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.*

## 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 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.

## Incorporating Markov Chains

A core challenge to applying basic statistics to real-world data is the assumption 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 1: 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 2 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 signals that collaborate 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 3 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 4: Fashion MNIST

Diagram, schematic

Description automatically generated

### Observations

The first and most critical step in any data mining exercise is to determine the question and discover 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 need to 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 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 creates 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 any 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

## 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)

## Computer vision and autonomous driving

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

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 signal 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 human activity recognition (HAR) 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

## How does ML model training work

*Include a description of the problem and briefly describe the approaches.*

### Hardware acceleration

### Continuous Learning

### Optimization Patterns

## 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 (GA)

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 (CA) 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 CA 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

# References

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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/

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

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

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

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