- The costs of robots are a major reason for lack of adoption. But innovative companies, like Cobalt Robotics, are using new business models to help out, such as with subscriptions.

- Consumer robots are still in the initial stages, especially compared to industrial robots. But there are some interesting use cases, such as with machines that can be companions for people.

- During the 1950s, science fiction writer Isaac Asimov created the Three Laws of robotics. For the most part, they focused on making sure that the machines would not harm people or society. Even though there are criticisms of Asimov's approach, they are still widely accepted.

- Security has not generally been a problem with robots. But this will likely change—and fast. After all, more robots are connected to the cloud, which allows for the intrusion of viruses and malware.

- The Robot Operating System (ROS) has become a standard for the robotics industry. This middleware helps with planning, simulations, mapping, localization, perception, and prototypes.

- Developing intelligent robots has many challenges because of the need to create physical systems. Although, there are tools to help out, such as by allowing for sophisticated simulations.

Implementation of AI

Moving the Needle for Your Company

In March 2019, a shooter live-streamed on Facebook his brutal killing of 50 people in two mosques in New Zealand. It was viewed about 4,000 times and was not shut off until 29 minutes after the attack.[1] The video was then uploaded to other platforms and was viewed millions of times.

Yes, this was a stark example of how AI can fail in a horrible way.

In a blog post, Facebook's VP of Product Management, Guy Rosen, noted:

> AI systems are based on 'training data,' which means you need many thousands of examples of content in order to train a system that can detect certain types of text, imagery or video. This approach has worked very well for areas such as nudity, terrorist propaganda and also graphic violence where there is a large number of examples we can use to train our systems. However, this particular video did not trigger our automatic detection systems. To achieve that we will need to provide our systems with large volumes of

[1] www.cnbc.com/2019/03/21/why-facebooks-ai-didnt-detect-the-new-zealand-mosque-shooting-video.html

data of this specific kind of content, something which is difficult as these events are thankfully rare. Another challenge is to automatically discern this content from visually similar, innocuous content—for example if thousands of videos from live-streamed video games are flagged by our systems, our reviewers could miss the important real-world videos where we could alert first responders to get help on the ground.[2]

It also did not help that there were various bad actors that re-uploaded edited versions of the video in order to foil Facebook's AI system.

Of course, this was a big lesson in the shortfalls of technology, and the company says it is committed to keep improving its systems. But the Facebook case study also highlights that even the most technologically sophisticated companies have major challenges. This is why when it comes to implementing AI, there needs to be solid planning as well as an understanding that there will inevitably be problems. But it can be tough as senior managers at companies are under pressure to get results from this technology.

In this chapter, we'll take a look at some of the best practices for AI implementations.

Approaches to Implementing AI

Using AI in a company generally involves two approaches: using vendor software or creating in-house models. The first one is the most prevalent—and may be enough for a large number of companies. The irony is that you may already be using software, say from Salesforce.com, Microsoft, Google, Workday, Adobe, or SAP, that already has powerful AI capabilities. In other words, a good approach is to make sure you are taking advantage of these to the fullest.

To see what's available, take a look at Salesforce.com's Einstein, which was launched in September 2016. This AI system is seamlessly embedded into the main CRM (Customer Relationship Management) platform, allowing for more predictive and personalized actions for sales, service, marketing, and commerce. Salesforce.com calls Einstein a "personal data scientist," as it is fairly easy to use, such as with drag and drop to create the workflows. Some of the capabilities include the following:

- Predictive Scoring: This shows the likelihood that a lead will convert into an opportunity.

- Sentiment Analysis: This provides a way to get a sense of how people view your brand and products by analyzing social media.

- Smart Recommendations: Einstein crunches data to show what products are the most ideal for leads.

However, while these prebuilt features make it easier to use AI, there are still potential issues. "We have been building AI functions into our applications during the past few years and this has been a great learning experience," said Ricky Thakrar, who is Zoho's customer experience evangelist. "But to make the technology work, the users must use the software right. If the sales people are not inputting information correctly, then the results will likely be off. We also found that there should be at least three months of usage for the models to get trained. And besides, even if your employees are doing everything right, this does not mean that the AI predictions will be perfect. Always take things with a grain of salt."[3]

Now as for building your own AI models, this is a significant commitment for a company. And this is what we'll be covering in this chapter.

But regardless of what approach you may take, the implementation and use of AI should first begin with education and training. It does not matter whether the employees are non-technical people or software engineers. For AI to be successful in an organization, everyone must have a core understanding of the technology. Yes, this book will be helpful but there are many online resources to help out as well, such as from training platforms like Lynda, Udacity, and Udemy. They provide hundreds of high-quality courses on many topics about AI.

To give a sense of what a corporate training program looks like, consider Adobe. Even though the company has incredibly talented engineers, there are still a large number who do not have a background in AI. Some of them may not have specialized in this in school or their work. Yet Adobe wanted to ensure that all the engineers had a solid grasp of the core principles of AI. To this end, the company has a six-month certification program, which trained 5,000 engineers in 2018. The goal is to unleash the data scientist in each engineer.

The program includes both online courses and in-person sessions, which not only cover technical topics but also areas like strategy and even ethics. Adobe also provides help from senior computer scientists to assist students to master the topics.

Next, early on in the implementation process, it's essential to think about the potential risks. Perhaps one of the most threatening is bias since it can easily seep into an AI model.

An example of this is Amazon.com, which shut down its AI-powered recruiting software in 2017. The main issue was that it was biased for hiring males. Interestingly enough, this was a classic case of a training problem for the model. Consider that a majority of the resume submissions were from men—so the data was skewed. Amazon.com even tried to tweak the model, but still the results were far from being gender neutral.[4]

In this case, the issue was not just about making decisions that were based on faulty premises. Amazon.com was also probably exposing itself to potential legal liability, such as with discrimination claims.

Given the tricky issues with AI, more companies are putting together ethics boards. But even this can be fraught with problems. Hey, what may be ethical for one person may not be a big deal for someone else, right? Definitely.

For example, Google closed down its own ethics board in about a week of its launch. It appears the main reason was the backlash that came from including a member from the Heritage Foundation, which is a conservative think tank.[5]

The Steps for AI Implementation

If you plan to implement your own AI models, what are the main steps to consider? What are the best practices? Well, first of all, it's critically important that your data is fairly clean and structured in a way to allow for modelling (see Chapter 2).

Here are some other steps to look at:

- Identify a problem to solve.
- Put together a strong team.
- Select the right tools and platforms.
- Create the AI model (we went through this process in Chapter 3).
- Deploy and monitor the AI model.

Let's take a look at each.

Identify a Problem to Solve

Founded in 1976, HCL Technologies is one of the largest IT consulting firms, with 132,000 employees across 44 countries, and has half the Fortune 500 as customers. The company also has implemented a large number of AI systems.

Here's what Kalyan Kumar, who is the corporate vice president and global CTO of HCL Technologies, has to say:

> Business leaders need to understand and realize that the adoption of Artificial Intelligence is a journey and not a sprint. It is critical that the people driving AI adoption within an enterprise remain realistic about the timeframe and what AI is capable of doing. The relationship between humans and AI is mutually empowering, and any AI implementation may take some time before it starts to make a positive and significant impact.[6]

It's great advice. This is why—especially for companies that are starting in the AI journey—it's essential to take an experimental approach. Think of it as putting together a pilot program—that is, you are in the "crawl and walk phase."

But when it comes to the AI implementation process, it's common to get too focused on the different technologies, which are certainly fascinating and powerful. Yet success is far more than just technology; in other words, there must first be a clear business case. So here are some areas to think about when starting out:

- No doubt, decisions in companies are often ad hoc and, well, a matter of guessing! But with AI, you have an opportunity to use data-driven decision-making, which should have more accuracy. Then where in your organization can this have the biggest benefit?

- As seen with Robotic Process Automation (RPA), which we covered in Chapter 5, AI can be extremely effective when handling repetitive and mundane tasks.

- Chatbots can be another way to start out with AI. They are relatively easy to set up and can serve specific use cases, such as customer service. You can learn more about this in Chapter 6.

---

[4] www.reuters.com/article/us-amazon-com-jobs-automation-insight/amazon-scraps-secret-ai-recruiting-tool-that-showed-bias-against-women-idUSKCN1MK08G

Andrew Ng, who is the CEO of Landing AI and the former head of Google Brain, has come up with various approaches to think about when identifying what to focus on with your initial AI project:[7]

- *Quick Win:* A project should take anywhere from 6 to 12 months and must have a high probability of success, which should help provide momentum for more initiatives. Andrew suggests having a couple projects as it increases the odds of getting a win.

- *Meaningful:* A project does not have to be transformative. But it should have results that help improve the company in a notable way, creating more buy-in for additional AI investments. The value usually comes from lower costs, higher revenues, finding new extensions of the business, or mitigating risks.

- *Industry-Specific Focus:* This is critical since a successful project will be another factor in boosting buy-in. Thus, if you have a company that sells a subscription service, then an AI system to lessen churn would be a good place to start.

- *Data:* Do not limit your options based on the amount of data you have. Andrew notes that a successful AI project may have as little as 100 data points. But the data must still be high quality and fairly clean, which are key topics covered in Chapter 2.

When looking at this phase, it is also worth evaluating the "tango" between employees and machines. Keep in mind that this is often missed—and it can have adverse consequences on an AI project. As we've seen in this book, AI is great at processing huge amounts of data with little error at great speed. The technology is also excellent with predictions and detecting anomalies. But there are tasks that humans do much better, such as being creative, engaging in abstraction, and understanding concepts.

Note the following example of this from Erik Schluntz, who is the co-founder and CTO at Cobalt Robotics:

Our security robots are excellent at detecting unusual events in workplace and campus settings, like spotting a person in a dark office with AI-powered thermal-imaging. But one of our human operators then steps

in and makes the call of how to respond. Even with all of AI's potential, it's still not the best mission-critical option when pitted against constantly changing environmental variables and human unpredictability. Consider the gravity of AI making a mistake in different situations—failing to detect a malicious intruder is much worse than accidentally sounding a false alarm to one of our operators.[8]

Next, make sure you are clear-cut about the KPIs and measure them diligently. For example, if you are developing a custom chatbot for customer service, you might want to measure against metrics like the resolution rate and customer satisfaction.

And finally, you will need to do an IT assessment. If you have mostly legacy systems, then it could be more difficult and expensive to implement AI, even if vendors have APIs and integrations. This means you will need to temper your expectations.

Despite all this, the investments can truly move the needle, even for old-line companies. To see an example of this, consider Symrise, whose roots go back more than 200 years in Germany. As of this writing, the company is a global producer of flavors and fragrances, with over 30,000 products.

A few years ago, Symrise embarked on a major initiative, with the help of IBM, to leverage AI to create new perfumes. The company not only had to retool its existing IT infrastructure but also had to spend considerable time fine-tuning the models. But a big help was that it already had an extensive dataset, which allowed for more precision. Note that even a slight deviation in the mixture of a compound can make a perfume fail.

According to Symrise's president of Scent and Care, Achim Daub:

Now our perfumers can work with an AI apprentice by their side, that can analyze thousands of formulas and historical data to identify patterns and predict novel combinations, helping to make them more productive, and accelerate the design process by guiding them toward formulas that have never been seen before.[9]

---

[8]This is based on the author's interview, in April 2019, with Erik Schluntz, who is the co-founder and CTO at Cobalt Robotics.

Forming the Team

How large should the initial team be for an AI project? Perhaps a good guide is to use Jeff Bezos' "two pizza rule."[10] In other words, is this enough to feed the people who are participating?

Oh, and there should be no rush to build the team. Everyone must be highly focused on success and understand the importance of the project. If there is little to show from the AI project, the prospects for future initiatives could be in jeopardy.

The team will need a leader who generally has a business or operational background but also has some technical skills. Such a person should be able to identify the business case for the AI project but also communicate the vision to multiple stakeholders in the company, such as the IT department and senior management.

In terms of the technical people, there will probably not be a need for a PhD in AI. While such people are brilliant, they are often focused primarily on innovations in the field, such as by refining models or creating new ones. These skillsets are usually not essential for an AI pilot.

Rather, look for those people who have a background in software engineering or data science. However, as noted earlier in the chapter, these people may not have a strong background in AI. Because of this, there may be a need to have them spend a few months of training on learning the core principles of machine learning and deep learning. There should also be a focus on understanding how to use AI platforms, such as TensorFlow.

Given the challenges, it may be a good idea to seek the help of consultants, who can help identify the AI opportunities but also provide advice on data preparation and the development of the models.

Since an AI pilot will be experimental, the team should have people who are willing to take risks and are open minded. If not, progress could be extremely difficult.

The Right Tools and Platforms

There are many tools for helping create AI models, and most of them are open source. Even though it's good to test them out, it is still advisable to first conduct your IT assessment. By doing this, you should be in a better position to evaluate the AI Tools.

Something else: You may realize that your company is already using multiple AI Tools and platforms! This may cause issues with integration and the management of the process with AI projects. In light of this, a company should develop a strategy for the tools. Think of it as your AI Tools stack.

OK then, let's take a look at some of the more common languages, platforms, and tools for AI.

Python Language

Guido van Rossum, who got his master's degree in mathematics and computer science from the University of Amsterdam in 1982, would go on to work at various research institutes in Europe like the Corporation for National Research Initiatives (CNRI). But it was in the late 1980s that he would create his own computer language, called Python. The name actually came from the popular British comedy series *Monty Python*.

So the language was kind of offbeat—but this made it so powerful. Python would soon become the standard for AI development.

Part of this was due to the simplicity. With just a few scripts of code, you can create sophisticated models, say with functions like filter, map, and reduce. But of course, the language allows for much sophisticated coding as well.

Van Rossum developed Python with a clear philosophy:[11]

- Beautiful is better than ugly.
- Explicit is better than implicit.
- Simple is better than complex.
- Complex is better than complicated.
- Flat is better than nested.
- Sparse is better than dense.

These are just some of the principles.

What's more, Python had the advantage of growing in the academic community, which had access to the Internet that helped accelerate the distribution. But it also made it possible for the emergence of a global ecosystem with thousands of different AI packages and libraries. Here are just some:

- *NumPy:* This allows for scientific computing applications. At the heart of this is the ability to create a sophisticated array of objects at high performance. This is critical for high-end data processing in AI models.

- *Matplotlib:* With this, you can plot datasets. Often Matplotlib is used in conjunction with NumPy/Pandas (Pandas refers to "Python Data Analysis Library"). This library makes it relatively easy to create data structures for developing AI models.

- *SimpleAI:* This is an implementation of the AI algorithms from the book *Artificial Intelligence: A Modern Approach,* by Stuart Russel and Peter Norvig. The library not only has rich functionality but also provides helpful resources to navigate the process.

- *PyBrain:* This is a modular machine learning library that makes it possible to create sophisticated models—neural networks and reinforcement learning systems—without much coding.

- *Scikit-Learn:* Launched in 2007, this library has a deep source of capabilities, allowing for regression, clustering, and classification of data.

Another benefit for Python is that there are many resources for learning. A quick search on YouTube will show thousands of free courses.

Now there are other solid languages you can use for AI like C++, C#, and Java. While they are generally more powerful than Python, they are also complex. Besides, when it comes to building models, there is often little need to create full-fledged applications. And finally, there are Python libraries built for high-speed AI machines—with GPUs—like CUDA Python.

AI Frameworks

There are a myriad of AI frameworks, which provide end-to-end systems to build models, train them, and deploy them. By far the most popular is TensorFlow, which is backed by Google. The company started development of this framework in 2011, through its Google Brain division. The goal was to find a way to create neural networks faster so as to embed the technology across many Google applications

By 2015, Google decided to open source TensorFlow, primarily because the company wanted to accelerate the progress of AI. And no doubt, this is what happened. By open sourcing TensorFlow, Google made its technology an

41 million times, and there are more than 1,800 contributors. In fact, TensorFlow Lite (which is for embedded systems) is running on more than 2 billion mobile devices.[12]

The ubiquity of the platform has resulted in a large ecosystem. This means there are many add-ons like TensorFlow Federated (for decentralized data), TensorFlow Privacy, TensorFlow Probability, TensorFlow Agents (for reinforcement learning), and Mesh TensorFlow (for massive datasets).

To use TensorFlow, you have the option of a variety of languages to create your models, such as Swift, JavaScript, and R. Although, for the most part, the most common one is Python.

In terms of the basic structure, TensorFlow takes in input data as a multidimensional array, which is also known as a tensor. There is a flow to it, represented by a chart, as the data courses through the system.

When you enter commands into TensorFlow, they are processed using a sophisticated C++ kernel. This allows for much higher performance, which can be essential as some models can be massive.

TensorFlow can be used for just about anything when it comes to AI. Here are some of the models that it has powered:

- Researchers from NERSC (National Energy Research Scientific Computing Center) at the Lawrence Berkeley National Laboratory created a deep learning system to better predict extreme weather. It was the first such model that broke the expo (1 billion billion calculations) computing barrier. Because of this, the researchers won the Gordon Bell Prize.[13]

- Airbnb used TensorFlow to build a model that categorized millions of listing photos, which increased the guest experience and led to higher conversions.[14]

- Google used TensorFlow to analyze data from NASA's Kepler space telescope. The result? By training a neural network, the model discovered two exoplanets. Google also made available the code to the public.[15]

Google has been working on TensorFlow 2.0, and a key focus is to make the API process simpler. There is also something called Datasets, which helps to streamline the preparation of data for AI models.

[12] https://medium.com/tensorflow/recap-of-the-2019-tensorflow-dev-summit-1b5ede42da8d

[13] www.youtube.com/watch?v=p45k0klIsd4&feature=youtu.be

Then what are some of the other AI frameworks? Let's take a look:

- *PyTorch:* Facebook is the developer of this platform, which was released in 2016. Like TensorFlow, the main language to program the system is Python. While PyTorch is still in the early phases, it is already considered the runner-up to TensorFlow in terms of usage. So what is different with this platform? PyTorch has a more intuitive interface. The platform also allows for dynamic computation of graphs. This means you can easily make changes to your models in runtime, which helps speed up development. PyTorch also makes it possible for having different types of backend CPUs and GPUs.

- *Keras:* Even though TensorFlow and PyTorch are for experienced AI experts, Keras is for beginners. With a small amount of code—in Python—you can create neural networks. In the documentation, it notes: "Keras is an API designed for human beings, not machines. It puts user experience front and center. Keras follows best practices for reducing cognitive load: it offers consistent and simple APIs, it minimizes the number of user actions required for common use cases, and it provides clear and actionable feedback upon user error."[16] There is a "Getting Started" guide that takes only 30 seconds! Yet the simplicity does not mean that it is not powerful. The fact is that you can create sophisticated models with Keras. For example, TensorFlow has integrated Keras on its own platform. Even for those who are pros at AI, the system can be quite useful for doing initial experimentations with models.

With AI development, there is another common tool: Jupyter Notebook. It's not a platform or development tool. Instead, Jupyter Notebook is a web app that makes it easy to code in Python and R to create visualizations and import AI systems. You can also easily share your work with other people, similar to what GitHub does.

During the past few years, there has also emerged a new category of AI Tools called automated machine learning or autoML. These systems help to deal with processes like data prep and feature selection. For the most part, the goal is to provide help for those organizations that do not have experienced data scientists and AI engineers. This is all about the fast-growing trend of the "citizen data scientist"—that is, a person who does not have a strong technical background who can still create useful models.

Some of the players in the autoML space include H2O.ai, DataRobot, and SaaS. The systems are intuitive and use drag-and-drop ease with the development of models. As should be no surprise, mega tech operators like Facebook and Google have created autoML systems for their own teams. In the case of Facebook, it has Asimo, which helps manage the training and testing of 300,000 models every month.[17]

For a use case of autoML, take a look at Lenovo Brazil. The company was having difficulty creating machine learning models to help predict and manage the supply chain. It had two people who coded 1,500 lines of R code each week—but this was not enough. The fact is that it would not be cost-effective to hire more data scientists.

Hence the company implemented DataRobot. By automating various processes, Lenovo Brazil was able to create models with more variables, which led to better results. Within a few months, the number of users of DataRobot went from two to ten.

Table 8-1 shows some other results.[18]

Table 8-1. The results of implementing an autoML system

Tasks	Before	After
Model creation	4 weeks	3 days
Production models	2 days	5 minutes
Accuracy of predictions	<80%	87.5%

Pretty good, right? Absolutely. But there are still come caveats. With Lenovo Brazil, the company had the benefit of skilled data scientists, who understood the nuances of creating models.

However, if you use an autoML tool without such expertise, you could easily run into serious trouble. There's a good chance that you may create models that have faulty assumptions or data. If anything, the results may ultimately prove far worse than not using AI! Because of this, DataRobot actually requires that a new customer have a dedicated field engineer and data scientist work with the company for the first year.[19]

Now there are also low-code platforms that have proven to be useful in accelerating the development of AI projects. One of the leaders in the space is Appian, which has the bold guarantee of "Idea to app in eight weeks."

[17] www.aimlmarketplace.com/technology/machine-learning/the-rise-of-automated-machine-learning

[18] https://3ep10c1vpv442i63me73pv3s-wpengine.netdna-ssl.com/wp-content/

With this platform, you can easily set up the data structure that is clean. There are even systems in place to help guide the process, such as alerting for issues. No doubt, this provides a solid foundation for building a model. But low-code also helps in other ways. For example, you can test various AI platforms—say from Google, Amazon, or Microsoft—to see which one performs better. Then you can create the app with a modern interface and deploy it to the Web or mobile apps.

To get a sense of the power of low-code, take a look at what KPMG has done with the technology. The company was able to help its clients transition away from the use of LIBOR in loans. First of all, KPMG used its own AI platform, called Ignite, to ingest the unstructured data and use machine learning and Natural Language Processing to remediate the contracts. Next, the company used Appian to help with document sharing, customizable business rules, and real-time reporting.

Such a process—when done manually—could easily take thousands of hours, with the error rate of 10% to 15%. But when using Ignite/Appian, the accuracy was over 96%. Oh, and the time to process the documents was in seconds.

Deploy and Monitor the AI System

Even when you build an AI model that works, there is still more work to do. You need to find ways to deploy and monitor it.

This requires change management, which is always complex and difficult. AI is different than a typical IT implementation since it involves using predictions and insights for decision-making. This means people will need to rethink how they interact with the technology.

Also consider that the chances are that the end-users will be non-technical people, whether employees or consumers. This is why there needs to be so much work on making the AI model as easy as possible. For example, if you have built a system for online marketing, you might want to limit the options for the user—say to just four or five of them.

Why? If there are too many, then users may get frustrated and not even know where to start. This is all part of the so-called "analysis paralysis" problem. When this happens, there will inevitably be little adoption of the AI model, which will severely result in impeded progress.

Another good strategy is to use visualizations that are interactive. In other words, you can easily see how the trends change by adjusting some variables. You can also allow for clicking a certain part of the chart to drill down into more details.

It's also essential to create documentation. But this should be more than just written materials. For example, an effective approach is to develop video tutorials. Such an effort will go a long way in creating strong adoption.

As a best practice, the initial deployment should be limited. Perhaps this could be to a small group of beta users and a small section of the customer base. There should also be warnings that the AI model is in the early stages and may have bugs.

Therefore, this phase is about learning. What works? What should be removed? Where can things be improved?

This is definitely an iterative process that must not be rushed.

Then once the AI model is ready for full deployment, there should be enough support in place and someone to lead the management of the project. There also must be recognition for the team for the win. Hopefully, the praise will come from the highest levels of the company, which will help encourage more and more innovation.

There are a variety of automated platforms to help streamline the workflow process, such as Alteryx. The company's vision is to democratize data science and analytics, regardless if someone has a technical background or not. The Alteryx system handles the key areas of the process: data discovery, data preparation, analytics, and deployment. And all of this is done with code-free drag-and-drop tools. Furthermore, many of the company's customers are non-technology operators like Hyatt, Unilever, and Kroger.

Again, AI development is really a journey—and your strategy will inevitably change. This is inevitable. According to Kurt Muehmel, who is the VP of Sales Engineering at Dataiku[20]:

> What businesses sometimes fail to realize is that the path to AI is a long-term evolution of not only technology but in the way the company collaborates and works together. So, in addition to education, one of the key components to an AI strategy should be overall change management. It is important to create both short- and long-term roadmaps of what will be accomplished with first maybe predictive analytics, then perhaps machine learning, and ultimately—as a longer-term goal—AI, and how each roadmap impacts various pieces of the business as well as people who are a part of those business lines and their day-to-day work.

Conclusion

As shown in this chapter, when approaching implementing AI, it's critical to look at two paths. The first is to get the maximum use of any third-party systems that use the technology. But there should also be a focus on data quality. If not, the results will likely be off the mark.

The second path is to do an AI project, which is based on your company's own data. To be successful, there must be a strong team that has a blend of technical, business, and domain expertise. There will also likely be a need for some AI training. This is the case even for those with backgrounds in data science and engineering.

From here, there should be no rush in the steps of the project: assessing the IT environment, setting up a clear business objective, cleaning the data, selecting the right tools and platforms, creating the AI model, and deploying the system. With early projects, there will inevitably be challenges so it's critical to be flexible. But the effort should be well worth it.

Key Takeaways

- Even the best companies have difficulties with implementing AI. Because of this, there must be great care, diligence, and planning. It's also important to realize that failure is common.

- There are two main ways to use AI in a company: through a vendor's software application or an in-house model. The latter is much more difficult and requires a major commitment from the organization.

- When using off-the-shelf AI applications, there is still much work to be done. For example, if the employees are not correctly inputting the data, then the results will likely be off.

- Education is critical with an AI implementation, even for experienced engineers. There are excellent online training resources to help out with this.

- Be mindful of the risks of AI implementations, such as bias, security, and privacy.

- Some of the key parts of the AI implementation process include the following: identify a problem to solve; put together a strong team; select the right tools and

- When developing a model, look at how the technology relates to people. The fact is that people can be much better at certain tasks.

- Forming the team is not easy, so do not rush the process. Have a leader who has a good business or operational background, with a mix of technical skills.

- It's good to experiment with the various AI Tools. However, before doing this, make sure you do an IT assessment.

- Some of the popular AI Tools include TensorFlow, PyTorch, Python, Keras, and the Jupyter Notebook.

- Automated machine learning or autoML tools help to deal with processes like data prep and feature selection for AI models. The focus is on those who do not have technical skills.

- Deployment of the AI model is more than just scaling. It's also critical to have the system easy to use, so as to allow for much more adoption.