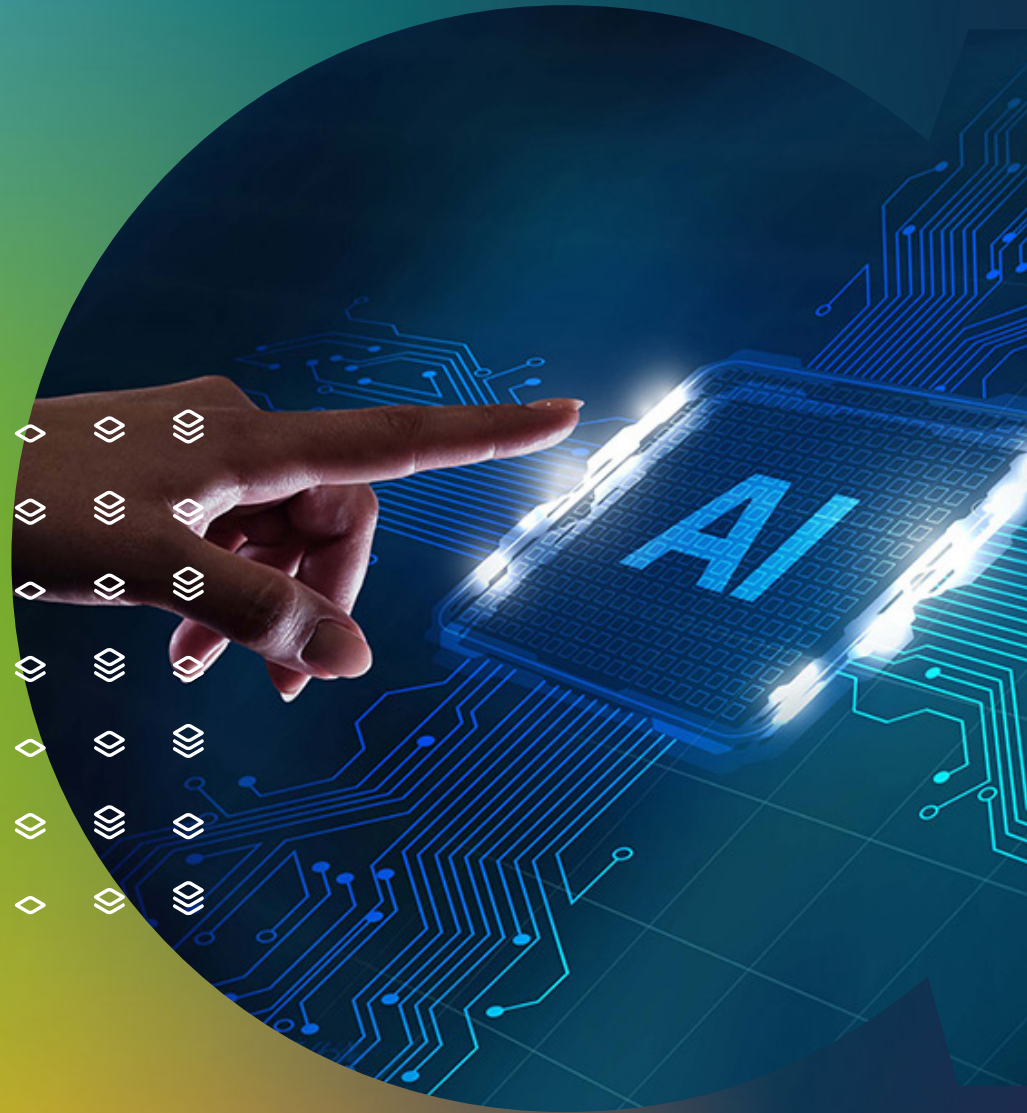


White paper

Limitless possibilities? Exploring the **potential** of **generative AI** in software



Mastering the Microsoft stack





Limitless possibilities? Exploring the potential of generative AI in software

For as long as we can remember, the world of software development has been captivated by the vast potential of artificial intelligence (AI).

AI can already draw on an impressive highlights reel. Nowadays, almost everyone uses some sort of AI service on a daily basis as part of the apps on their phone, from natural language processing to machine learning (ML) algorithms. More recently, AI and ML have been touted as the next big thing in numerous (Microsoft) keynotes. So far, the technology has mainly focused on recognition or prediction.

But the latest wave of AI has shifted gears and is now generative, meaning it's capable of producing content. This opens up a whole new world of possibilities, of which we believe we've only seen the beginning. Watching some of the demos and experimenting with the technology, it's hard not to be impressed.

In this whitepaper, we'll explore artificial intelligence models such as ChatGPT, Midjourney, and DALL-E, as well as Microsoft Azure offerings, known as Cognitive Services. We'll discuss the opportunities as well as the potential threats (to quote the Spiderman movies: "With great power comes great responsibility."). This will not be a deep-dive into the technology, but rather a targeted approach aimed at chief technology officers and other key decision-makers, to provide them with a solid understanding of the developments in this field. Let's get started ...

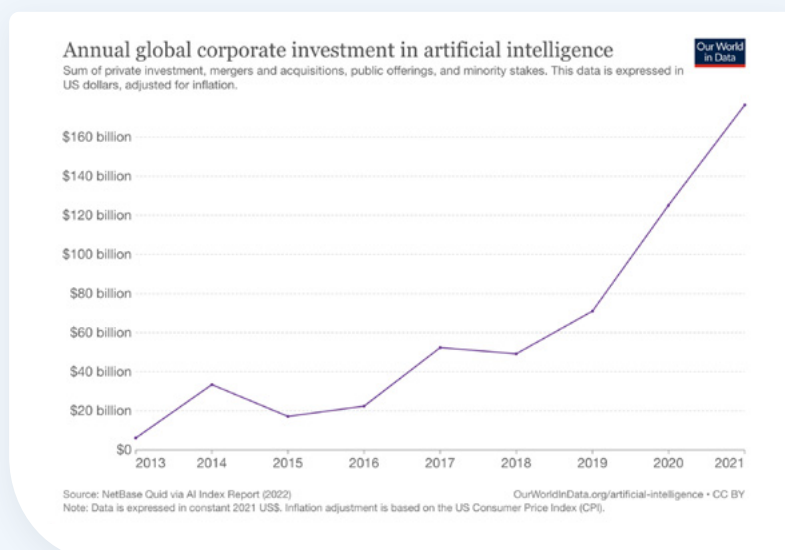
What is artificial intelligence?

Before discussing the impact of AI on software development, we need to establish a common understanding of what artificial intelligence actually means.

In its most basic terms, artificial intelligence (AI) involves the creation of intelligent software and hardware that can perform tasks that would typically require human intelligence. This includes tasks that require reasoning, perception, or decision-making.

Consider a simple everyday task such as screwing on a bottle cap. While this task has long been automated, it's not typically considered to require human intelligence. Some animals could be trained to perform this task. The machines that perform this task are considered "dumb" because they simply screw caps onto bottles.

However, the perception of AI changes when the machine is able to inspect the label on the bottle, determine the brand, and select the correct cap to use. While this is still within the realm of traditional programming, it feels more intelligent. In the same way, modern factories have become more "intelligent" over time. Industrial automation is growing up, much like a child. And with each passing day, factories are getting smarter and more efficient.

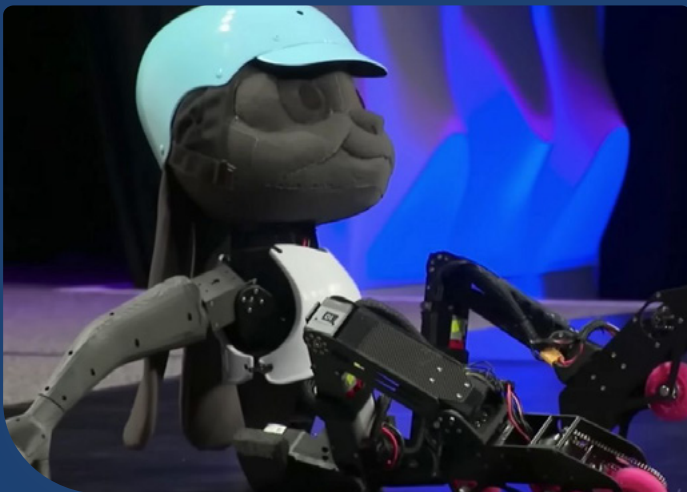


Source: ourworldindata/ai-investments

Growing fast:
Increase in global AI investment

The notion that automation is coming of age is not just a metaphorical one. The latest iteration of the Atlas robot, developed by Boston Dynamics, can parkour, carry tools, and perform other tasks that could easily be considered intelligent. While many people and companies have yet to encounter these kinds of high-tech robots in their daily business operations, tasks that require human intelligence can be found in almost every industry. And some of these tasks may be ripe for software takeover in the near future. A recent research paper ¹ concludes: "Up to 49% of workers could have half or more of their tasks exposed to LLMs (Large Language Models)", which we'll discuss in more detail in this white paper. This shift towards automation in various industries presents both opportunities and challenges for companies.

¹ [2303.10130] GPTs are GPTs: An Early Look at the Labor Market Impact Potential of Large Language Models ([arxiv.org](https://arxiv.org/abs/2303.10130))



Fantasia reborn: Disney puts its own spin on AI

At the 2023 South by Southwest[®] Conference, Disney showcased a prototype robot of their beloved character: Judy Hopps. This robot is designed to behave like a child on roller skates, and its software is largely responsible for making the hardware perform intelligent actions.

Disney's Judy Hopps robot prototype
at SXSW 2023



Boston Dynamics' Atlas robot



A pattern emerges

As discussed, AI has evolved from focusing primarily on pattern recognition to being able to generate new content based on those patterns. To better understand how data and patterns work, let's again use children as an example.

Every parent remembers the moment when their child starts to recognize them, and the joy that comes with that little smile. Over time, children begin to recognize and group other things. By using picture books, parents can teach their children to understand the difference between a cat and a dog. Children will make mistakes along the way, which is perfectly fine – as we're taught from our first day at pre-school: mistakes are how we learn.

A similar principle applies to artificial intelligence, particularly in the realm of machine learning. We feed our algorithm images of cats and dogs, and the result is a fully trained model. And, just like natural intelligence, AI is prone to making errors during the course of its learning.

In other words, trial and error is at the basis of how the technology improves, just like in humans.

Yet society is generally less forgiving of machines than it is of children. The bar is set too high; to unlock the long-term potential of generative AI, we need to transition to a situation where we embrace making mistakes as part of the learning process.

There are many technical details that we won't go into in this whitepaper. But it's sufficient to say that AI models have matured over time, with deep learning being an advancement that uses neural networks to learn from data and make predictions. This technology can be applied to streaming video to identify potential hazards in a workplace or to inspect potential cracks in pipes, among other things. It's important to note, however, that all of these examples are based on taking input and performing some sort of analysis or classification based on what was learned during training.



Generative AI – Now for the good stuff

In recent decades, the primary focus of AI and ML research has shifted from detection toward prediction and, most recently, generation. The poster child of this evolution has been San Francisco-based AI research lab, OpenAI, which released its flagship solution, ChatGPT, in late 2022.

Detection handles things like classification – for instance, picking out malformed products in a product line. A well-known example of this is found in the field of predictive maintenance, where machines can be taken offline for maintenance in a scheduled way to prevent them from failing in an unscheduled way. The newer, generative-based family of models are primarily targeted to generating different types of data on their own. Generative models work similarly to the machine learning and deep-learning algorithms we discussed earlier, with the main difference being that the generative models are designed to generate new data that is similar but not identical to the training data. For example, Midjourney is an AI that can generate an image in response to a textual prompt.

AI steps up its game

We asked Midjourney to create an image of a Nike shoe in the style of street photography. The resulting images show authentic-looking, algorithm-generated shoes that do not exist in real life. The algorithm takes the prompt, gathers knowledge based on the training data, and synthesizes what it thinks would be a valid result based on the prompt. And because it knows what a normal Nike shoe looks like, it produces something similar. It's worth noting that the algorithm's ability to recognize Nike shoes is based on training data obtained from the internet.

This picture was created by Midjourney.

This is not a real photo.



It begs the question: if an algorithm is capable of producing these kinds of images based on our prompts, why would Nike still need designers for its shoes? Or maybe the company can simply reduce the number of designers on its books over time? Nike may be asking itself the same question, because even if it doesn't use these algorithms, its competitors probably will.

It's clear that where existing algorithms weren't really creating new data, these new algorithms can. This creates exciting new opportunities but it also raises a lot of questions.



ChatGPT: A star is born

Now to the latest star of the AI show: ChatGPT: an OpenAI-built service based on a large multimodal model (LMM) named GPT. GPT-4 (the latest release at the time of writing) is capable of processing both image and text inputs, and emitting text outputs. Like Midjourney, GPT-4 has been trained on massive amounts of data from the internet using deep learning techniques to process and analyze text data, allowing the model to learn patterns, structure, and nuances of natural language. ChatGPT is OpenAI's web-based chat interface for this model.

When given a prompt, the model generates the next token (either a word or part of a word) in a sentence based on the context and words (or tokens) that came before it, resulting in sentences that sound like they were written by a human. The prompts can even be used to instruct the model on specific writing styles, such as formal or informal, or even the style of a particular author.

While the text produced by GPT-4 is often impressive and difficult to distinguish from human-written content, it is not always completely accurate. The model is trained to produce linguistically and grammatically correct sentences, but it may not always be able to verify facts or produce entirely accurate information. This phenomenon is known as "hallucination", and it's one of the main shortcomings of this technology.

Despite this limitation, GPT-4 can still provide valuable feedback and suggestions. For example, we had GPT-4 review this white paper and provide feedback on how to improve it. The model offered some good suggestions, which we could choose to implement or not. In fact, with the right prompts, GPT-4 could even write an entire white paper.

It's hard not to be impressed by this technology and wonder what similar advances will bring in the future. While it's impossible to predict, we can start by exploring what the technology can do now and how it will impact the workplace.



Microsoft enters the fray – with GitHub Copilot

As a Microsoft partner, we wanted to take a closer look at a new AI-powered technology from Microsoft that is having a significant impact on the day-to-day jobs of developers: GitHub Copilot.

Like GPT, GitHub Copilot runs off the generative AI concept; however, this model has been trained on a smaller and more specific subset of data, with all the code stored in public GitHub.com code repositories. The model behind Copilot is called “codex”. By studying all the code in GitHub, codex learned how to code – or rather, how to predict what code should be written based on textual prompts.

Copilot uses what the developer has already typed to start predicting the code they want to write next. For example, if we want to sort a list of people objects by their first name, all we have to do is give it a little context (in this case the name of the function), and it will predict the rest, as shown in the following example:

```
public List<Person> SortByFirstName(List<Person> people)
{
    return people.OrderBy(p => p.FirstName).ToList();
}
```

The gray text is what it predicts the method should look like, and hitting TAB accepts the suggestion (at least that’s the developer’s experience in Visual Studio) and types all that code with a single keystroke. The developer will likely still review the code, but “writing” it becomes easier. As confidence in the predictions grows, reviews may become more and more superficial. And with that, we might start to miss any hallucinations that the models produce.

The above is a small simple example. But Copilot can write entire method bodies for you. Let’s give it the task of summarizing all of the digits of a given number.

```
0 references | 0 changes | 0 authors, 0 changes | 0 exceptions, - live
public int SumAllIndividualDigits(int someNumber)
{
    var digits = someNumber.ToString().ToCharArray();
    var sum = 0;
    foreach (var digit in digits)
    {
        sum += int.Parse(digit.ToString());
    }
    return sum;
}
```

```
0 references | 0 changes | 0 authors, 0 changes | 0 exceptions, - live
public int SumAllIndividualDigits(int someNumber)
{
    var sum = 0;
    while (someNumber > 0)
    {
        sum += someNumber % 10;
        someNumber /= 10;
    }
    return sum;
}
```

The two screenshots above show two valid solutions to a specific problem as produced by GitHub Copilot. In this case, both solutions are factually correct (with the small exception of the second example, which has a redundant set of braces), but use a different approach. One developer might care about which solution is “best” (arguably), others might not. When generating code like this becomes more and more the default way of working, who will safeguard quality?

GitHub Copilot can be seamlessly integrated into the developer’s IDE, making it very easy to include in their normal coding routine. It’s not limited to a specific language, although Microsoft says it excels in Python but also “knows” most other common languages (the above examples are in C#). It’s also great at performing repetitive tasks, such as writing a series of similar unit tests. It will understand patterns and be able to complete them. In the hands of a developer who understands how to use it efficiently, this can be a major time-saver. More on GitHub Copilot can be found on its website. ⁴



⁴ [GitHub Copilot: Your AI pair programmer](#)

Note: GPT-4 is also quite capable of writing code, but currently requires a specific prompt to do so. For the example above, we asked it: "Write a C# method that takes an integer as input, and returns the sum of all individual digits that make up said integer. The method name should be SumAllIndividualDigits." Here's what it produced:

```
csharp Copy code

public int SumAllIndividualDigits(int num)
{
    int sum = 0;

    // Convert the input number to a string
    string numStr = num.ToString();

    // Loop through each character in the string
    foreach (char c in numStr)
    {
        // Convert the character to an integer
        int digit = int.Parse(c.ToString());

        // Add the digit to the running sum
        sum += digit;
    }

    // Return the final sum
    return sum;
}
```

Again, this is a different but perfectly correct solution to the given problem. ChatGPT even adds comments to explain what the code does.

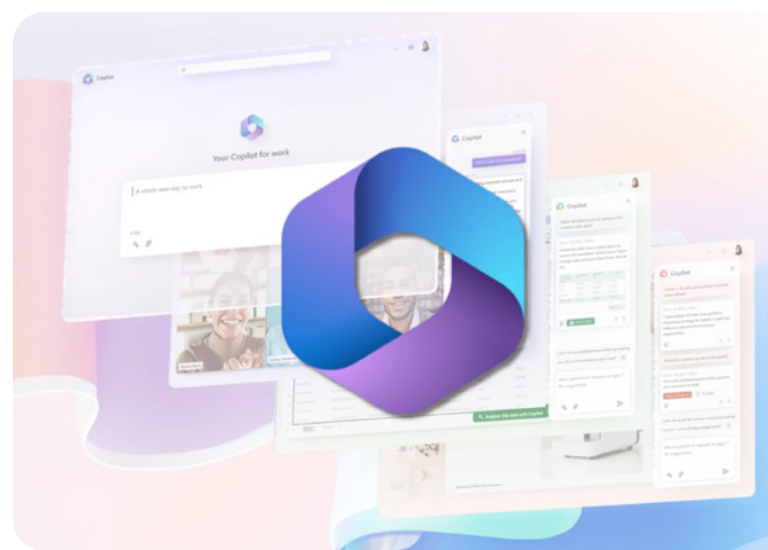
To the next level: Microsoft 365 Copilot

Reusing the “Copilot” name, Microsoft also announced Microsoft 365 Copilot. At its core, this is essentially the same GPT-4 model that both ChatGPT and Bing use, albeit with two main differences:

- **First**, 365 Copilot will be integrated into the Microsoft 365 suite of products, such as Word, Excel, and PowerPoint. Those who remember Office 97 may remember Clippy, the Office assistant that tried to be helpful. However, Clippy was not very successful due to its reputation for being “dumb” and therefore irritating to many virtual assistant users. The power of the GPT model aims to eliminate this annoyance and give users the results they’re looking for.
- **Second**, Microsoft 365 Copilot will have access to a user’s internal data to provide functionality such as summarizing documents, transcribing meetings, and creating task lists based on the agreed information.

It will be able to distill information from Excel sheets and create ready-made presentations in PowerPoint. This is similar to Bing’s ability to search internal documents, emails, and chat.

Essentially, Microsoft 365 Copilot is a private corporate instance of GPT-4 – it’s ChatGPT for business, in layman’s terms. Like Github Copilot, this technology can be a major time saver, and its ability to create may even put some jobs at risk. More details on Microsoft 365 Copilot can be found on the Microsoft blog.⁵



⁵ [Introducing Microsoft 365 Copilot – your copilot for work – The Official Microsoft Blog](#)

As with the previous examples, Microsoft 365 Copilot has significant potential in a software development context. And it's only logical that Microsoft's announcements focus primarily on this opportunity, while also noting that this technology comes with its own set of challenges. Copilot will enable all Microsoft 365 and Office users to be much more productive than they have been. It will empower regular users to take advantage of application functionality that was previously only available to skilled "power users".

For example, think about creating more complex formulas within an Excel sheet: certainly not all Excel users feel comfortable or even know how to do this, but Copilot will translate your prompt into the right formula for you. (however, there are inherent risks: if Copilot recommends incorrect formulae, how will users be able to detect this?). The same goes for creating slide decks with nice animations in PowerPoint, or automating your own workflow with Power Automate. This technology has the potential to open up previously unused capabilities to a much wider audience. And as we saw in the previous Nike example: if you don't use it, your coworker competitor probably will.



Proceed with caution: Where AI falls short

So far, we've explored the vast potential of artificial intelligence and, in particular, generative models. But there are two sides to every story. Scroll through LinkedIn or Twitter, and you'll find countless people passionately opposed to this type of technology. Their concerns include:

- **Reliability:** If you can't rely on the model's output, what's the point? You'll have to double-check everything, which defeats the purpose of efficiency. **Those who fail to do their due diligence also run the real (and potentially dangerous) risk of using incorrect information.**

A GPT-4 research paper on this topic says: "Despite its capabilities, GPT-4 has similar limitations as earlier GPT models. Most importantly, it still is not fully reliable (it "hallucinates" facts and makes reasoning errors). Great care should be taken when using language model outputs, particularly in high-stakes contexts, with the exact protocol (such as human review, grounding with additional context, or avoiding high-stakes uses altogether) matching the needs of a specific use-case."⁶

- **Bias:** Algorithms are known to have hidden biases that are difficult to detect. The model may provide a biased answer without being aware of it.
- **Limited creativity:** Although the output generated by these models is largely unique and can be considered creative, it's simply the product of combining known elements. Without new human-generated content, the model's creativity is limited. This is especially true for a technology such as Copilot, which is based on a much smaller dataset to begin with, and code is an area that is still evolving and changing over time.
- **Energy consumption:** Training and running these large models is expensive and consumes a significant amount of energy. Not good news given the current climate crisis.

⁶ [GPT-4 \(openai.com\)](#)



- **Legal implications:** There have been examples whereof ChatGPT user input, which very well might contain (and potentially intellectual property,) being leaked to unauthorized users. It's important to stress that in – as far as we know – all the leaks in each of these cases the leaks were caused by software built around the models, and not by the models themselves. The core model of ChatGPT is a trained model without any form of conversation history or state. The conversation history itself is stored by the chat application built around the model, just like in the same way that the input into feeding the model is not directly stored within the model itself. Put differently: your input is not by default also training input, but it can be. Closely keeping an eye on the terms of conditions and other legal contexts is therefore very important. Cloud-hosted private instances of these models help to mitigate some of these risks.

There are also ethical concerns to consider:

- **Impact on employment:** As AI is used to make tasks more efficient, there is the potential for job displacement. We've seen a similar debate in the translation industry – however, despite warnings that Google Translate would make their jobs obsolete, translators have yet to see a significant impact on demand.⁷
- **Accountability:** There are still many unanswered questions in the legal and regulatory communities about how we want to proceed with this type of technology. As algorithms and neural networks become more complex, it's increasingly difficult to understand how these models arrive at a particular answer. Although users can ask the model to explain why it did what it did, it's uncertain whether it will provide the right answer. This raises questions of accountability: who is responsible for the information that's being served up?

⁷ [When Will Translators And Interpreters Disappear? | by Artur Prokopchuk | Medium](#)

- **Malicious use:** “With great power comes great responsibility.” But history has shown that power can – and will – be abused. In this case, it already is: many spam and scam programs are now actively AI-powered, which makes them harder to detect.

These are just some of the questions we should be asking. But while the priority is to ensure that technology doesn’t do more harm than good, particularly since it will inevitably fall into the wrong hands, risk aversion is unlikely to stand in the way of progress. Ultimately, as long as commercial interests drive this technological arms race without any government involvement, we can be sure that the big tech giants will continue making strides as they race to compete against each other. Regulation seems inevitable, leaving potential investments in this technology at risk.

‘It’s time to embrace AI and all that it can offer, while also considering the implications and risks that come with it.’



Let's talk about AI as a service

Having a solid understanding of what artificial intelligence is and the benefits that generative models can offer is a good starting point. But how do you get from there to actively using AI in your business? Very few companies have a dedicated team of deep-learning experts on the payroll, let alone the computing power to train and run these models.

Fortunately, the cloud has helped to “democratize” this type of technology. Microsoft, Google, and OpenAI all offer their own AI services, which helpfully run on their respective cloud infrastructures. These providers offer multiple options for models, so you can choose the one that best fits your needs.

For the scope of this whitepaper, we focus primarily on Microsoft cloud-based services. In addition to being a Microsoft partner, alfa1 also has expertise working with Google Cloud and AWS. In actual fact, the choice of platform does not make a significant difference as all major cloud providers will have similar capabilities in the long run.

The lure of Azure: Key facts about the current state of Microsoft's AI service

The Microsoft Azure offering, “*Azure OpenAI Service*,” provides a set of generative AI models as a ready-to-use API. At the time of writing, they include GPT-4, GPT-3.5, Codex, and Embeddings.⁸ However, there are a few things to keep in mind when using this service:

- At the time of writing, the service is not yet available to all customers; you will need to request it before adding it to your Azure environment.⁹
- There are restrictions on the use cases for which the service can be used for (terms of service).
- It's only available in select regions, and the exact models available vary by region.
- Pricing is on a pay-per-use basis, based on both training a model and using it.

We can be confident that Microsoft Azure's shortcomings will be addressed in the coming months, and that newer and better models will follow in the years ahead. For companies that want to experiment, everything you need is available right now. Note that Azure's AI offerings extend beyond the OpenAI service. Microsoft's Cognitive Services ¹⁰ include a number of other APIs that can be used in software applications. These APIs deliver a variety of functionalities, such as:

- Speech-to-text and text-to-speech
- Speech and text translation (including question answering)
- Conversational language understanding
- Image and video content analysis
- Anomaly detection

A complete list of features can be found on the Azure website. These APIs make it easy for developers to access complex functionality that would otherwise be difficult to build and host on their own, and require only a small amount of time to digest the API documentation and learn how to use it.

We should note that OpenAI also offers APIs that allow you to build AI functionality into your own applications. A list of examples can be found on the OpenAI website. ¹¹



⁸ [Azure OpenAI Service models - Azure OpenAI | Microsoft Learn](#)

⁹ [Limited access to Azure OpenAI Service - Azure Cognitive Services | Microsoft Learn](#)

¹⁰ [Cognitive Services—APIs for AI Solutions | Microsoft Azure](#)

¹¹ [Examples - OpenAI API](#)

What does this mean for *your* company?

Exactly how large language models will impact your business will depend on your specific circumstances. However, we expect this technology to have significant implications for most businesses. This could take the form of companies adapting their products or services to incorporate some form of AI, or building AI into their internal processes and software applications to automate tasks that previously required human intervention. It could also lead to a fundamental shift in the way businesses operate, with greater emphasis on increasing worker efficiency through AI-powered services such as Copilot.

As it stands, generative AI is enormously powerful but still not perfect. The various models currently in existence have limitations on the amount of content they can process, so you can't simply copy and paste an entire book (or white paper...) into their prompts. The same goes for Codex and similar models when processing code: nice for a single method or class, but not for an entire solution. Still, we're confident that these shortcomings will be addressed over time. And the correctness of the answers will improve as the models expand.

Even if your business doesn't act immediately, the tech giants will (and already are). If you're a Microsoft 365 customer, Copilot will likely become available to you relatively soon. You could choose to opt out and disable it, but doing so will only give your more intrepid competitors the upper hand. The same goes for Copilot: your developers may not use it, but their counterparts at other companies will. Whether the cost justifies the potential increase in efficiency is another story.

The availability of AI as out-of-the-box APIs opens up a whole world of possibilities. Implementing a new service in your Azure environment can be done in minutes. The addition of generative models opens up many additional use cases that were previously impossible. While ethical considerations must be taken into account, this technology has the potential to be a game changer, especially for tedious, repetitive, and easily predictable tasks.



As with any disruptive technology, the pace of change may be faster than you expected. It is important to have the agility to seize opportunities as they arise. While we cannot predict exactly where we will be in five years, it is certain that we will not be in the same place that we are today.

Conclusion

A time when artificial intelligence will start to have a profound impact on our society seems closer than ever, with generative AI and large language models leading the way. GPT-4, Midjourney, and Codex are just the beginning of what's possible with this technology. The IT industry has been quick to respond, with tech giants such as OpenAI, Microsoft, and Google investing heavily in AI offerings. Google's CEO, Sundar Pichai, even put the company on "red alert" because of the potential disruption which ChatGPT could have on its search business, which accounts for no less than 71% of its revenue.

As a business leader, it's up to you to consider how this technological advancement might apply to your organization. Is it a useful tool to make your knowledge workers more efficient? Will AI, including generative AI, play a role in the software your internal development team builds? Or do you see AI-enabled services or products as a golden opportunity in your market? One thing is clear, in our opinion: those who completely ignore or disregard this moment in time will eventually find themselves falling behind. It's time to embrace AI and all that it can offer, while also considering the implications and risks that come with it. The future may be uncertain, but artificial intelligence is here to stay – and it's up to us to harness its power for good.

Contact us

Are you ready to embrace the power of artificial intelligence and take your business to the next level? alfa1 is here to help: with our expertise in software development, we can help you brainstorm and implement cutting-edge solutions that will take your business to new heights. Don't wait until your competitors have already adopted AI technology – [contact alfa1](#) today to start unlocking the power of AI for your business.