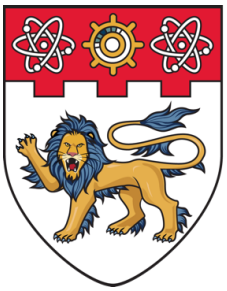

NANYANG TECHNOLOGICAL UNIVERSITY
SINGAPORE

The Technologies of Artificial Intelligence (AI)

Nils Backlund
Oscar Stommendal

N2502406B
N2402294F

SCHOOL OF PHYSICAL AND MATHEMATICAL SCIENCES

PH4418 Physics In the Industry

Assignment Report

April, 2025

Contents

Introduction

Stating that Artificial Intelligence (AI) has seized a firm grip of our world is definitely not an underestimation anymore. Some find this development more intimidating than exciting, and many experts, including OpenAI CEO Sam Altman, have warned governments about this rapid progress [1]. Nevertheless, the fact still stands – that AI now is a part of our daily lives to the highest degree. Nobody missed the outburst of the NVIDIA stock, which as of now has the 3rd largest market cap in the world, or the daily usage of ChatGPT, which has become a habit in many people's lives [2] [3].

The idea of intelligent life in non-human forms goes back far more than one can imagine. In fact, there are traces of myths and stories about this from ancient times [4]. However, one of the first recorded mentions of AI as we know it today is from the classic "Gulliver's Travels", written 1726 by Jonathan Swift. He introduces "The Engine", a machine used by scientists in the fictional stage to generate new information and ideas.

From that point, it would take more than two centuries until Swift's fictive creation received a name, after Alan Turing posed the question "Can machines think" in 1950 [5]. The term Artificial Intelligence was introduced five years later, which later gave birth to the research field [4]. AI later saw substantial development, beginning with early neural networks like the Perceptron (1957) and the introduction of Lisp (1958), which became the dominant AI programming language. After some setbacks in the 70's and 80's due to reduced funding, the resurgence of AI came with breakthroughs like backpropagation for neural networks (1986) and increased interest in machine learning. By the late 1990s, AI research had laid the groundwork for modern deep learning and other applications used today. Since then, AI has advanced rapidly, with breakthroughs in deep learning, generative models (like ChatGPT), and autonomous systems.

This project will cover a rather broad spectrum within modern AI, however with the main focus on the technical parts. We will start by providing the backbone of AI – the prerequisites that make it possible, with a special emphasis on semiconductors. After that follows a rather heavy theoretical part on the technology behind AI, starting with an overview of what AI actually is, and later covering the simplest neural networks to more advanced types of AI. In the following chapter, we will move to the practical part, where applications of AI will be discussed, both today and in the past. Finally, we will provide an outlook for the future of AI. However, in order to set the foundation, we will firstly provide a brief introduction to what AI actually is, and how we characterize it as of today.

What is AI?

Today, we of course have a good understanding of AI and what it is, but we lack a general definition. However, most larger organizations and educational institutions agree on the basic idea of AI, so a general definition could probably be instituted rather easily. For instance, IBM provides a great description of AI [6]:

> "*Artificial intelligence (AI) is technology that enables computers and machines to simulate human learning, comprehension, problem solving, decision making, creativity and autonomy.*"

Moving one step closer, AI can basically be divided into types – Machine Learning (ML), Deep Learning and the new Generative AI. These can be seen as subcategories to AI as a whole, where AI provides the general idea, and the others are applications of this. As mentioned, this report will cover AI as a whole, i.e. all of these subcategories, providing the technologies behind them, and their respective applications.

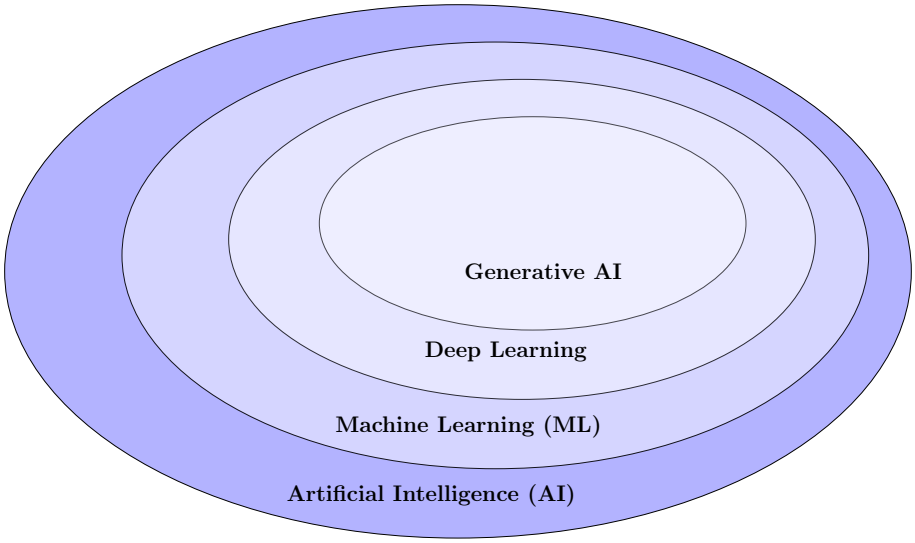

Figure 1: An overview of Artificial Intelligence (AI), with the three largest subcategories: Machine Learning (ML), Deep Learning and Generative AI.

Chapter 1

The Recipe – Foundations of AI

Just as a car needs its engine, and the marathon runner needs water, Artificial Intelligence also has prerequisites to function. Generally, these are quite simple mathematical and technical concepts that we have known for a long time. However, as with everything there is, one must know how to put the different parts together correctly for the machinery to start working. This chapter covers the basic foundations of AI, starting with a short introduction to the mathematical concepts used within AI, followed by a heavier part on the technical ingredients in the AI recipe.

1.1 Mathematical Foundations

The mathematical concepts used in AI have been known and used during a long period before AI entered the picture, and are in general very basic. The most important ones are linear algebra, calculus and probability theory, used in different ways depending on the type of AI [7]. For instance, linear algebra can be used to represent data in a matrix form, which is essential for many machine learning algorithms. Matrix multiplication is another important, yet again simple concept, used to combine data sets and perform operations on them. This is for example crucial in deep learning, which often involves large data sets. This will be more clear in Chapter 2. Calculus can be used to optimize the parameters of such algorithms, while probability theory, e.g. Bayesian methods, can be used to model uncertainty and make predictions. However, as this project focuses on the technical aspects of AI, we will not go into detail on these mathematical concepts.

1.2 Technical Foundations

The technical foundations of AI are the building blocks within hardware and software technologies that enable development and usage of AI systems. Simply put, this includes the semiconductor industry, providing the computing power, and the algorithms and frameworks that allow for the implementation of AI models. Of course, we could dig into *all* the different prerequisites for AI, from memory and storage all the way to the actual computers. However, as this would result in lots of information without actual value for the project, we will only cover the most important ones, i.e. the semiconductor industry and the software technologies used in AI.

1.2.1 Semiconductor Industry

The semiconductor industry plays a crucial role of the technology behind AI, as it provides the hardware needed to run AI algorithms [8]. In order to match the almost daily breakthroughs and increased complexity of AI, the demand for computing power in the shape of chips has increased significantly. Compared to the regular Central Processing Unit (CPU), AI chips carries several advantages, such as speed, performance and energy efficiency, with the most important one being the ability to perform several operations in parallel. Moreover, they are also more flexible than their predecessors, meaning that they can be manufactured in different sizes and shapes, depending on the application.

As of today, there are many types of AI chips, including Graphics Processing Units (GPUs), Field Programmable Gate Arrays (FPGAs), Application-Specific Integrated Circuits (ASICs), and Neural Processing Units (NPUs) [8]. Each of these has its own strengths and weaknesses, making them suitable for different types of applications. For example, GPUs are well-suited for training AI models, while FPGAs has the benefit of being customized for specific applications.

However, one of the shared characteristics of these chips is that they are based on the same technology – semiconductors. In short, semiconductors are materials that have electrical conductivity between that of conductors and insulators, enabling usage in a wide range of electronic devices [9]. They are typically made from silicon, which is relatively common and cheap, making it an ideal material for the industry. Of course, the semiconductor industry is also evolving, with new technologies being developed. For example, the use of FinFET transistors has allowed for smaller and more efficient chips [10]. These advancements are essential for the continued growth of AI, as they enable researchers to build larger and more complex models that can handle more data.

A great example of the rapid development of the semiconductor industry is Moore's Law, which has been a driving force behind the growth of AI and computing in general. Moore's Law states that the number of transistors on a microchip doubles approximately every two years, leading to an exponential increase in computing power [10] [11]. This has allowed for the development of increasingly complex AI models and algorithms, as well as the ability to process larger data sets.

1.2.2 Software Technologies

In its most basic form, the software used in AI are the tools that allow developers to build, train, and utilize AI models. In an everyday context, these are essentially programming languages and frameworks that provide the necessary libraries and functions to implement AI algorithms. The first language used for AI was Lisp, which was developed in the late 1950s and is still used today [4]. However, today, the playing field is much larger, with many options available. For instance, the Google team has developed an open source framework called TensorFlow, which is widely used for building and training machine learning models [12]. This is also compatible and can be used together with many common programming languages, such as Python and C++.

Chapter 2

The Framework – Technologies of AI

How often do we actually think about the technology that we use in our daily life? New inventions and technologies have become standard in today's society. New iPhones or cars does not turn lots of heads anymore, and we see 7- and 8-year-olds using cutting-edge technologies that people 50 years ago could not imagine. To some degree, this results in a world where nobody except the inventors actually know the functionality behind those technologies. People walk around the streets with noise-canceling headphones, drives their cars to work and uses the World Wide Web daily, without necessarily thinking about what actually makes this possible. And who can blame them – in the rushing life of today, we hardly have the time, nor interest to dig into this. However, this chapter provides the foundations to what makes Artificial Intelligence work the way it does. This way, you will essentially know what happens the next time you ask ChatGPT for help.

2.1 Machine Learning and Deep Learning

The concept of *machine learning* can be found directly underneath AI in the AI hierarchy, as shown in Figure 1. Within machine learning, we have a number of different models and algorithms used to make predictions based on input data [6]. The predictions are made by *training* the model on a given data set, which can be done using a number of different methods. We will cover the most common ones used today, i.e. supervised and unsupervised learning, as well as reinforcement learning. Today, there are many different machine learning algorithms, such as linear regression, decision trees, and support vector machines. However, we will only cover the most common model used today – the *neural network*. From neural networks, the step is not too far *deep learning* is a subcategory of machine learning that uses more complex types of neural networks, in order to solve more complex problems. Nevertheless, we will start with a brief introduction to the neural network, which is the backbone of deep learning.

Neural networks are based on neurons from biological organisms, adapted and simplified to fit into a computer model [6]. Instead of having a synaptic cleft with different connections, we have numerical weights, and instead of biological neurons we have artificial neurons, taking inputs from each weight and applying a function to interpret the input weights. This system is scalable, and the original models of Boltzmann machine linear solvers have been expanded to feed forward networks, convolutional networks and recurrent networks.

With the development of transformers, it has also been possible to expand these models to language and visualization.

The simplest of networks consists of a single layer of input neurons and one output layer, as illustrated in Figure 2.1. A single neuron corresponds to a non-linear function

$$f(\vec{x};\vec{w}) = \Phi \circ g = \Phi(g(\vec{x};\vec{w})),$$

that takes input data $\vec{x}$ and weight terms $\vec{w}$ and produces the output $\vec{y}$ [13]. The function Φ is known as the *activation function* and is in general non-linear, whereas the function g in general is linear (though its structure can differ). This approach of combining a linear and non-linear function in the neuron increases the modeling power, while also simplifying analytical results in the neural network. There are a number of activation functions used for different purposes. Today, the most used is the so called ReLU function. Other similar functions, like the arctan and Sigmoid that are linear around zero, but tapers off, eliminating too large values, are widely used.

A simple neural network can easily become more interesting and capable, yet more complicated, by adding more layers. This introduces non-linearity and therefore enables the model to solve more problems. An illustration of this is made in Figure 2.1. This is an example of a deep neural network (DNN), which is a type of machine learning model that consists of multiple layers of neurons. Each layer is connected to the next one, and the output from one layer becomes the input for the next. The DNN can learn complex patterns in data by adjusting the weights and biases of the neurons during training. The more layers and neurons in the network, the more complex patterns it can learn.

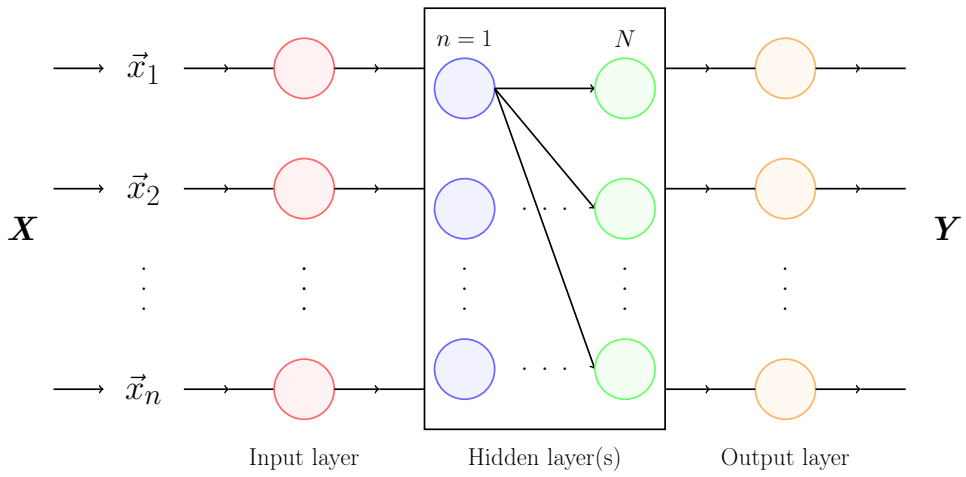

Figure 2.1: Schematic view of a deep neural network where each circle represents a single neuron. An input signal $\boldsymbol{X}$ enters the model in the input layer, proceeds through the N hidden layers, and exits from the output layer. This process produces an output $\boldsymbol{Y}$, which can be interpreted as a prediction based on the input data.

We can also make neurons feed back to themselves or to earlier parts of the network. Networks that utilize this principle is called recurrent networks (RNN). This makes them useful when receiving new information that the network should take into account, e.g. if a time series is being analyzed. These networks were also used in the translation algorithms created by Google for Google Translate [14].

When analyzing larger images it a lot of inputs parameters are usually used, for example a HD image will have at least $1\,280 \cdot 720 \cdot 3 = 2\,764\,800$ parameters. This would make the networks described above extremely large, which will make them slow and hard to train. In images, pixels nearby are often related, which makes it possible to analyze how nearby images relate to each other in multiple steps before running it through a fully connected network. This is done by using a kernel cycling through the inputs of the image, and taking for example the maximum or mean of the numeric values of the kernel, which is called convolutional networks (CNN).

The performance of a neural network is measured using a loss function $L(\hat{y})$, relating the prediction data to the original data. For this, the mean square error can be used,

$$MSE = L(\hat{y}) = \frac{1}{p}\sum_{k=1}^{p}(y_k - \hat{y}_k)^2, \tag{2.1}$$

where y is the original data, $\hat{y}$ is the predicted data and p is the number of data points. This can be used to measure how well the model fits the data, and can be used to optimize the model during training.

2.1.1 Model Training

The networks are usually trained to optimize the weights by minimizing the loss function in the network. This is done iteratively throughout a number of epochs and each data points is used at least ones in one epoch. Data can also be used in different sized batches during training to promote a more stochastic training. The weights are updated by adding the downward gradient from the lost function,

$$w_{ij} \leftarrow w_{ij} - \alpha\frac{\partial L}{\partial w_{ij}} \tag{2.2}$$

where α is the learning rate. The same can also be done for the biases. The derivation is straight-forward and with the MSE as the loss function it will for one point be $\frac{1}{2}(y-\hat{y})^2$. Thus, the change for the weight will be

$$\Delta w_{ij} = \alpha(y-\hat{y})\frac{\partial L}{\partial \hat{y}} = \alpha(y-\hat{y})\Phi'(x)x \tag{2.3}$$

where x is the output from the layer before. The same can also be done for the biases in the function. This can then be repeated working backward through the network, i.e. backpropagation. This is also used for CNNs and RNNs. Transformers can also be accounted for here by unfolding the time dimension from the network.

A problem that occurs when training deep networks is that they depend on all later parts of the network, and there will a recurrent multiplication based on all earlier layers. Repeated multiplication with numbers above or below 1 will lead to a exploding or vanishing gradient. This will make the network hard to train, and to take this into account it is preferable to for example normalize the data before training, which will make the network more stable.

Another problem developing when training a network is overfitting, where the network learns to perfectly predict the training data. To combat this the network is split into 3

independent parts. One training set, used for training the network, one validation set used to test the performance in each epoch, and a test set to test final performance. The training and validation set can also be combined via e.g. k-fold cross validation which utilizes the data more effectively. With the use of the training set we can see where the model starts overfitting and end the training when this starts, called early stopping.

Another way to avoid overfitting is to let some neurons drop out during parts of the training process. This creates redundant representations and prevents co-adaptation of neurons. Weight decay can also be used to encourage smaller weights and deduce overfitting. We can also speed up the training using methods like a variable learning rate where it gets smaller as the optimal solution gets closer. Most of the mentioned methods can also be combined to train the network better.

2.1.2 Supervised and Unsupervised Learning

The "learning" part in machine learning can be done in different ways, depending on the data and model that are used. The most common types are supervised and unsupervised learning. In supervised learning, the model is trained on a labeled data set, meaning that we know the correct output from a certain input data [15]. This allows the model to test its accuracy and by that learn from its mistakes. As mentioned before, this is often done using a loss function, e.g. the mean square error, which measures how well the model fits the data. This type of learning is often used for classification and regression tasks, where the goal is to predict a certain output based on the input data. For instance in weather forecasting or pricing predictions.

In unsupervised learning, the model is trained on an unlabeled data set, meaning that there is no correct output [15]. This allows the model to learn patterns and relationships in the data without being told what to look for. Unsupervised learning is often used for clustering and dimensionality reduction, where the goal is to group similar data points together or reduce the number of features in the data. This can be done using algorithms like k-means clustering or principal component analysis (PCA). Unsupervised learning models for example has applications in image recognition and recommendation engines, where the "correct" output is not known or necessary.

One can also combine these two methods into one. The result is then called semi-supervised learning [15]. This is a hybrid approach that uses both labeled and unlabeled data to train the model. The idea is to use the labeled data to guide the learning process, while also allowing the model to learn from the unlabeled data. This can be useful when there is a large amount of data available, for instance in medical imaging.

2.1.3 Reinforcement Learning

In constrant to both supervised and unsupervised learning, reinforcement learning (RL) is a type of machine learning where the model learns by interacting with its environment [16]. The model receives feedback in the form of rewards or penalties based on its actions, and it learns to maximize the rewards over time. The actual learning can either be done online, i.e. during the environment interaction, or offline, i.e. after the interaction has been completed. RL is often used in robotics and game playing, where the goal is to learn a policy that maximizes the expected reward. A board game is a perfect example

to understand this. While playing (many, repeated times), the AI agent will learn which decisions lead to a win, and which lead to a loss. This is done by assigning a value to each action taken, and then updating the values based on the feedback received. The model then learns to take actions that maximize the expected reward over time. In fact, reinforcement learning has been used to train AI agents to play, and master, games like chess and Go [17].

2.2 Generative AI

Just as deep learning is a subset of machine learning, *generative AI* is a subset of deep learning [6]. The basic idea behind this is to use deep learning models to generate new data as response to the user's input. Models like this has been used for a long time. However, during the last decade they have improved significantly, and are today able to analyze and generate much more complex data. This development has been made possible due to essentially three diffrent deep learning models – Variational Autoencoders (VAEs), Diffusion models and Transformers. Although, we will only cover the latter one in this project, as it is the most common one used today.

2.2.1 Variational Autoencoders

The first generative AI model to see the world's light was the Variational Autoencoder (VAE) [18]. These are a subset of the larger family of so called autoencoders. An autoencoder is a type of deep learning model that in its most basic form is used to extract the most important information (this is the latent data) in a data set, e.g. a text input. During training, this is done by first compressing the data, essentially simplifying (the encoder) and then reconstructing it (the decoder). The difference among autoencoders lies in the strategy they use for encoding the data.

However, what differs VAEs from regular autoencoders is that they use a probabilistic approach when encoding the data [18]. This essentially means that they, instead of deterministically encode each data point, encode it as a distribution. In other words, instead of mapping the latent data to a single point x, they encode it to a probability distribution $p(x)$. In Bayesian terms, this distribution of the latent variables is called the prior distribution. The goal then is to determine the posterior distribution, $p(x|D)$, where D is the data set [18]. This describes the probability distribution of the latent variables given the data. The VAEs then use a technique called variational inference to approximate the posterior distribution, since the goal is to generate output similar to the input data, and not reconstruct it exactly. In practice, VAEs are used in a wide range of AI applications, mostly within image generation, however also in anomaly detection and to generate new drug molecules.

2.2.2 Diffusion Models

Similarly to VAEs, diffusion models are a type of generative model uses deep learning models, primarily within computer vision [19]. The basic idea for these models is borrowed from physics, and diffusion processes. Such processes essentially describe how particles move from areas of high concentration to areas of low concentration [20]. In the context of AI, the process behind diffusion models can be broken down in three-step schedule:

Forward diffusion → Reverse diffusion → Generation

In the forward diffusion step, the model takes input data, e.g. and image, and progressively add (gaussian) noise until the data is completely random [19]. Mathematically, this is described by a Markov Chain – a kind of stochastic process where the future system state depends only on the current state, and not on the system history. The actual machine learning then happens in the reverse diffusion step, where the model learns how to denoise the data. This is essentially done by predicting the total noise in the data, and remove a fraction of this. During training, the objective function is similar to the one in VAEs, i.e. maximizing the Evidence Lower Bound (ELBO). In this case, however, the loss function for which the ELBO is maximaized is consisting of three different terms. These can in the end be simplified to the mean squared error between the predicted and actual noise. The actual generation is then done by sampling from the learned data distribution and denoising it. The small randomness in each step enables the diffusion model to generate new data that is similar to the training data, rather than identical. As meantioned, diffusion models are mainly used for image generation. Some well-known examples of this include OpenAI's DALL-E and Google's Imagen.

2.2.3 Transformers and Large Language Models

Compared to images, both order and correlation between words matters when generating and analyzing text. For instance, to show the correlation between 10 values, we would need 100 values to represent this, which then scales as n^2. To solve this, we use positional encoding, and the Google Brain team developed the transformer in 2017, which instead scales like $n\log(n)$ [21]. The transformer is based on self attention where important between tokens are shown. This is done by splitting the input data into three parts, a query Q, key K and value V part, where each part is the product between the data and a weight matrix for each part. A scaled dot-product is made between these as

$$\text{SoftMax}\left(\frac{QK^T}{\sqrt{d_k}}\right)\dot{V} \tag{2.4}$$

where d_k is the dimensionality of the query and key. This is usually done in multiple heads to get a more robust result.

To complete the transformer, the input is first passed through a multi-headed attention layer. This output is added to the original input and normalized. This is also done on a separate path for the output it yet has created. The input is past through a fully connected feed forward network (FFN) and once again added and normalized to the values before the FFN. After this step the both paths merges in a attention layer, where the queries and keys come from the input, and the values comes from the outputs. This is added to the vector that became the values and then normalized. This is passed through an FFN, which is linearized and put through the SoftMax function, which results in a likelihood for the next word in the output sequence. This recursion continues until a end of sequence token is produced. A diagram of how the transformers work is shown in Figure 2.2

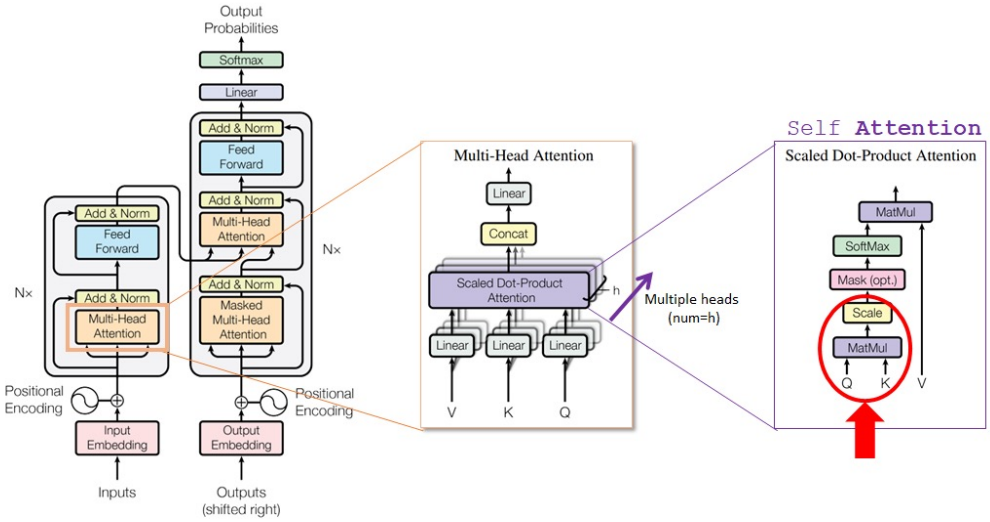

Figure 2.2: The different parts of the transformer, as outlined by the Google Brain team. Figure from [21].

A type, or application of the transformer that has gained a lot of attention during the last few years is the Large Language Model (LLM), for instance ChatGPT and Microsoft Copilot. When texting an LLM, the first issue from a computational perspective is to translate the text into some computer-friendly language. To do this, we translate the words into floating points to perform different types of operations on them (we will come to this). We also know that the position of words matter in a sentence, which we must handle. Furthermore, we have correlation between words in paragraphs. All of these issues are what transformers were invented to do this in an efficient way.

The transformer has set the foundation for LLMs and scaling of compute and training data as well as other models improvements has to the LLMs like ChatGPT and Claud we use today. This is currently the most noticed field and alot of investments is being made into this. For example Microsoft is on track to invest "approximately $80 billion to build out AI-enabled datacenters to train AI models and deploy AI and cloud-based applications around the world" this year [22]. This has lead to the chat interfaces we are so used to see today, and consent improvements in algorithms are being made. For example a recent development made is methods using reinforcement learning to create a separate step before the output to let the model "think".

As described above, there are two main parts of encoding text – replacing words and part of words with numbers, called tokenization, and encoding their position. The easiest approach to tokenization is to add every unique character or word to the input set and translate them into numbers. There are some problems with these approaches, like the tokenized sentence being very large or limited by vocabulary. Therefore, there has been other methods developed, e.g. byte pair encoding. This will help compress the text by replacing the most common sequences of characters with non-used characters and placed into a lookup table [23]. These can thereafter be changed into numbers using Unicode or UFT-8 conversion [24]. There are multiple ways of doing positioning encoding as well, some better than others. The most used one today comes from the paper "*Attention is all you need*". Here, the authors propose a system where the position encoding is of the same dimension d_{model} as the text embedding, and so they can be summed up. The positional encoding used can be formulated as

$$
\begin{aligned}
PE_{(p,\,2i)} &= \sin(p/10\,000^{2i/d_{\text{model}}}) \\
PE_{(p,\,2i+1)} &= \cos(p/10\,000^{2i/d_{\text{model}}})
\end{aligned}
\tag{2.5}
$$

where i is the dimension, p the position [21]. This will be a set of sinusoidal functions where the distance is described by different frequencies, which makes the model more easily learn to attend from different positions [21]. Another way to positional encoding is to use learned positional embeddings, but this might limit positional extrapolation in longer input sequences if this is not taken into account.

Due to the large amount of data that is used to train LLMs, they generally fall under the supervised learning category. However, the approach of reinforcement learning has more recently been applied to LLMs. In this case, they are set to complete a certain problem by using Chain of Thought (CoT), and then their weights are updated based on the quality of the result. Recently, DeepSeek showed that it is possible to train a model to have an effective CoT without having any supervised fine-tuning (SFT) in their DeepSeek-R1-zero model [25]. They used a reward system combining accuracy reward (providing specific answers or solving LeetCode problems) and formatting reward. These two rewards were enough to improve the model with reasoning capabilities. It evolved to re-evaluate its thought, investigating the correctness of the result. The authors describe this as an "aha moment" and states that it "serves as a powerful reminder of the potential of RL to unlock new levels of intelligence in artificial systems".

This open source model and the publication of this paper show huge promise, but there are also some problems. Without any SFT, the models they trained thoughts were not clear, so helping it with few examples of long CoT helped convergence and making the CoT easier to understand [25].

2.2.4 The Alignment Problem

As AI becomes more advanced, the number of advantages and excitement about it obviously increase. However, it also leads to possible challenges and risks. One well-known example is *the alignment problem*, which refers to the difficulty of ensuring that AI systems behave in ways that are aligned with human values [26]. As far-fetched as possible, this issue essentially describes a world where the stories in "Terminator" and "Transformers" are true, and the AI systems we have created are no longer under our control. Hence, as the complexity of AI increases, it becomes crucial to ensure that they are designed in ways that reflect our values. In the context of LLMs, this is particularly important as they could produce harmful or misleading content if they are not properly aligned.

As of today, there is no obvious solution to the alignment problem, and it is highly debated topic among researchers and experts in the field. Many experts, including OpenAI CEO Sam Altman, have expressed concerns about the rapid AI development [1]. However, there are approaches to address this challenge. One approach is to use reinforcement learning to train AI systems to behave in ways that are aligned with human values [26]. This involves providing real human feedback to the system based on its behavior, and using this feedback to adjust its training process. Another approach is AI governance, which is the implementation of standards and guidelines for the development and deployment of AI systems. For instance, IBM has the AI Ethics Council, which ensure that new AI products and services align with the company's values and principles.

Chapter 3

The Applications – Usage of AI

In the last few decades, AI has transitioned from theory and movies to being a large part of modern industries and everyday life. As of today, its applications essentially span across all fields, from healthcare and finance to manufacturing and entertainment, reshaping how tasks are performed and decisions are made. This chapter explores the various ways AI is utilized, highlighting the most influential application areas today.

3.1 Computer Vision

The evolution of CNNs has reshaped the ability for image classification, which transformed the field of computer vision [27]. It has made it possible to use less computing power with much better results compared to earlier image classification algorithms. One large effect of this is the evolution of the computer vision field, giving the network the ability to observe the environment. This is usually done with a mix of censors, like cameras, radars and/or lidars, where the use of sensor fusion can create simultaneous localization and mapping (SLAM). The use cases of this technology are vast and it is deployed in everything from Google Lens to self driving cars and humanoid robots.

3.2 Control systems

Advancements in deep reinforcement learning and model predictive control have significantly enhanced the capabilities of planning and control systems in robotics. These methods allow agents to learn optimal strategies through interaction with their environments, enabling more flexible behavior. With the integration of neural networks, control systems can now approximate complex dynamics and adapt to changes in real-time, making them suitable for uncertain or dynamic settings. This has led to more robust and intelligent motion planning in autonomous drones, self-driving cars, and legged robots. Combined with improvements in computational power and simulation environments, these technologies allow robots to safely train and optimize their behavior in virtual worlds before deployment, accelerating development and deployment across logistics, manufacturing, and exploration.

3.3 Security with AI

There are a number of ways AI can help us develop more secure systems and preventing faults in systems before they even happen. A key advantage of neural networks is there ability to detect small abnormalities accurately in systems. We can see companies already implementing neural networks to detect faults early and an example of this is the company Eneryield that detects faults in the energy system with the help of neural networks [28]. Another example is how we could make the situation for elders in the eldercare market more secure. It would for example be possible to predict falls before they happen with the help these predicting algorithms.

Furthermore an example on how LLMs can be used is to check if e-mails received are hazardous or not. This could prevent fishing attempts and improve the cybersecurity of companies. In the future when AI-agents become stronger we will see them being used to fight cybercrime as well, helping humans at all levels and being able to detect or initiate almost undetectable attacks on other countries or companies [29].

3.4 Other uses of AI

Of course, AI is not limited to the fields mentioned above. As the beginning of this chapter aligned, it is present in most fields in today's society. Below, we will briefly mention some other areas where AI is used today.

- **Customer service and support** Most companies, especially within e-commerce, have some form of customer support. This is nowadays driven via AI, where chatbots and virtual assistants are used to handle customer inquiries and provide support [6].

- **Simulations** With the help of AI it would be easier to create models that mimic reality. An example on where this could be used is in the acoustic, food and optics industry simulating more complex systems with more accuracy. We also see simulations of wether being enhanced with the use of AI and neural networks. The need of physical experiments will still be important to gather more data to train the models, which will lead to even better models used in production.

- **Healthcare** AI is used in a number of different ways in healthcare, from analyzing medical images to predicting patient outcomes [30]. For example, AI algorithms can analyze X-rays and MRIs to detect abnormalities, while machine learning models can predict patient readmissions based on electronic health records.

- **Finance** AI is used in finance for a number of different purposes, including fraud detection, risk assessment, algorithmic trading and many more [31]. For example, machine learning algorithms can analyze transaction data to detect fraudulent activity, while AI models can predict stock prices based on historical data.

- **Manufacturing** AI is used in manufacturing to optimize production processes, improve quality control, and reduce downtime [32]. For example, machine learning algorithms can analyze sensor data from machines to predict when maintenance is needed, while AI models can optimize production schedules based on demand.

Chapter 4

The Next Generation – Future of AI

The current evolution of LLMs is massive, and we can see huge investments being made to build the next generation of models. This leads to even larger data centers to train the models, more resources to develop them, and increase in the awareness around it. There are a lot of companies taking advantage of these investments to create the best models, e.g. OpenAI, Enthropic, DeepSeak and many more. As discussed previously, there have also been huge improvements in algorithm efficiency, leading to even larger scaling [33]. With this rapid development, it is not impossible that we would reach a point where the models reach a level of competence similar to subject-specific humans, often referred to as Artificial General Intelligence (AGI).

This could lead to an even faster era of development, where AI improves itself and reaches an intelligence level that is hard to comprehend. Experts are trying to communicate how this could shape our future and how dangerous it could be if it is not handled correctly. For example, OpenAI CEO Sam Altman expresses great concern about an "AI apocalypse" being a realistic scenario [34].

Of course, this is not the only way forward. If AI alignment is successful, it is not impossible that we would reach an "AI paradise". Superintelligence, the potential form of AI that is beyond human intellect, could help us solve problems of the world and create new purposes for humans [35]. It is also not impossible that our development reaches some kind of limit, and we try to implement the current technology more into our daily lives. Either way, we can not deny that artificial intelligence will be a huge part of our society and that there is an interesting, possibly even frightening, period ahead of us.

Conclusion

In conclusion, AI development sees no end in sight. The rapid evolution of LLMs and the increasing amount of data available for training these models have greatly enhanced the advancements in this domain. From the various applications that we studied, AI has already been applied to healthcare, finance, manufacturing, and entertainment. With such rapid development come challenges and risks associated with AI use, including the alignment problem and possible malicious use. Therefore, one of the principal things is to monitor the development of AI and provide necessary regulation to ensure that it is undertaken in a responsible and ethical manner. The future of AI is uncertain, but one thing is clear – AI will have a significant role in shaping our society. Whether we reach the epoch of superintelligence or continue improving our existing models, the effect of AI on our lives will be heavily noticeable. Henceforth, we must become alert yet active to tackle the challenges and opportunities posed by this swiftly changing arena.

References

[1] C. Vallance, *Artificial intelligence could lead to extinction, experts warn*, Accessed: 2025-03-11, 2023. [Online]. Available: https://www.bbc.com/news/uk-65746524.

[2] Yahoo Finance, *Yahoo finance*, Accessed: 2025-03-11, 2025. [Online]. Available: https://finance.yahoo.com/.

[3] K. Hu, *Chatgpt sets record for fastest-growing user base - analyst note*, Accessed: 2025-03-11, 2023. [Online]. Available: https://www.reuters.com/technology/chatgpt-sets-record-fastest-growing-user-base-analyst-note-2023-02-01/.

[4] T. Mucci, *History of artificial intelligence*, Accessed: 2025-03-11, 2025. [Online]. Available: https://www.ibm.com/think/topics/history-of-artificial-intelligence.

[5] A. M. Turing, "Computing machinery and intelligence," *Mind*, vol. 49, no. 195, pp. 433–460, 1950.

[6] C. Stryker and E. Kavlakoglu, *What is artificial intelligence (ai)?* Accessed: 2025-04-01, 2025. [Online]. Available: https://www.ibm.com/think/topics/artificial-intelligence.

[7] P. Norvig and S. J. Russell, *Artificial Intelligence: A Modern Approach*, 3rd ed. Harlow: Pearson Education Limited, 2016, ISBN: 9781292153971.

[8] M. Flinders and I. Smalley, *What is an ai chip?* Accessed: 2025-04-02, 2024. [Online]. Available: https://www.ibm.com/think/topics/ai-chip.

[9] M. Murphy, *What are semiconductors?* Accessed: 2025-04-02, 2023. [Online]. Available: https://research.ibm.com/blog/what-are-semiconductors.

[10] G. Y. Guanggeng, *Lecture on semiconductors in physics industries*, Lecture, Nanyang Technological University, Singapore, Feb. 2025.

[11] Encyclopædia Britannica, *Moore's law*, Accessed: 2025-04-02, 2025. [Online]. Available: https://www.britannica.com/technology/Moores-law.

[12] Martín Abadi, Ashish Agarwal, Paul Barham, *et al.*, *TensorFlow: Large-scale machine learning on heterogeneous systems*, Software available from tensorflow.org, 2015. [Online]. Available: https://www.tensorflow.org/.

[13] C. C. Aggarwal, *Neural Networks and Deep Learning*, 2nd. Springer, 2023, ISBN: 978-3031296420. [Online]. Available: https://link.springer.com/book/10.1007/978-3-031-29642-0.

[14] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," 2016. arXiv: 1609.08144 [stat.ML]. [Online]. Available: https://arxiv.org/abs/1609.08144.

[15] J. Delua, *Supervised versus unsupervised learning: What's the difference?* Accessed: 2025-04-03, 2021. [Online]. Available: https://www.ibm.com/think/topics/supervised-vs-unsupervised-learning.

[16] J. Murel and E. Kavlakoglu, *What is reinforcement learning?* Accessed: 2025-04-04, 2024. [Online]. Available: https://www.ibm.com/think/topics/reinforcement-learning.

[17] D. Silver, T. Hubert, J. Schrittwieser, *et al.*, *Mastering chess and shogi by self-play with a general reinforcement learning algorithm*, 2017. arXiv: 1712.01815 [cs.AI]. [Online]. Available: https://arxiv.org/abs/1712.01815.

[18] D. Bergmann and C. Stryker, *What is a variational autoencoder?* Accessed: 2025-04-03, 2024. [Online]. Available: https://www.ibm.com/think/topics/variational-autoencoder.

[19] D. Bergmann and C. Stryker, *What are diffusion models?* Accessed: 2025-04-06, 2024. [Online]. Available: https://www.ibm.com/think/topics/diffusion-models.

[20] Encyclopædia Britannica, *Diffusion*, Accessed: 2025-04-06, 2025. [Online]. Available: https://www.britannica.com/science/diffusion.

[21] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, "Attention is all you need," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[22] Microsoft On the Issues, *The golden opportunity for american ai*, Accessed: 2025-03-14, Jan. 2025. [Online]. Available: https://blogs.microsoft.com/on-the-issues/2025/01/03/the-golden-opportunity-for-american-ai/.

[23] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2015. arXiv: 1508.07909 [cs.LG]. [Online]. Available: https://arxiv.org/abs/1508.07909.

[24] The Deep Hub. "All you need to know about tokenization in llms." Accessed: 2025-03-14. (Mar. 2024), [Online]. Available: https://medium.com/thedeephub/all-you-need-to-know-about-tokenization-in-llms-7a801302cf54.

[25] DeepSeek-AI, "DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning," 2025. arXiv: 2501.12948 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2501.12948.

[26] A. Jonker and A. Gomstyn, *What is ai alignment?* Accessed: 2025-04-05, 2024. [Online]. Available: https://www.ibm.com/think/topics/ai-alignment.

[27] IBM, *What is computer vision?* Accessed: 2025-04-20, 2021. [Online]. Available: https://www.ibm.com/think/topics/computer-vision.

[28] Eneryield, *Careers — eneryield*, https://eneryield.com/contact-us/careers, Accessed: 2025-04-01, 2025.

[29] S. Alexander and D. Kokotajlo. "Ai 2027: A deeply researched, month-by-month scenario." (2025), [Online]. Available: https://ai-2027.com (visited on 04/09/2025).

[30] M. North, *6 ways ai is transforming healthcare*, Accessed: 2025-04-20, 2025. [Online]. Available: https://www.weforum.org/stories/2025/03/ai-transforming-global-health/.

[31] M. Finio and A. Downie, *What is ai in finance?* Accessed: 2025-04-20, 2023. [Online]. Available: https://www.ibm.com/think/topics/artificial-intelligence-finance.

[32] M. Finio and A. Downie, *How is ai being used in manufacturing?* Accessed: 2025-04-20, 2024. [Online]. Available: https://www.ibm.com/think/topics/ai-in-manufacturing.

[33] L. Aschenbrenner. "Situational awareness: The decade ahead." Accessed: 2025-03-14. (2024), [Online]. Available: https://situational-awareness.ai.

[34] CNN. "Sam altman has always been a risk taker. now he's taking on ai." Accessed: 2025-03-14. (2023), [Online]. Available: https://edition.cnn.com/2023/10/31/tech/sam-altman-ai-risk-taker/index.html.

[35] T. Mucci and C. Stryker, *What is artificial superintelligence?* Accessed: 2025-04-20, 2023. [Online]. Available: https://www.ibm.com/think/topics/artificial-superintelligence.