machine learning technologies will need to synthesize those capabilities by making predictions by pairing vehicle-local sensors with ubiquitous cloud services.

### Observations

Three takeaways of this paper are (1) that artificial systems need to augment human processes; (2) a suitable starting place is safety and cost optimization; and (3) intelligent cities are several years away, requiring cloud services to fill that void.

The most powerful artificial intelligence applications use machines to enhance human capabilities rather than replace them. Motor vehicles contain many integration points. Machine learning can handle tedious aspects of the journey with greater precision and accuracy than humans. After freeing the driver from the burden of mundane work, she can focus on value-differentiating traveling qualities, such as talking with passengers and thinking about the day ahead. Meanwhile, intelligent systems remain vigilant in the background looking for risks like a child running into the road. Another collection of machine learning tasks revolves around cruise control capabilities. These systems keep us out of harm’s way and reduce wear-and-tear on internal parts. Looking further into the future, the integration of intelligent vehicles within smart cities promises more efficient traffic shaping and risk awareness. However, the necessary infrastructure investments are unlikely to arise in the next decade. Instead, machine learning will pair vehicular telemetry into ubiquitous cloud computing to provide a similar experience.

## How does the reproducibility crisis impact ML design

There is an abundance of non-reproducible experiments because researchers do not account for nuances in the data collection (Rivera-Landos, Khomh, & Nikanjam, 2021). These challenges originate from *Non-Determinism Introducing Factors* (NDIF), such as software updates and defects, hardware-specific differences, data uniqueness, randomization seeding, and dropout rates, among other reasons.

### Sources of Non-Determinism Introducing Factors

NDIF issues also originate from incorrect industry assumptions and unknown unknowns. For instance, software taint analysis is a process for discovering security-critical variables and parameters within an application. Generally, these tools parse the underlying bytecode through language-specific Reflection APIs. Next, they build call trees and use graph analysis methodologies to discover vulnerable code paths (Kilgallon, De la Rosa, & Cavazos, 2017). When researchers publish new techniques, they present their findings against existing open-source products like FlowDroid, AmanDroid, and DroidSafe. However, the detection rate for those tools is highly-dependent on configuration-specific settings (Qiu, Wang, & Rubin, 2018). This subtle dependency can be problematic for researchers and small businesses that lack the technical expertise necessary to support multiple competing toolchains.

It’s challenging and expensive to recreate many foundational experiments, so researchers must assume that previous authors are correct. Unverified facts present a significant risk that can have cascading ramifications. Consider Majoranas, an extremely sensitive nanowire that operates at absolute zero temperatures and in extreme magnetic fields, to measure individual electrons within quantum computers. This precision “plays an important role in protecting quantum information and enabling reliable computation (Langston, 2022).” Limited access to equipment forces many researchers to rely on computerized physics simulation processes. Recently, Pennsylvania State University could not reproduce several foundational studies and received opposite results (Frolov, 2021). After disproving basic expectations, researchers revisited other assumptions and found at least six other core tenants were false. This situation highlights the criticality of revalidating technological assumptions. The body of knowledge is never precise and continually evolving with new facts.

Negligence and maliciousness also create reproducibility challenges. Miyakawa has handled 180 manuscripts since early 2017 and identified 41 potentially fabricated experiments. When he requested raw for these studies, “more than 97% [(40 of 41)] pulled their publication request, suggesting a possibility that the data didn’t exist from the beginning (Miyakawa, 2020, p. 1).” The editor claims to spot these situations by looking for data too perfect, error rates improbable, and study impact too significant. It’s relatively easy to catch blatant lies, though humans commonly only cheat a little (Ariely, 2009). Behavioral economists consistently demonstrate that people mispresent small details that lead to a better story with believably exaggerated results (e.g., an 75% accurate score becomes 82%).

### Influence of societal norms and ethical design

Hudson (2021) argues that researchers should focus on replicability over reproducibility. He identifies incorrect study design, not disproving the null hypothesis, wrong statistical methods, societal norms, and publication bias, among other factors. These factors impact research reproducibility. Therefore, researchers should accept that incorrect facts exist, and that’s because humans aim to prove what they believe. Douglas & Elliott (2022) responds to Hudson’s article, asserting it conflates value-ladenness with bias and mispresents values as evidential factors. They state that researchers are generally well-intentioned and aim to make reliable, repeatable studies. However, it is impractical for those practitioners to wait for results to be flawless, as this means science no longer evolves. Put another way, “all models are wrong, but some are useful (Das, 2019).” Academics should perceive the book of knowledge as continually developing principles and tenants, where more precise instruments will eventually supersede those ideas.

### Role of ethical design

Researchers need to manage ethical challenges that arise from their work. These issues originate from societal norms and internal biases. While several frameworks exist to guide the conversation, they can be ambiguous or focus on a subset of the problem.

Ethics are a system of moral principles that dictate the norms of a group. Societies implement these systems through social constructivism, enabling and constraining the group’s actions (Burr, 2015). Communities leverage this mechanism to assign truths and infer values about concepts (Gergen, 2010). Consider a project that seeks to prove that men are superior to women. Within a chauvinistic cohort, these results align with their world views and are ethical. However, a diverse group would chastise the very idea, regardless of methodology. Further complicating the matter, ethical identities are dynamic and evolve (or regress) over time.

Scholars need to understand their audience and the group’s customs. These social contracts limit the researcher’s influence and ability to solicit their work. These implicit rule sets vary between cohorts, making it impossible to remove these subtle biases entirely.

### Internal biases

~~Researchers need to understand their internal biases. Everyone has historical and cultural defaults that lead to prejudices. These subtle classification differences influence language and construct our reality~~ (Owen, 2017)~~. Picture two people, one fat, another thin. Then change those definitions to obese and anorexic. Did all four imagined people have the same gender and race? Words matter and one needs to choose them carefully. These biases sneak into our written and verbal communication. They cause us to gloss over issues of Diversity, Equity, and Inclusion (DEI). For instance, the terms such as whitelist and blacklist have a racial connotation. These modifiers become a sub-conscience reinforcement that one’s worldview is the only perspective.~~

### Sources of ethical frameworks

Numerous professional, regulatory, and advisory groups create frameworks that outline strategies for approaching ethical designs. These professional standards can contain conflicts of interest, hidden agendas, and inconsistent moral standards (Tan, 2021). The Belmont Report (1979) famously defines three core principles: respect for persons, beneficence, and justice. These tenants ask researchers to treat everyone fairly and avoid harm. However, even this simple statement has ambiguity.

After forty years, the ethical code requires modernization to align with the evolving worldviews. Adashi et al. (2018, p. 1347) argue that the Belmont Report’s “distinction between research and practice is disappearing within the commercialization of present-day research.” Businesses actively debate the definition of “harm” and propose a notion of “harm versus setback.” This worldview states that any action that is not directly harmful is, at worse, an indirect setback. Roberts (2021, p. 15) proposes that researchers “must focus on risks of the research process itself, not outcome-related risks as downstream consequences are beyond the purview of ethical gatekeeping.” Facebook has a moral (and potentially legal) mandate to protect its user’s privacy. Under Robert’s definition, the social-media juggernaut can ethically track relationships between billions of people. However, it is not bound to prevent malicious auxiliary use-cases (e.g., election interference). While this position resonates with specific cohorts, it faces fierce opposition from others.

### Controversial Subjects

Many academic and business communities embrace Diversity, Equity, and Inclusion (DEI) concepts. These ideas are becoming mainstream, and that will cause them to become shared truths and social norms. Researchers that fight against this force are likely to find exclusion and isolation (Owen, 2017; Adashi et al., 2018). Instead, they should adopt the social standards and assume “people are people.” However, this is often easier said than done. Human data sets contain numerous highly correlated variables (e.g., race and income). These statical properties prevent merely dropping an individual column and making the results racially neutral. Researchers can explicitly call out the risk in their findings, but fully addressing this situation is an open problem.

*~~Talk about Hudsons perspectives here that it’s a function of reproducibility versus replicability~~. He says that there’s a lot of challenges with statistical methods and that segways into Parker and BachmeierNTIM7101-5.docx ~~There’s also a rebuttal paper that should come into scope.~~ I had a paper that talked about incorrect statistical methods that’s probably from data mining or statistics. That would go nicely into this section.*

## Ethical considerations of A.I.

Artificial intelligence is a scary black box that spreads malicious propaganda, destroys jobs, and seeks to destabilize honest citizens’ values. This statement is intentionally farcical, yet it also touches on fundamental concerns of ethical A.I. designs. People fear what they do not understand and use science-fiction to fill these gaps. Within those futuristic worlds, machines become the dominant species that control every decision of an enslaved human population. However, several challenges prevent this transition of power from becoming a reality, such as intelligent systems that lack actual *intelligence* (Wildberger, 1996; Hole & Ahmad, 2019; Upchurch, 2018). Instead, organizations need to assess these tools rationally, explore applications that enhance human capabilities, and remove undifferentiating overhead.

### Roles of Artificial Intelligence

Despite artificial intelligence already being well-entrenched in everyday life, there are concerns about its role. First, does the advancement of machine learning mean fewer jobs? Second, of those remaining jobs, are humans giving away control unnecessarily? Third, are those machines capable of manipulating the public to steal dominion?

#### Role in Employment

Before 1949, digging a ditch would take hours or even days with a crew of manual workers. After the invention of the backhoe, these jobs required less time with fewer employees. From the organization’s perspective, these efficiencies translate into faster time to market at lower costs. Meanwhile, the former diggers became displaced into new roles, repairing, operating, and supervising the machinery. Each of these positions requires entire supply chains of support. For instance, it takes factories to produce the backhoe parts, each staffed with hundreds of blue-collar jobs. Cities must also build universities and technical schools to train team members to fill these roles, expanding the job market.

Similarly, modern businesses are actively seeking methods that reduce costs and improve efficiencies through automation. The most powerful artificial intelligence applications use machines to enhance human capabilities rather than replace them (Heer, 2019; Boire, 2017). For instance, a person can write a more profound business case than a machine; however, the same device will have fewer grammatical errors. This dichotomy exists because humans specialize in contextualizing thought versus using patterns to make predictions (Schleer et al., 2019). Many professions exist as a combination of decision-making, pattern recognition, and mechanical tasks. Expert systems address specific aspects of the job requirements; however, superseding the soft skills that unify these role components is challenging (Huang et al., 2019).

Specific low-skilled jobs, such as bank tellers and office clerical staff, are at risk of being replaced (Hamid et al., 2017). Similarly, expert pattern matching tasks like identifying tumors in MRI (Magnetic Resonance Imaging) becomes commoditized through A.I. systems. Given the lower entry barrier, some low-skilled workers will transition to better-paying jobs that operate those sophisticated and commoditized systems. For instance, many workers cannot access foreign markets due to language and communication limitations. Artificial intelligence can aid these in these translation scenarios while leaving control with humans.

#### Role in Decision Making

Many decision-making processes can benefit from machines providing recommendations and validations. For instance, a court judge could use an intelligent system to assess how their sentencing aligns with existing norms. Perhaps the device predicts the defendant should receive five years of probation, while a judge considers fifteen years in prison. When the validation check expresses such a difference in opinions, it could suggest that unconscious bias is taking place and warrants additional considerations. That bias either provides ammunition for appeals processes or incarcerates people unjustifiably long.

While this approach has much potential, there are concerns that professionals arbitrarily accept recommendations. However, these challenges occur everywhere that automation controls the ‘last mile’ of decision making. If the suggestion comes from a machine or peer, the person in charge of the process must be accountable for the final call. Blindly delegating control to machines is dangerous because learning algorithms are greedy, brittle, rigid, and opaque (Hole & Ahmad, 2019). Until artificial brains can rationalize abstract thought, humans must perform this task.

#### Role in Manipulation

Modern censorship does not restrict free speech; instead, it increases the noise and drowns the signal (Thomas, 2019). Fundamentally, marketing campaigns and propaganda machines follow the same process of Segmentation, Targeting, and Positioning (STP) (Kane, 2019). Delivering on this objective requires pattern matching, content delivery, and human intuition. Automation is well-suited for these tasks and can use social media channels, like Facebook and Twitter, to connect with billions of people and manage significant portions of those interactions.

Congressional and media sources raise ethical questions around the ease of access to these capabilities for political manipulation. Unfortunately, these questions are mostly talking points rather than a call for action. Artificial intelligence comes with many abstract concepts that do not fit within the complex and opaque legal language (Guiffrida et al., 2018). For instance, machines cannot reason about their instructions, so can the courts hold *them* accountable? Perhaps the system designers should be responsible for their creations. However, the algorithms are primarily algebraic formulas controlled by end-users. Without a mechanism to define and enforce a standard operating behavior, it is impossible to expect a different outcome.

### Design Considerations

Two recent attempts to define this process for ensuring ethical A.I. are the European Union’s Ethics Guidelines for Trustworthy A.I. and the OECD’s Principals of Ethical A.I. (E.U., 2019; OECD, 2019). Both documents describe the need for artificially intelligent systems to be human-centric, transparent, explainable, robust, and secure.

### Human-Centric

Robotics’s Three Law states that automation should not injure humans, ignore people’s commands, and protect their existence (Asimov, 1942). These rules lay a foundation for the idea that devices exist to cooperate and enhance humanity. Unfortunately, the machines cannot reason and are bound to their program designs. Since machines cannot devise these criteria independently, it becomes the system engineers’ responsibility to enforce these requirements. Those decisions are predominately a matter of business priorities and vary across different use-cases. For instance, Lockheed Martin, a military weapons designer, views its human-centric role as protecting American interests at foreign nations’ expense. This perspective is radically different from other organizations yet equally valid.

### Transparent and Explainable

Artificial brains often rely on deep learning techniques through neural network solutions. These networks approximate a function that maps inputs and outputs through multiple non-parametric transforms. While data scientists can perform experiments to verify the model’s accuracy, they often cannot explain it (Gilpin et al., 2018). This limitation prevents broader adoption in places like the European Union, where the General Data Protection Regulation (GDPR) grants citizens a Right to Explanation.

Further complicating matters, neural networks learn the patterns we *ask*, not necessarily the ones we *mean*. For instance, Beauty.ai, an algorithm for rating female attractiveness, lost credibility due to only giving high scores to light-skinned candidates (Upchurch, 2018). This outcome was not intentionally malicious but the byproduct of not sufficiently representing minorities in the training set. Similar imbalanced issues occur across many real-world domains and require sophisticated data handling strategies (Kaur et al., 2019). Even with expert data scientists, it is possible to miss these edge cases and produce invalid predictions.

### Robust and Secure

Engineers who become data scientists follow a different curriculum than their peers who become security specialists. This distinction in training is most evident in the lack of controls across artificially intelligent solutions (Lin et al., 2018; Sethi & Kantardzic, 2018). Malicious actors can influence these predicted decisions by either inserting erroneous samples into the training set or directly attacking the probability distributions. For instance, researchers have shown that applying tiny amounts of distortion to images can change the graphic’s predicted class (e.g., cat versus dog) (Sethi & Kantardzic, 2018). If people cannot trust the classification algorithms' integrity, how can mission-critical environments effectively use them?

### Observations

Artificial intelligence is a tool that can automate mechanical tasks, pattern match data, and enhance human capabilities. Organizations can use these means to improve efficiency and reduce wastefulness. These innovations deprecate the need for specific skill sets and lower the entry barrier into other expert systems. While this causes an initial decrease number of jobs necessary, entirely new industries follow shortly afterward. This promotion justifies the short-term pain when a society can replace low-paying jobs with high-paying alternatives.

Machine learning technology is too immature to delegate business-critical decisions. Instead, professionals should consider these technologies for initial recommendations and verify that their choices are free of unconscious biases. For example, a court judge should assess their sentencing aligns with a regression algorithm’s prediction, not blindly issue that verdict. Humans must maintain control of our actions and consequences. However, it can be challenging to prevent machines from manipulating our free will.

Laws cannot keep up with technology’s high-velocity innovation, causing businesses to define and self-regulate their ethical behavior. Without an official solution for maintaining accountability, this moral desire must compete against existing business priorities. Those priorities will vary significantly between organizations, as defining ‘human-centric systems’ is ambiguous. Moving past those challenges are issues with the fundamental integrity of neural network technologies. Implementing transparency and explainability are open research problems for all but the most trivial systems. After solving those issues, ensuring the use of only inclusive training data requires significant investments into unverifiable results.

These limitations bring the discussion around full circle to the beginning. Artificial intelligent systems are not ethical, evil, or corrupt. They are tools that automate everyday tasks and lower the barrier to entry. Users of that tool need to be cognizant of what these predictions mean and how they influence decisions. However, that is not the same thing as delegating control with impunity. After all, would you trust big business to do the right thing in a world that lacks accountability and legal enforcement?

# References

Adashi, E., Walters, L., & Menikoff, J. (2018). The Belmont Report at 40: Reckoning With Time. *American Journal of Public Health, 108*(10), 1345-1348. Retrieved from https://search-ebscohost-com.proxy1.ncu.edu/login.aspx?direct=true&db=s3h&AN=131743373&site=eds-live

Alsudais, A., Leroy, G., & Corso, A. (2014). We Know Where You Are Tweeting From. *IEEE International Congress on Big Data* (pp. 594-600). Anchorage, AK: The Institute of Electrical and Electronics Engineers. doi:https://doi-org.proxy1.ncu.edu/10.1109/BigData.Congress.2014.91

Ariely, D. (2009). *Predictably irrational: the hidden forces that shape our decisions.* HarperCollins Publishers.

Asimov, I. (1942). *Runaround.*

Atienza, R. (2018). *Advanced deep learning with Tensorflow 2 and Keras.* Birminghan, UK: Packt Publishing.

Bajcsy, R. (1988). Active perception. *Proceedings of the IEEE, 76*(8), 966-1005. doi:10.1109/5.5968

Ballard, D., & Zhang, R. (2021). The hierarchial evolution in human vision modeling. *Topics in Cognitive Science, 13*(2), 309-328. doi:https://doi.org/10.1111/tops.12527

Banjarey, K., Prakash, S., & Kumar, D. (2021). A survey on human activity recognition using sensors and deep learning methods. *5th International Conference on Computing Methodologies and Communication* (pp. 1610-1617). Erode, India: IEEE. doi:https://doi.org/10.1109/ICCMC51019.2021.9418255

Barua, H., & Mondal, K. (2019). A Comprehensive Survey on Cloud Data Mining (CDM) Frameworks and Algorithms. *CM Computing Surveys. Sep2019, Vol. 52 Issue 5, p1-62. 62p*, 1-62.

Bell, Koren, & Volinsky. (2009). *The BellKor solution to the Netflix Prize.* Retrieved from Netflix Prize: https://netflixprize.com/assets/GrandPrize2009\_BPC\_BellKor.pdf

Bhoopathi, & Rama. (2017). A Novel Framework for Stock Trading Analysis Using Casual Relationship Mining. *2017 Third International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB)* (pp. 136-141). AEEICB.

Brown, T., Mann, B., & Ryder, N. (2020, May 28). *Language Models are Few-Shot Learners*. (OpenAI) Retrieved from arXiv: https://arxiv.org/abs/2005.14165

Bryar, C., & Carr, B. (2021). *Working Backwards: Insights, Stories, and Secrets from Inside Amazon.*

Buchanan, B. (2005). A very brief history of artifical intelligence. *AI Magazine, 26*(4), 53-60. Retrieved from ttps://search-ebscohost-com.proxy1.ncu.edu/login.aspx?direct=true&db=ofs&AN=501189619&site=eds-live

Burr, V. (2015). *Social Constructionism.* Routledge.

Carver, S. (2007). Feedback Loop. *Sage Reference: Encyclopedia of Social Psychology*.

CDC. (2016, July 6). *Motor Vehicle Crash Deaths*. Retrieved from Centers for Disease Control and Prevention: https://www.cdc.gov/vitalsigns/motor-vehicle-safety/index.html

Chen, B., & Curtmola, R. (2017). Remote data integrity checking with server-side repair. *Journal of Computer Security 25*, 537-584.

Cheng, Y., Wang, D., Zhou, P., & Zhang, T. (2018, January). Model compression and acceleration for deep neural networks. *IEEE Signal Processing Magazine, 35*(1), 126-136. doi:https://doi-org.proxy1.ncu.edu/10.1109/MSP.2017.2765695

Cohen, B. (2013, November 14). *The 10 smartest cities In North America*. Retrieved from Fast Company: https://www.fastcompany.com/3021592/the-10-smartest-cities-in-north-america

Darwin, C. (1859). *On the origin of species.*

Das, S. (2019). The future of fintech. *Financial Management 2019 Volume 48 pg., 48*(4), 981–1007. doi:10.1111/fima.12297

de Waal, D., & du Toit, J. (2011). Automation of generalized additive neural networks for predictive data mining. *Applied Artificial Intelligence, 25*(5), 380-425. doi:10.1080/08839514.2011.570156

Dickson, A., Emad, H., & Adu-Agyum, J. (2018). Theoetical and conceptual framework: mandatory ingredients of quality research. *International Journal of Scientific Research, 7*, 438-441. Retrieved from https://www.researchgate.net/publication/322204158\_THEORETICAL\_AND\_CONCEPTUAL\_FRAMEWORK\_MANDATORY\_INGREDIENTS\_OF\_A\_QUALITY\_RESEARCH

Douglas, H., & Elliott, K. (2022). Addressing the reproducibility crisis: a response to Hudson. *Journal for General Philosophy of Science*, 1-9. doi:https://doi.org/10.1007/s10838-022-09606-5

Edureka. (2016). *What is Apriori Algorithm*. Retrieved from YouTube: https://youtu.be/guVvtZ7ZClw

Edureka. (2018, October 16). *Natural language processing in 10 minutes*. Retrieved from YouTube: https://www.youtube.com/watch?v=5ctbvkAMQO4

Fedus, W., Zoph, B., & Shazeer, N. (2022). Switch Transformers: scaling to trillion parameter models. (A. Clark, Ed.) *Journal of Machine Learning, 23*, 1-40. doi:https://jmlr.org/papers/v23/21-0998.html

Fonskea, & Liyanage. (2008). A Data mining algorithm to analyse stock market data using lagged correlation. *4th International Conference on Information and Automation for Sustainability* (pp. 163-166). IEEE. doi:10.1109/ICIAFS.2008.4783968

Ford, D. (2012, March 18). *As Cars Are Kept Longer, 200,000 Is New 100,000*. Retrieved from The New York Times: https://www.nytimes.com/2012/03/18/automobiles/as-cars-are-kept-longer-200000-is-new-100000.html

Fridman, L. (2017, January 16). *MIT 6.S094: Introduction to Deep Learning and Self-Driving Cars*. Retrieved from YouTube: https://www.youtube.com/watch?v=1L0TKZQcUtA&feature=youtu.be

Fridman, L. (2020, January). *Deep Learning State of the Art*. (Massachusetts Institute of Technology (MIT)) Retrieved from YouTube: https://youtu.be/0VH1Lim8gL8

Frolov, S. (2021). Quantum computing’s reproducibility crisis: Majorana fermions. *Nature: International Weekly Journal of Science, 592*(7854), 350-352. doi:https://doi.org/10.1038/d41586-021-00954-8

Fu, Z. (2019). An introduction of deep learning based word representation applied to natural language processing. *International Conference on Machine Learning, Big Data and Business Intelligence*, (pp. 92-104). doi:https://doi-org.proxy1.ncu.edu/10.1109/MLBDBI48998.2019.00025

Gergen, K. (2010). *Social Constructionist Ideas, Theory and Practice*. (The Taos Institute) Retrieved from Vimeo: https://vimeo.com/15676699

Gu, F., Chung, M., Chignell, M., Valaee, S., B, Z., & Liu, X. (2021). A survey on deep learning for human activity recognition. *ACM Computing Surveys, 8*, 1-34. doi:https://doi.org/10.1145/3472290

Hevner, A., March, S., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS Quarterly, 28*(1), 75-105. doi:10.2307/25148625

Hole, H., & Ahmad, S. (2019). Biologically driven artificial intelligence. *Computer, 52*(8), 72-75. doi:10.1109/MC.2019.2917455

Huang, M., Rust, R., & Maksimovic, V. (2019). The feeling economy: managing in the next generation of artificial intelligence. *California Management Review, 61*(4), 43-65. doi:https://doi-org.proxy1.ncu.edu/10.1177/0008125619863436

Hudson, R. (2021). Should we strive to make science bias-free? A phiolsophical assessment of the reproducibility crisis. *Journal for General Philosophy of Science, 52*(3), 389-405. doi:https://doi.org/10.1007/s10838-020-09548-w

Iivari, J. (2020). A critical look at theories in design science research. *Journal of the Association for Information Systems, 21*(3), 502-519. doi:doi:10.17705/1jais.00610

Jackson, B., & Rege, M. (2019). Machine learning for classification of economic recessions. *IEEE 20th International Conference on Information Reuse and Integration for Data Science* (pp. 31-38). Los Angeles, CA, USA: Institute of Electrical and Electronics Engineers. doi:10.1109/IRI.2019.00019

Jaisswal, A., & Naik, A. (2021). Effect of Hyperparameters on Backpropagation. *Pune Section International Conference* (pp. 1-5). IEEE. doi:10.1109/PuneCon52575.2021.9686489

Kahn Academy. (2014). *Origin of Markov Chain*. Retrieved from Kahn Academy: https://www.khanacademy.org/computing/computer-science/informationtheory/moderninfotheory/v/markov\_chains

Kane, T. (2019, March). Artificial Intelligence in Politics: Establishing Ethics. *Technology and Society Magazine, 38*(1), 72-80. doi:10.1109/MTS.2019.2894474

Keller, J., Liu, D., & Fogel, D. (2016). *Fundamentals of Computational Intelligence.* John Wiley & Sons.

Kilgallon, S., De la Rosa, L., & Cavazos, J. (2017). Improving the effectiveness and efficiency of dynamic malware analysis with machine learning. *Resilience Week* (pp. 30-36). Wilmington, Delaware: IEEE. doi:10.1109/RWEEK.2017.8088644

Kim, K., & Cho, S. (2008). Evolutionary ensemble of diverse artificial neural networks using speciation. *Neurocomputing, 71*(7-9), 1604-1618. doi:https://doi.org/10.1016/j.neucom.2007.04.008

Krizhevshy, A., Sutskever, I., & Hinton, G. (2012). ImageNet classification with deep convolutional neural networks. *25th International Conference on Neural Information Processing Systems* (pp. 1097-1105). Red Hook, NY: ACM. doi:https://dl.acm.org/doi/10.5555/2999134.2999257

Langer, M., He, H. Z., Rahayu, W., & Xue, Y. (2020). Distributed training of deep learning models: a taxonomic perspective. *Transactions on Parallel and Distributed Systems, 31*(12), 2802-2818. doi:https://doi.org/10.1109/TPDS.2020.3003307

Langston, A. (2022, March 14). *In a historic milestone, Azure Quantum demonstrates formerly elusive physics needed to build scalable topological qubits*. Retrieved from Microsoft Innovation Stories: https://news.microsoft.com/innovation-stories/azure-quantum-majorana-topological-qubit/

Lee, J., & Yoo, H. (2021). An Overview of Energy-Efficient Hardware Accelerators for On-Device Deep-Neural-Network Training. *Open Journal of the Solid-State Circuits Society, 1*, 115-128. doi:https://doi.org/10.1109/OJSSCS.2021.3119554

LeiosOS. (2017). *How algorithms evolve*. Retrieved from YouTube: https://www.youtube.com/watch?v=qiKW1qX97qA

Li, D., Zhao, D., Zhang, Q., & Chen, Y. (2019, May). Reinforcement Learning and Deep Learning Based Lateral Control for Autonomous Driving. *IEEE Computational Intelligence Magazine May, 14*(2), 83-98. doi:10.1109/MCI.2019.2901089

Lui, H., K, S., Fernando, C., & Kavukcuoglu, K. (2018). Hierarchical representations for efficient architecture search. *International Conference on Learning Representations* (pp. 1-13). Vancouver, WA: Carnegie Mellon University. doi:https://doi.org/10.48550/arXiv.1711.00436

Lukac, D., Milic, M., & Nikolic, J. (2018). From artificial intelligence to augmented age an overview. *Zooming Innovation in Consumer Technologies Conference*, (pp. 100-103). doi:https://doi-org.proxy1.ncu.edu/10.1109/ZINC.2018.8448793

Makarenko, O., & Osaulenko, V. (2018). Application of cellular automates in some models of artificial intelligence. *IEEE First International Conference on System Analysis & Intelligent Computing* (pp. 1-4). Kyiv, Kyiv City, Ukraine: Institute of Electrical and Electronics Engineers. doi:10.1109/SAIC.2018.8516837

Marr, D. (n.d.). *Vision a computational investigation into the human representation and processing of visual information.*

Mason, G. (2020, March 1). *Shakespeare's plays*. Retrieved from OpenSource Shakespeare: https://www.opensourceshakespeare.org/views/plays/plays.php

Mejia, J., Quintero, D., & Builes, J. (2017). Knowledge-based model to support decision-making when choosing between two association data mining techniques. *Revista Lasallista de Investigación, 14*(2), 41-50. doi:https://doi.org/10.22507/rli.v14n2a4

Meta AI. (n.a.). *Tanh Activation*. Retrieved from Papers with Code: https://paperswithcode.com/method/tanh-activation

Mirjalili, A. (2018, October 4). *How the Ant Colony Optimization algorithm works*. Retrieved from YouTube: https://www.youtube.com/watch?v=783ZtAF4j5g

Miyakawa, T. (2020). No raw data, no science: another possible source of the reproducibility crisis. *Molecular Brain, 13*(1). doi:https://doi.org/10.1186/s13041-020-0552-2

Ng, A. (2016). *Machine Learning*. Retrieved from Coursera: https://www.coursera.org/learn/machine-learning

Nugroho, H., Harmanto, D., & Al-Absi, H. (2018). On the Development of Smart Home Care: Application of Deep Learning for Pain Detection. *Biomedical Engineering and Sciences* (pp. 612-616). IEEE. doi:10.1109/IECBES.2018.8626710

Office of the Secretary. (1979). *The Belmont Report*. Retrieved from U.S. Department of Health and Human Services: https://www.hhs.gov/ohrp/sites/default/files/the-belmont-report-508c\_FINAL.pdf

Owen, C. (2017, November 8). *A Theorical Hands-on introduction to Fouculdian Discourse Analysis*. Retrieved from YouTube: https://www.youtube.com/watch?v=6I6b3ePAZ5M

Oxford. (2022). *Occam's razor*. Retrieved from Lexico: https://www.lexico.com/en/definition/occam's\_razor

Ozier, O. (2021). Replication redux: the reproducibility crisis and the case of deworming. *World Bank Research Observer, 36*(1), 101-130. doi:https://doi.org/10.1093/wbro/lkaa005

Parker, R. (1993). Threats to the validity of research. *Rehabilitation Counseling Bulletin, 36*(3), 130-138. Retrieved from https://search-ebscohost-com.proxy1.ncu.edu/login.aspx?direct=true&db=eric&AN=EJ458938&site=eds-live

Piirainen, K., & Gonzalez, R. (2013). Constructive Synergy in Design Science Research: A Comparative Analysis of Design Science Research and the Constructive Research Approach. *Liiketaloudellinen Aikakauskirja, 3*(4), 206-234. Retrieved from https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,shib&db=bth&AN=95116694&site=eds-live

Prapas, I., Derakshan, B., Mahirji, A., & Markl, V. (2021). Continuous Training and Deployment of Deep Learning Models. *Datenbank-Spektrum, 21*(3), 203-213. doi:https://doi.org/10.1007/s13222-021-00386-8

Qiu, L., Wang, Y., & Rubin, J. (2018). Analyzing the Analyzers: FlowDroid/IccTA, AmanDroid, and. *ISSTA’18, July 16–21, 2018, Amsterdam, Netherlands*.

Qiu, L., Wang, Y., & Rubin, J. (2018). Analyzing the analyzers: FlowDroid/IccTA, AmanDroid, and DroidSafe. *27th ACM SIGSOFT International Symposium on Software Testing and Analysis* (pp. 176-186). Amsterdam, Netherlands: Association for Computing Machinery. doi:https://doi.org/10.1145/3213846.3213873

Right Vision Media. (2021, March 5). Global data center accelerators market trajectory & analytics 2020-2027. *Gale General OneFile*, NA. Retrieved from https://ink.gale.com/apps/doc/A653973707/ITOF

Rivera-Landos, E., Khomh, F., & Nikanjam, A. (2021). The challenge of reproducible ML. *International Conference on Software Quality, Reliability and Security (QRS).* *21*, pp. 1079-1088. Hainan Island, China: IEEE. doi:10.1109/QRS54544.2021.00116

Roberts, C. (2021). The Ethics of Research That May Disadvantage Others. *Ethics & Human Research, 43*(1), 2-16. doi:10.1002/eahr.500074

Rosebrock, A. (2019, February 11). *Fashion MNIST with Keras and Deep Learning*. Retrieved from PyImageSearch: https://www.pyimagesearch.com/2019/02/11/fashion-mnist-with-keras-and-deep-learning/

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review, 65*(6), 386-408. doi:10.1037/h0042519

Sethi, T., & Kantardzic, M. (2018). Data driven exploratory attacks on black box classifiers in adversarial. *Neurocomputing, 289*, 129-143. doi:10.1016/j.neucom.2018.02.007

Silvestrini, R. P., & Sammito, G. (2012). Design of Experiments for Information Technology Systems. *Defense AT&L, 41*(5), 30-35. Retrieved from https://search-ebscohost-com.proxy1.ncu.edu/login.aspx?direct=true&db=bth&AN=80409129&site=eds-live

Smedley, R. (2019, December 4). *Rob Smedley From Formula 1 Talks About Using AWS to Improve the Fan Experience*. Retrieved from YouTube: https://youtu.be/eBX7lPk5qmA

Snee, R. (2015). Practical approach to data mining. *Quality Engineering, 27*, 477-487. doi:10.1080/08982112.2015.1065322

Sonmez et al. (2018). Anomaly Detection Using Data Mining Methods in IT Systems: A Decision Support Application. *Sakarya University Journal of Science, 22(4)*, 1109-1123.

Starmer, J. (2017). *What is Principal Component Analysis*. Retrieved from YouTube: https://www.youtube.com/watch?v=HMOI\_lkzW08

Tan, Z. (2021). Ethics Events and Conditions of Possibility. *Business Ethics Quarterly, 31*(1), 106-137. Retrieved from https://search-ebscohost-com.proxy1.ncu.edu/login.aspx?direct=true&db=edb&AN=147839336&site=eds-live

Thomas, R. (2019). The new era of NLP. *Scientific Computing with Python.* Austin, Texas: SciPy. Retrieved from https://youtu.be/KChtdexd5Jo

Ting, W., Chun-Yang, C., Di, G., Xiao-ming, T., & Heng, W. (2014). Clock Synchronization in Wireless Sensor Networks: A New Model and Analysis Approach Based on Networked Control Perspective. *Mathematical Problems in Engineering Volume 2014, Article ID 731980*, 1-19.

Tun, S., Madanian, S., & Mirza, F. (2021). Internet of things (IoT) applications for elderly care: a reflective review. *Aging Clinical & Experimental Research, 33*(4), 855-867. doi:10.1007/s40520-020-01545-9

Ünal, H., & Başçiftçi, F. (2021). Evolutionary design of neural network architectures: a review of three decades of research. *Artif Intell Rev, 55*, 1723-1802. doi:https://doi.org/10.1007/s10462-021-10049-5

Upchurch, M. (2018). Robots and AI at work: the propects for singularity. *New Technology, 33*(3), 205-218. doi:10.1111/ntwe.12124

Vosshall, P. (2018, November 27). *How AWS Minimizes the Blast Radius of Failures*. Retrieved from YouTube: https://youtu.be/swQbA4zub20

Waal, d., & Toit, d. (2011, May/June). Automation of generalized additive neural networks for predictive data mining. *Applied Artificial Intelligence, 25*(5), 380-425. doi:10.1080/08839514.2011.570156

Wilensky, U. (2014). *BeeSmart hive finding*. Retrieved from Netlogo: https://ccl.northwestern.edu/netlogo/models/BeeSmartHiveFinding

Zeller, A. (2014). *What makes useful research in software engineering*. Retrieved from YouTube: https://youtu.be/4MbixFVWwck

Zhao, W. (2014). *Building Dependable Distributed Systems : Building Dependable Distributed Systems.* John Wiley & Sons, Incorporated.