



**TRAILHEAD**  
TECHNOLOGY PARTNERS

# Is Everyone AI-ing Without Me?

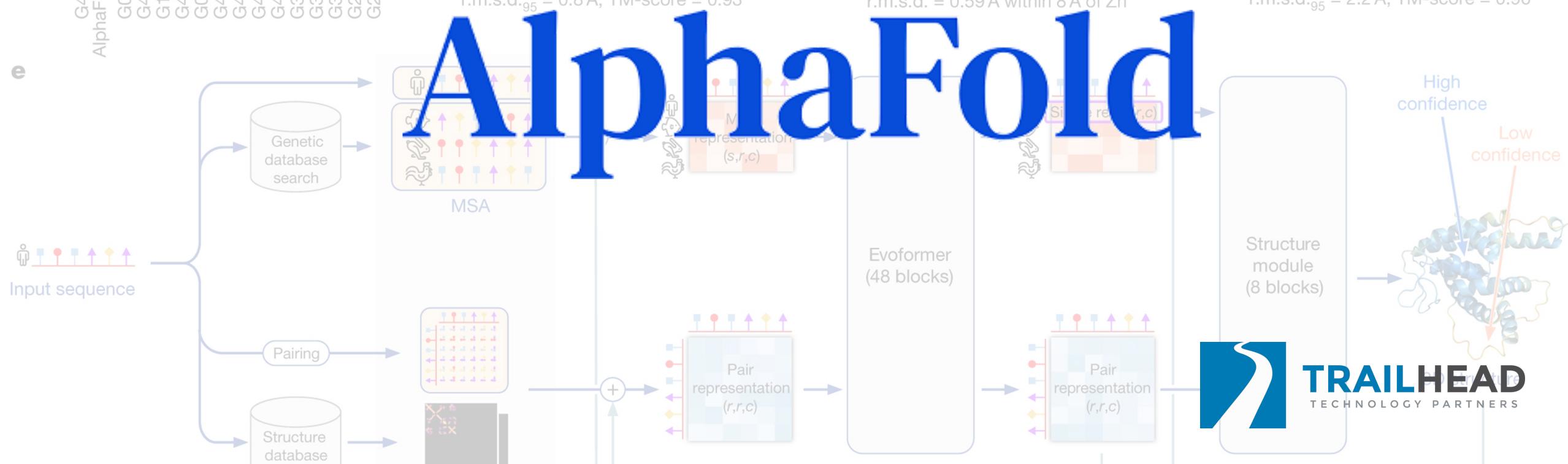
A .NET Developer's Guide to AI



Jonathan "J." Tower

# AI is a Buzzword, But Did You Know...







TESLA

VisionR:  
NL(0.00  
NRP: 0.00  
COP: 0.00  
+0.0001  
+0.0001 IDEAS  
+0.0003 TRAINING  
+0.0002 TIRE SPRAY  
+0.0013 WET ROAD  
0.7000 RESTRICTED  
0.1539 CONTROLLED ACCESS

L:0 R:0 F:2 ON:0  
AP:0.6 ID: -  
4076 MPH St: 1  
merge: 1.0 v: 163.7 R



# VISA



**TRAILHEAD**  
TECHNOLOGY PARTNERS

# AI: Not Just a Buzzword

# **What AI Is When To Use It How To Integrate it Into Your .NET Applications**

# Jonathan "J." Tower

Principal Consultant & Partner

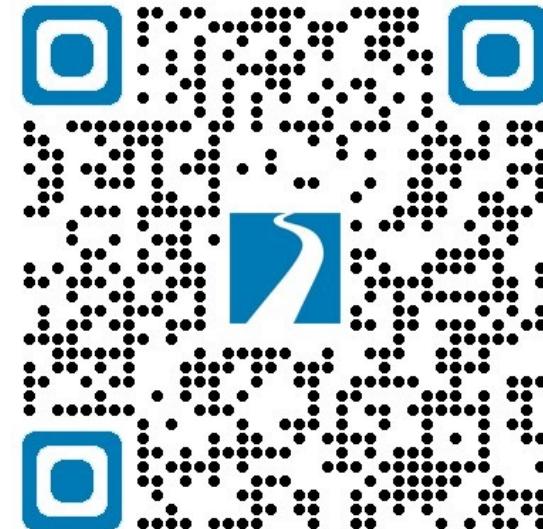


**TRAILHEAD**  
TECHNOLOGY PARTNERS

- 🏆 Microsoft MVP in .NET
- ✉️ jtower@trailheadtechnology.com
- 🌐 trailheadtechnology.com/blog
- 🐦 jtowermi
- linkedin jtower

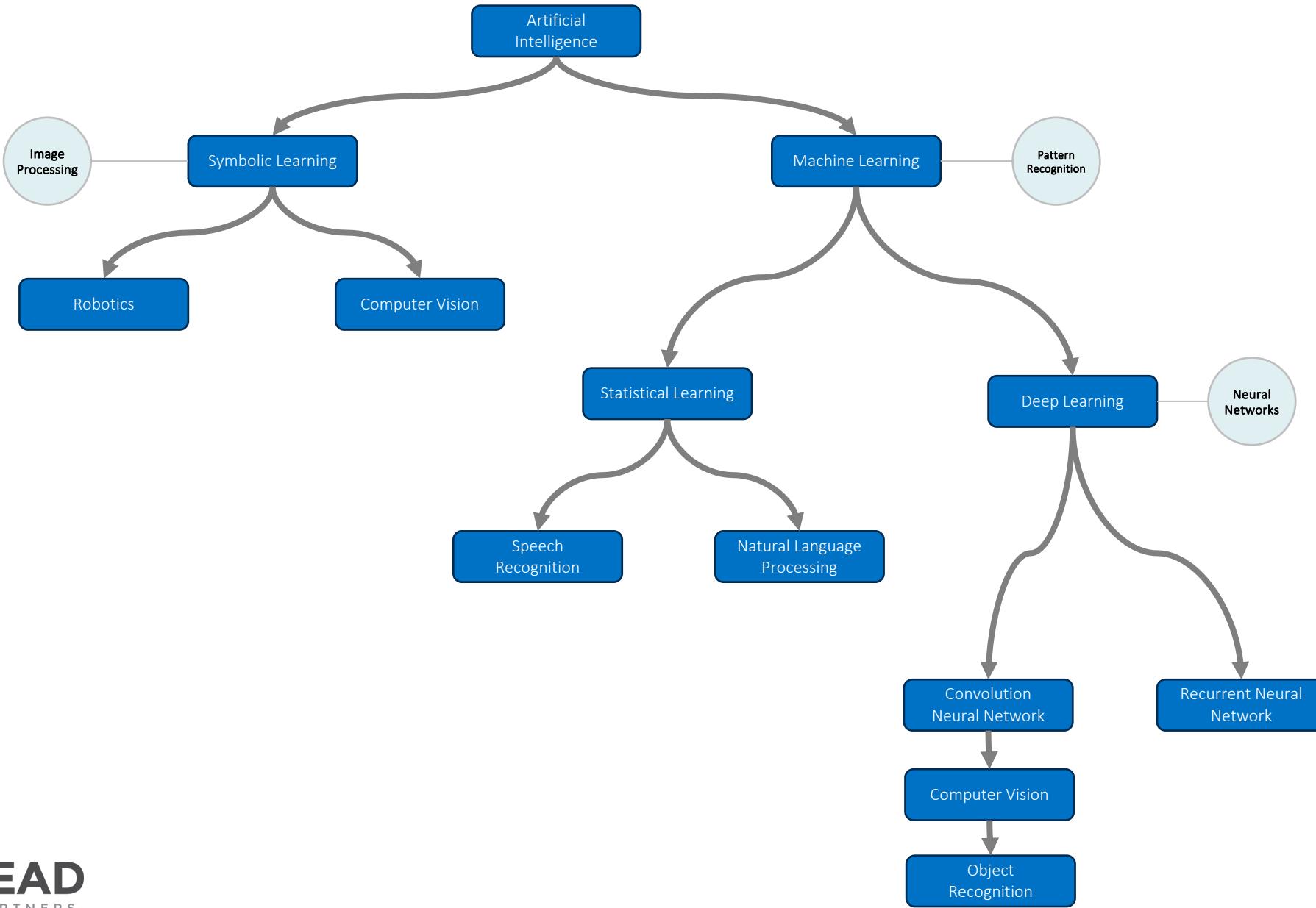
[github.com/trailheadtechnology/ai-for-dotnet](https://github.com/trailheadtechnology/ai-for-dotnet)

**FREE  
CONSULTATION**



[bit.ly/th-offer](https://bit.ly/th-offer)

# Understanding the AI Landscape





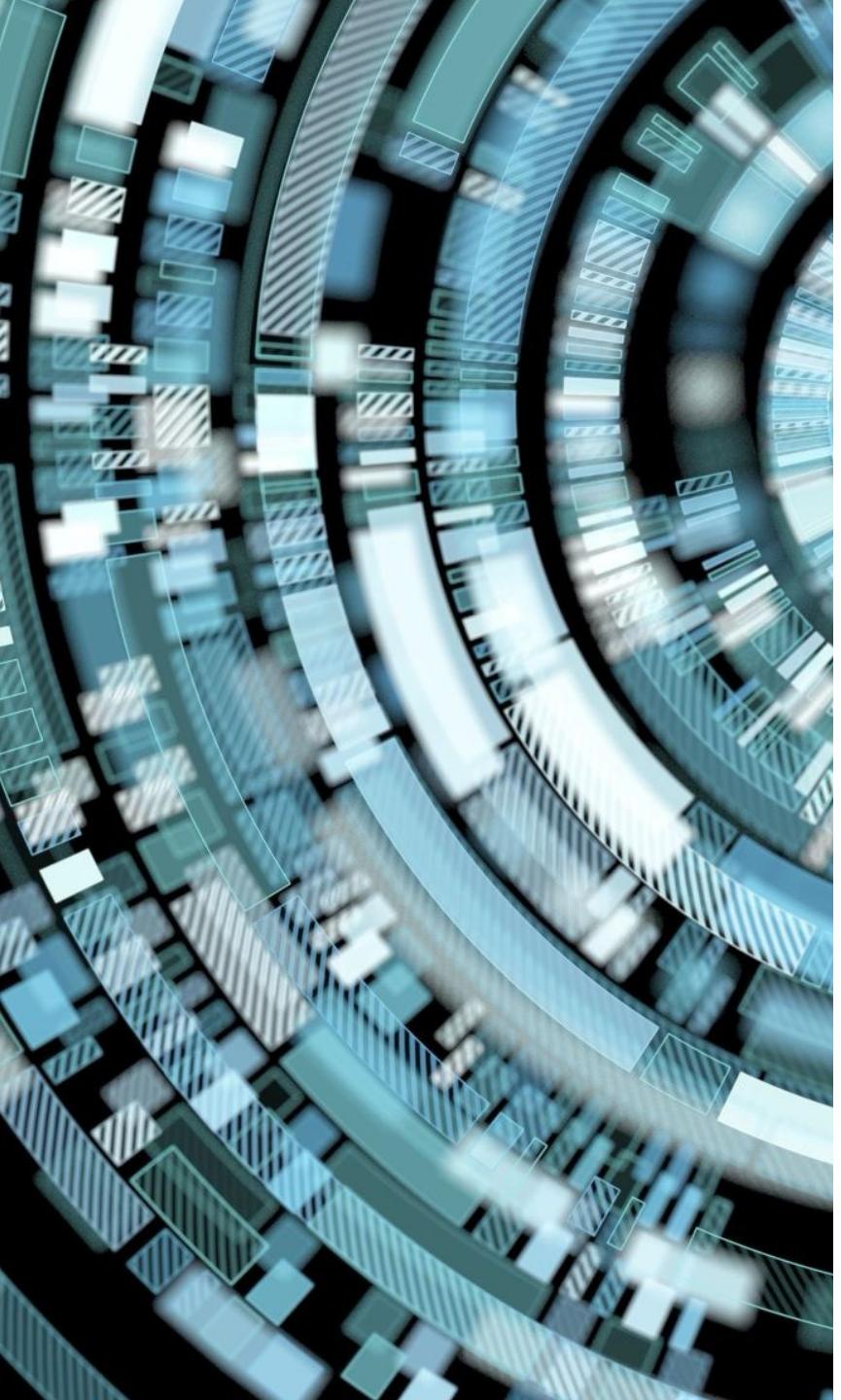
# Machine Learning

A field of study in **artificial intelligence** concerned with the development and study of **statistical algorithms** that can **learn from data** and generalize to unseen data.



# Neural Network

In **machine learning**, a neural network is a model inspired by the **neuronal organization** found in the biological neural networks in **animal brains**.



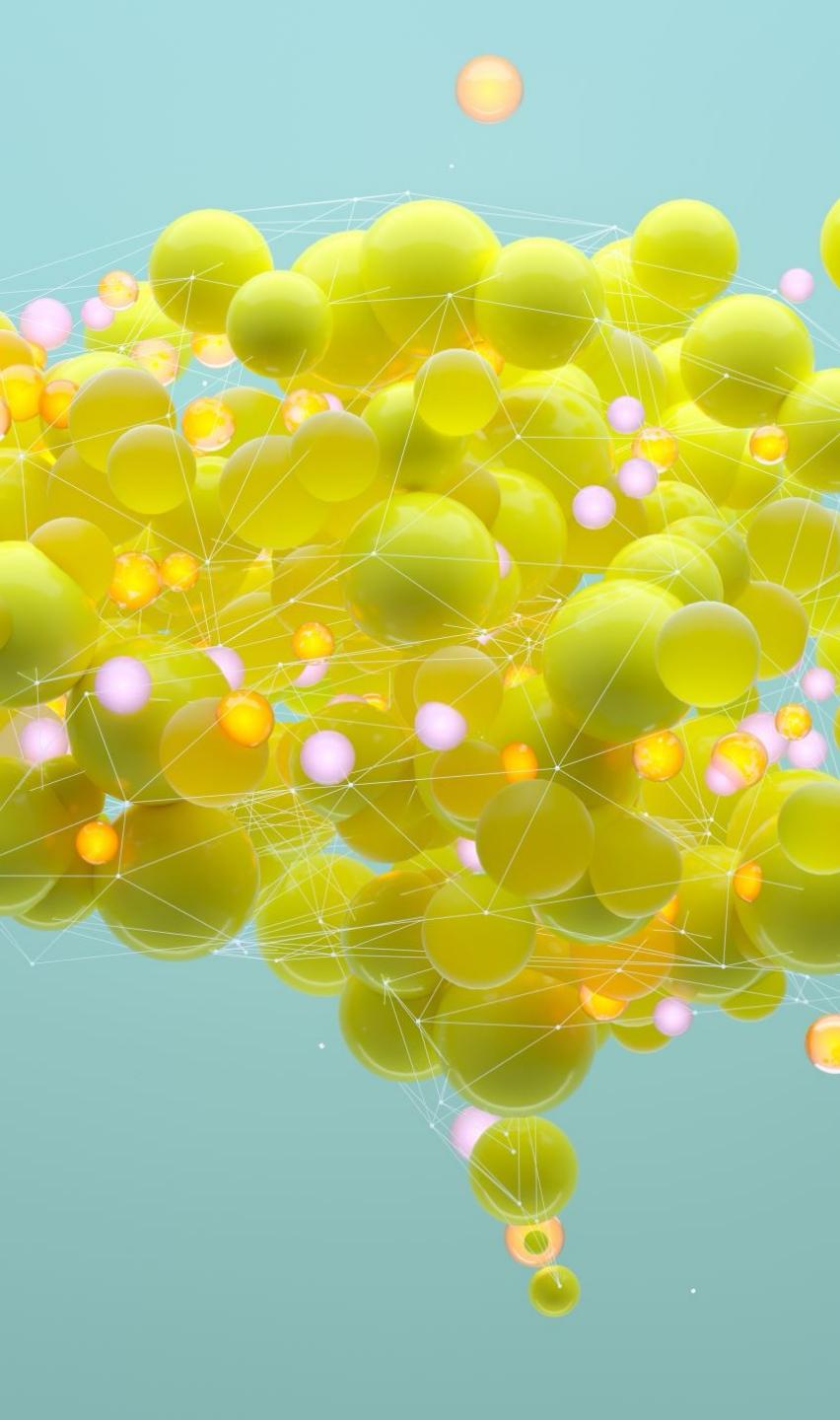
# Deep Learning

Deep learning is the **subset of machine learning** methods based on **artificial neural networks** with **representation learning**. The adjective "deep" refers to the use of **multiple layers** in the network.



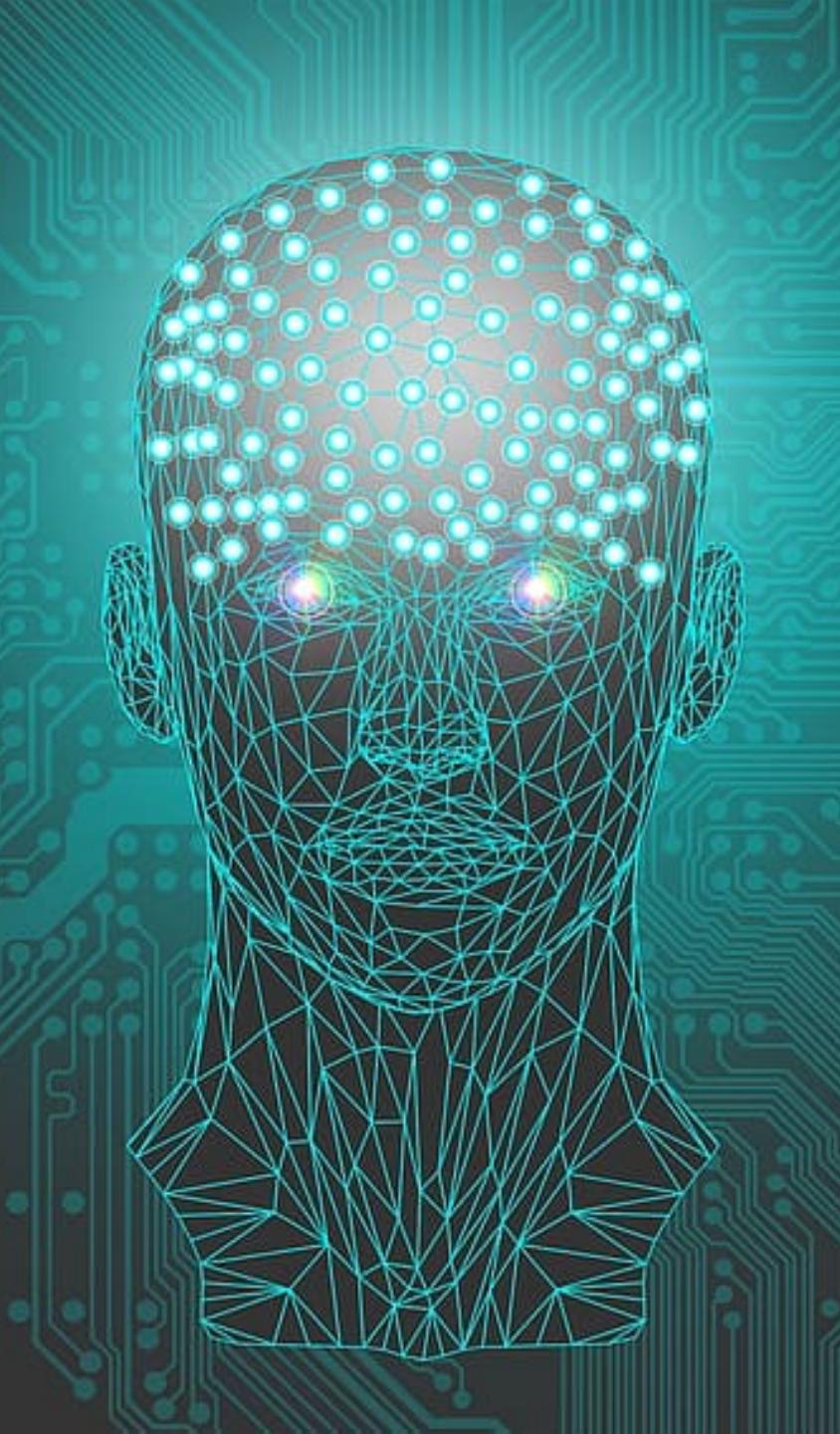
# Generative AI

A **deep learning model** capable of **generating** text, images, videos, or other data using generative models, often in response to prompts. Generative AI models learn the **patterns and structure of their input** training data and then generate new data that has **similar characteristics**.



# Large Language Models

A **large language model** is a **generative AI model** notable for its ability to achieve **general-purpose language** generation and other natural language processing tasks such as classification.



# Generative Pre-trained Transformer (GPT)

A type of **large language model** that **uses deep learning** to generate human-like text **based on the input** it receives.

# Retrieval-Augmented Generation (RAG)

A technique that **combines retrieval** of relevant information **with text generation** in language models to produce contextually informed responses.



**TRAILHEAD**  
TECHNOLOGY PARTNERS

# Data Science

An **interdisciplinary** field that uses **statistics, scientific computing, scientific methods, processes, and algorithms** extract or **extrapolate knowledge and insights** from potentially noisy, structured, or unstructured data.

# When To Use AI vs Traditional Algorithms

# When To Use AI



Image  
Recognition



Text Classification &  
Natural Language  
Processing



Speech Recognition &  
Synthesis



Recommendation  
Systems



Anomaly  
Detection



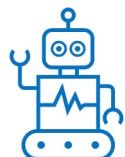
Predictive  
Analytics



Pattern  
Recognition



Content  
Generation



Chat  
Bots



**TRAILHEAD**  
TECHNOLOGY PARTNERS

# When To Use Traditional Algorithms



Rule-based  
Systems



Deterministic  
Rules



Forms  
Over Data



Predictability  
& Transparency



Precision &  
Simplicity

# Hybrid Approach



Going forward, most software will either:

1. Use **BOTH** AI and traditional algorithms, or
2. Use **ONLY** AI

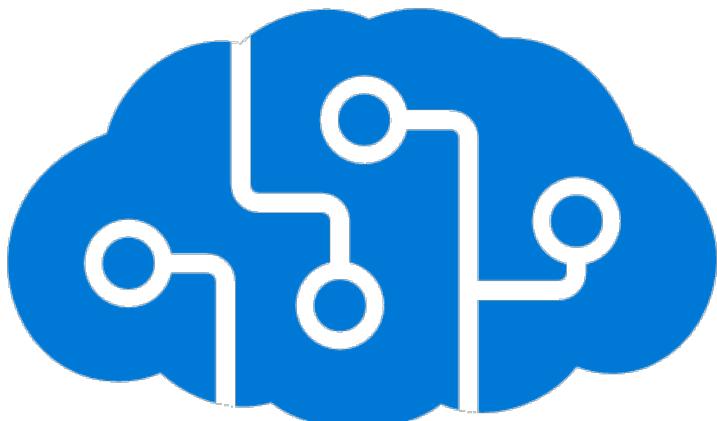
Less and less software will use only traditional algorithms.

# AI Tools & Services for .NET Developers

# Azure AI Service

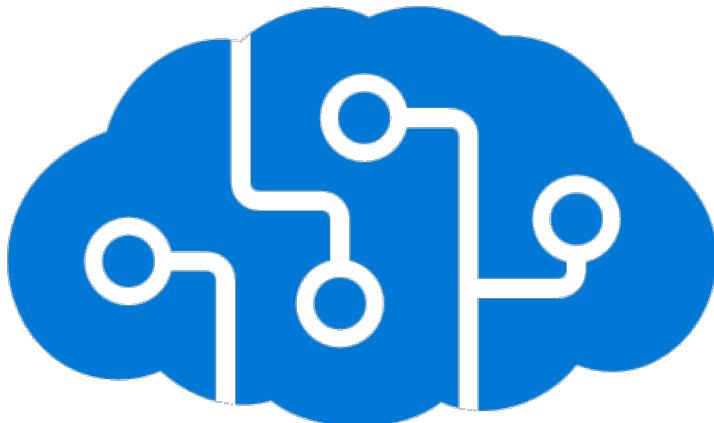
Outsource your AI to the Cloud

# Cognitive Services



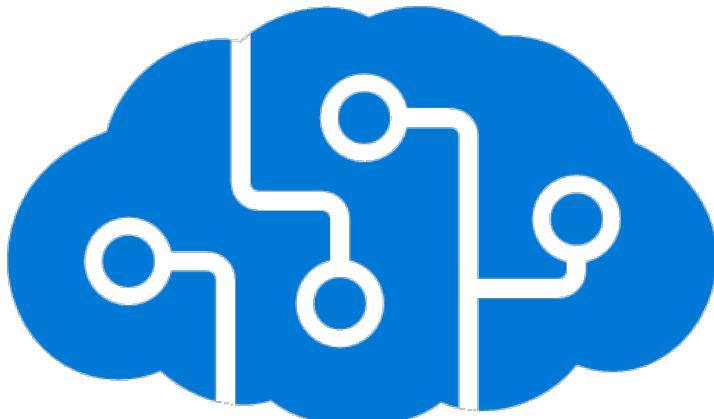
- Anomaly Detection
- Azure AI Search
- Azure OpenAI
- Bot Service
- Content Moderator
- Content Safety
- Custom Vision
- Document Intelligence
- Face
- Immersive Reader
- Language
- Language Understanding
- Metrics Advisor
- Personalizer
- QnA maker
- Speech
- Translator
- Video Indexer
- Vision

# Cognitive Services Azure AI Service



- Anomaly Detection
- Azure AI Search
- Azure OpenAI
- Bot Service
- Content Moderator
- Content Safety
- Custom Vision
- Document Intelligence
- Face
- Immersive Reader
- Language
- Language Understanding
- Metrics Advisor
- Personalizer
- QnA maker
- Speech
- Translator
- Video Indexer
- Vision

# Cognitive Services Azure AI Service



- Anomaly Detection
- Azure AI Search
- **Azure OpenAI**
- **Bot Service**
- Content Moderator
- Content Safety
- Custom Vision
- Document Intelligence
- **Face**
- Immersive Reader
- **Language**
- Language Understanding
- Metrics Advisor
- Personalizer
- QnA maker
- **Speech**
- **Translator**
- Video Indexer
- Vision

# Azure OpenAI Service



Access to OpenAI's generative models

**Azure.AI.OpenAI** package

Compatible with Azure OpenAI and OpenAI APIs

# OpenAI API

```
var endpointUri = new Uri("https://your-azure-openai-resource.com/");  
var apiKey = "your-azure-openai-resource-api-key";  
  
var client = new OpenAIclient(endpointUri, new AzureKeyCredential(apiKey));  
  
var chatCompletion = client.GetChatCompletions("Tell me a joke!");  
  
Console.WriteLine($"ChatGPT says: {chatCompletion.Message}");
```

# Azure OpenAI vs OpenAI

	Azure OpenAI	OpenAI
Doesn't use your data for training	✓	✓
Can opt-out of data retention	✓	✓
Can choose region(s)	✓	✗
Virtual networking	✓	✗
Private link	✓	✗
Access control	✓	✗

# Azure AI Bot Service

A platform that enables developers to build, deploy, and manage intelligent bots capable of natural language understanding and conversation management.



Bot Framework  
Composer

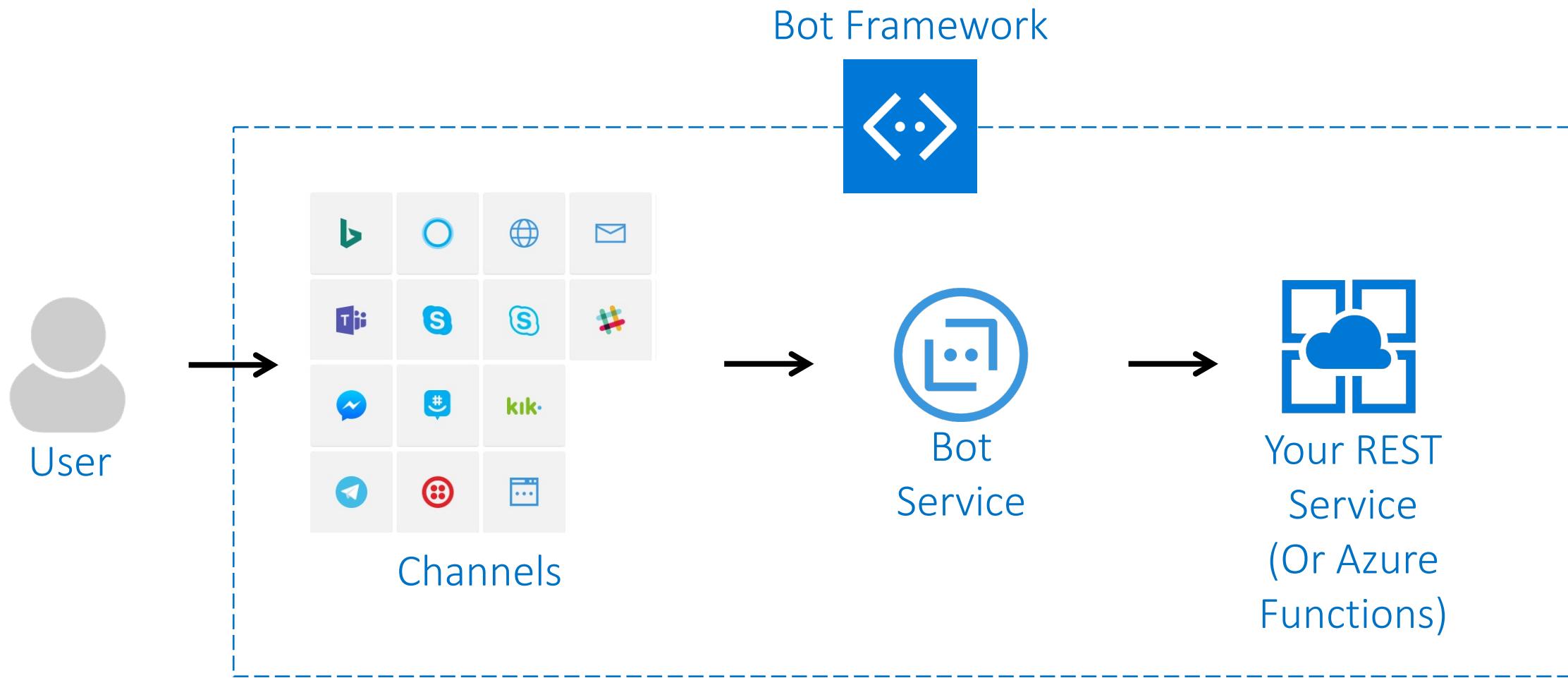


Bot Framework  
SDK



Power Platform  
Virtual Agents

# How Azure Bot Service Works



# Azure AI Bot Service

```
public class MessagesController: ApiController {  
  
    public async Task<Message> Post([FromBody] Message message) {  
        if (message.Type == "Message") {  
            int length = (message.Text ?? string.Empty).Length;  
  
            return message.CreateReplyMessage($"You sent {length} characters");  
        } else {  
            return HandleSystemMessage(message);  
        }  
    }  
  
    private Message HandleSystemMessage(Message message) {  
        ...  
    }  
}
```



# Azure AI Face



Detect



Recognize



Analyze

# Azure AI Face

- Face Detection
- Face Recognition
- Emotion Analysis
- Age and Gender Estimation
- Facial Landmarks
- Face Attributes
- Face Verification
- Face Grouping
- Face Similarity
- Face Identification

# Azure AI Face

```
string subscriptionKey = "YOUR_SUBSCRIPTION_KEY";
string endpoint = "YOUR_FACE_API_ENDPOINT";

IFaceClient faceClient = new FaceClient(new ApiKeyServiceClientCredentials(subscriptionKey)) {
    Endpoint = endpoint
};

// Detect faces in a remote image
string imageUrl = "https://example.com/your-image.jpg";
IList<DetectedFace> detectedFaces = await faceClient.Face.DetectWithUrlAsync(imageUrl);

foreach(var face in detectedFaces) {
    Console.WriteLine($"Face ID: {face.FaceId}");
    Console.WriteLine($"Age: {face.FaceAttributes.Age}");
    Console.WriteLine($"Gender: {face.FaceAttributes.Gender}");
    Console.WriteLine();
}
```



# Azure AI Language



Build apps with industry-leading natural language understanding capabilities

## Merges:

- Text Analytics
- QnA Maker
- Language Understanding (LUIS)

# Azure AI Language



Named Entity  
Recognition (NER)



Personally Identifiable  
Information (PII)  
Detection



Language Detection



Sentiment Analysis and  
Opinion Mining



Summarization

# Azure AI Language

```
string endpoint = "YOUR_TEXT_ANALYTICS_ENDPOINT";
string apiKey = "YOUR_TEXT_ANALYTICS_API_KEY";
var client = new TextAnalyticsClient(new Uri(endpoint), new AzureKeyCredential(apiKey));

var sentimentResults = await client.AnalyzeSentimentBatchAsync(reviews);

foreach(var result in sentimentResults.Value) {
    if (result.Sentiment == TextSentiment.Positive) positiveCount++;
    else if (TextSentiment.Negative) negativeCount++;
    else if (TextSentiment.Neutral) neutralCount++;
}

Console.WriteLine($"Positive: {positiveCount}, Negative: {negativeCount}, Neutral: {neutralCount}");
```



# Azure AI Speech



Provides advanced **speech recognition** and **synthesis** capabilities, enabling developers to integrate speech processing into applications for converting **spoken language into text** and synthesizing **speech from text**.

# Azure AI Speech

```
var config = SpeechConfig.FromSubscription(  
    "SUBSCRIPTION_KEY",  
    "SERVICE_REGION");  
  
config.SpeechSynthesisVoiceName = "en-US-JennyNeural";  
  
using(var audioConfig = AudioConfig.FromDefaultSpeakerOutput())  
{  
    using(var synthesizer = new SpeechSynthesizer(config, audioConfig))  
    {  
        await synthesizer.SpeakTextAsync("Hello, world.");  
    }  
}
```





You don't have any files saved yet.

Speech Studio > Audio Content Creation > My files > Untitled \*

File Save Export Template Auto pre...



00:00



00:09



24kHz

1. [Jenny] Hello, world
2. [Jenny] <Cheerful> I'm feeling great </>
3. [Jenny] <Terrified> Oh, no! </>
4. [Jenny] <Whispering> I have a secret </>

Total characters: 54 / 3,000

Billable characters: N/A ⓘ

Tuning

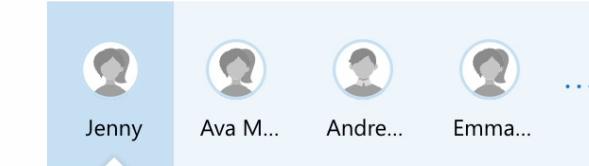
Comment

Recent

Voice ⓘ

Language

English (United States)



Jenny



English (United States)

15 styles Public voice

Speaking style Default

Break ⓘ

Standard

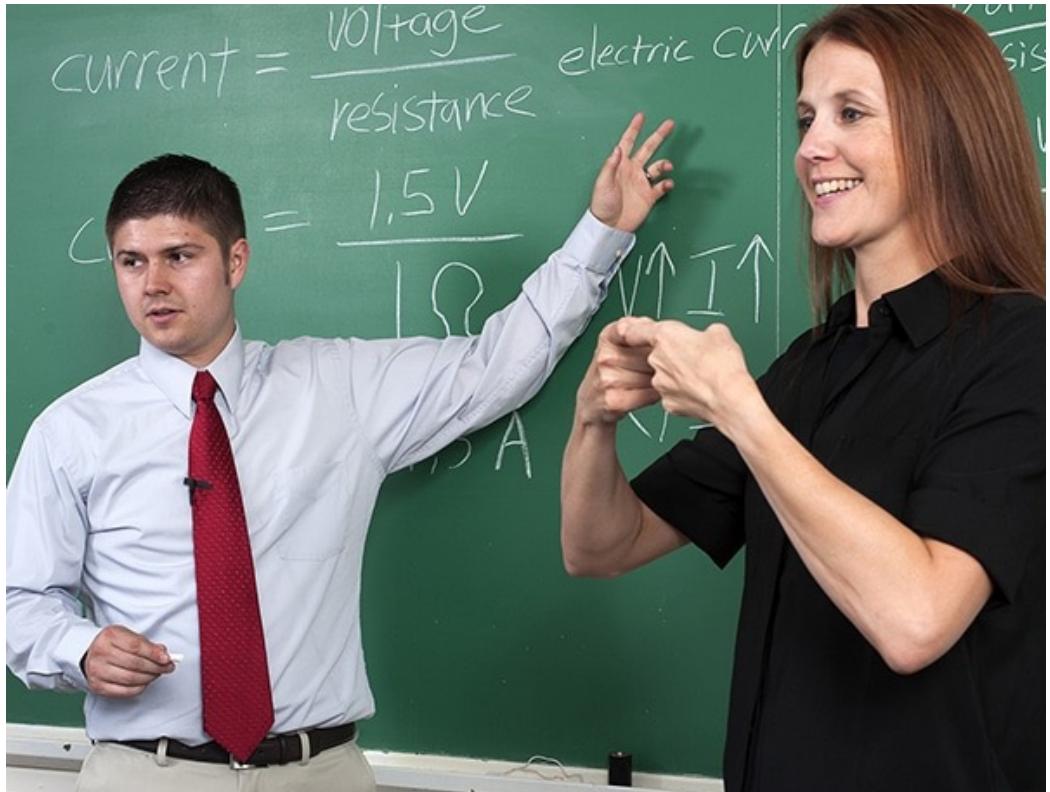
Advanced

Custom

Default

Silence ⓘ

# Azure AI Translator



**Translate** text instantly or in batches across more than **100 languages**. Support a wide range of use cases, such as translation for **call centers**, multilingual **conversational agents**, or **in-app communication**.

# Azure AI Translator

```
string endpoint = "YOUR_TEXT_ANALYTICS_ENDPOINT";
string apiKey = "YOUR_TEXT_ANALYTICS_API_KEY";
var client = new TextAnalyticsClient(new Uri(endpoint), new AzureKeyCredential(apiKey));

Uri sourceSasUri = new Uri("YOUR_SOURCE_SAS_URI");
Uri frenchTargetSasUri = new Uri("YOUR_FRENCH_TARGET_SAS_URI");

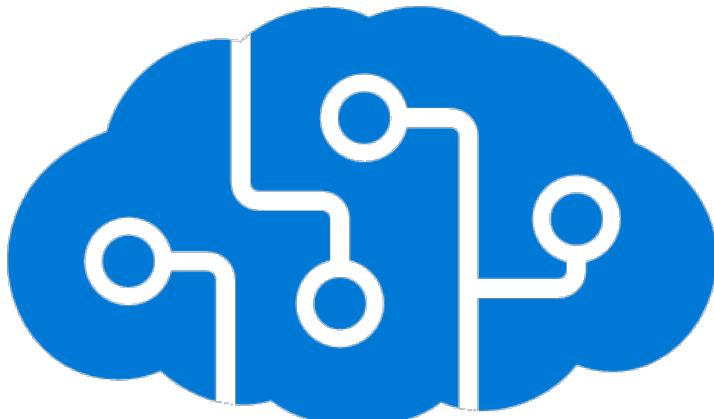
var input = new DocumentTranslationInput(sourceSasUri, frenchTargetSasUri, "fr"); // Translate to French
DocumentTranslationOperation operation = await client.StartTranslationAsync(input);

await operation.WaitForCompletionAsync();

BlobClient targetBlobClient = new BlobClient(frenchTargetSasUri);
using(MemoryStream memoryStream = new MemoryStream()) {
    await targetBlobClient.DownloadToAsync(memoryStream);
    string translatedText = System.Text.Encoding.UTF8.GetString(memoryStream.ToArray());
    Console.WriteLine("Translated Document:");
    Console.WriteLine(translatedText);
}
```



# Cognitive Services Azure AI Service



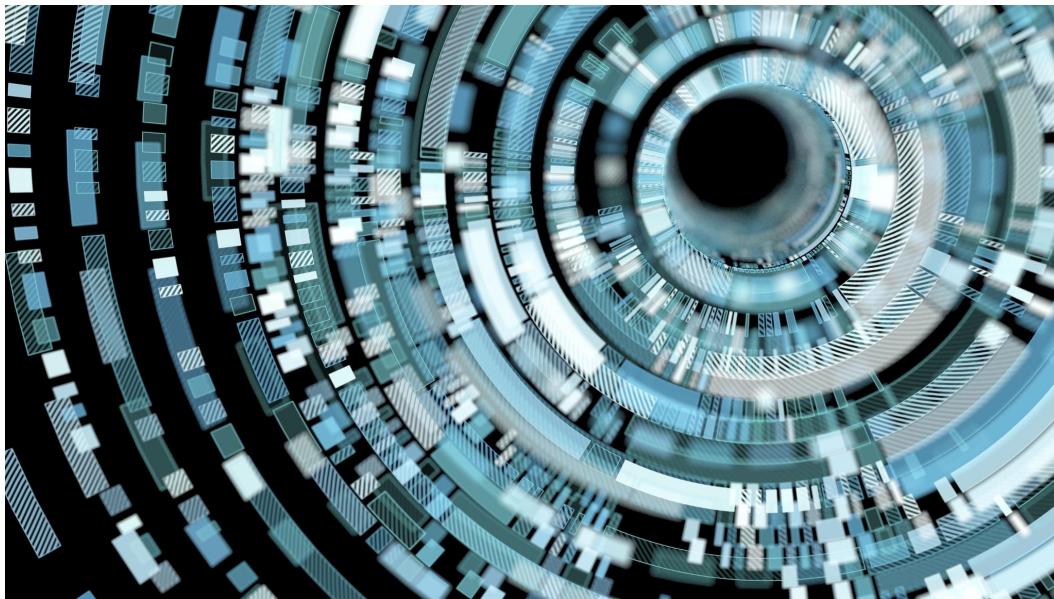
- Anomaly Detection
- Azure AI Search
- **Azure OpenAI**
- **Bot Service**
- Content Moderator
- Content Safety
- Custom Vision
- Document Intelligence
- **Face**
- Immersive Reader
- **Language**
- Language Understanding
- Metrics Advisor
- Personalizer
- QnA maker
- Speech
- **Translator**
- Video Indexer
- Vision

# Azure Machine Learning

ML in the Cloud



# Azure Machine Learning



A comprehensive cloud-based service that enables you to build, deploy, and manage machine learning models at scale.

# Azure Machine Learning



Azure Machine  
Learning Studio UI



AutoML



Model Catalog for  
Sharing and Versioning



Managed Endpoints



Responsible AI tools

# Azure Machine Learning

```
using (var client = new HttpClient()) {
    client.DefaultRequestHeaders.Add("Authorization", $"Bearer <YOUR_API_KEY>");

    var data = new StringContent(JsonSerializer.Serialize(new InputData {
        Temperature = 25.0f,
        Humidity = 60.0f
    }));

    var response = await client.PostAsync(
        "<YOUR_ENDPOINT>",
        data,
        System.Text.Encoding.UTF8,
        "application/json"));

    if (response.IsSuccessStatusCode)
        return JsonSerializer.Deserialize<float>(await response.Content.ReadAsStringAsync());

    throw new Exception($"Prediction request failed with status code {response.StatusCode}");
}
```



# ML.NET

ML in .NET

# ML.NET

ML.NET is a free, open-source, and cross-platform machine learning framework for the .NET developer platform.

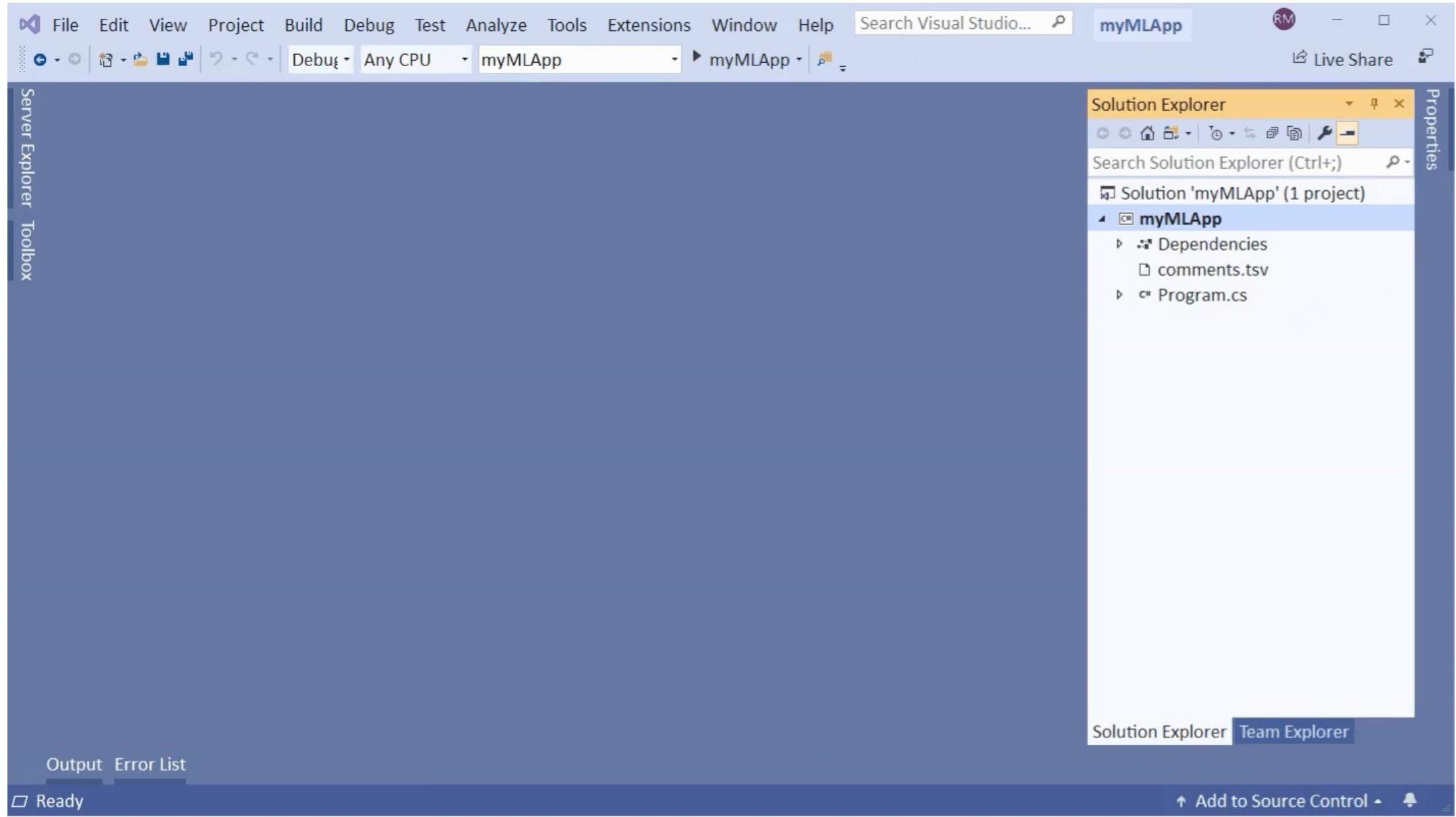
## **Allow you to:**

- Train models
- Build models
- Ship models

## **Key Features:**

- AutoML
- ML.NET CLI
- ML.NET Model Builder

# ML.NET Model Builder



# ML.NET

```
var mlContext = new MLContext();
var model = mlContext.Model.Load("model.zip", out var modelSchema);

// Create a prediction engine
var predictionEngine = mlContext.Model.CreatePredictionEngine<InputData, OutputData>(model);

// Define input data
var inputData = new InputData {
    Temperature = 25.0f, Humidity = 60.0f
};

// Make predictions
var outputData = predictionEngine.Predict(inputData);

// Display the prediction result
Console.WriteLine($"Predicted rainfall (mm): {outputData.Prediction}");
```

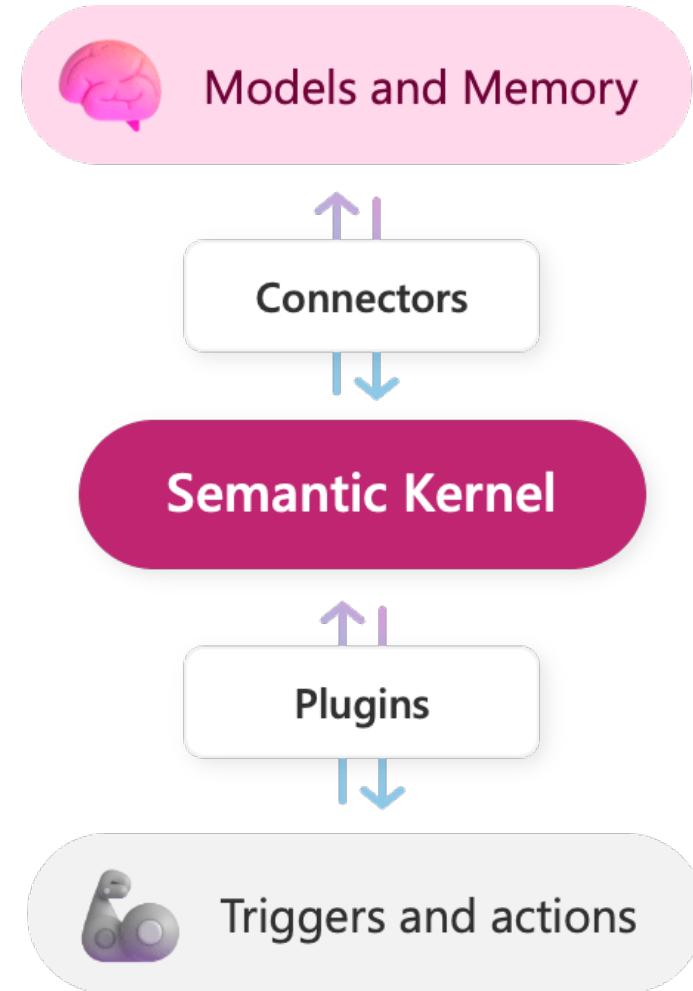


# Semantic Kernel

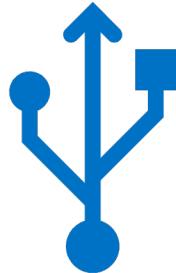
Orchestrates all Microsoft's Copilots

# Semantic Kernel

Semantic Kernel is an open-source SDK developed by Microsoft that allows you to seamlessly integrate cutting-edge Large Language Models (LLMs) like OpenAI, Azure OpenAI, and Hugging Face with conventional programming languages like C#, Python, and Java.



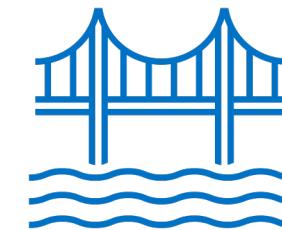
# Semantic Kernel



Plugins



Planners



Connectors

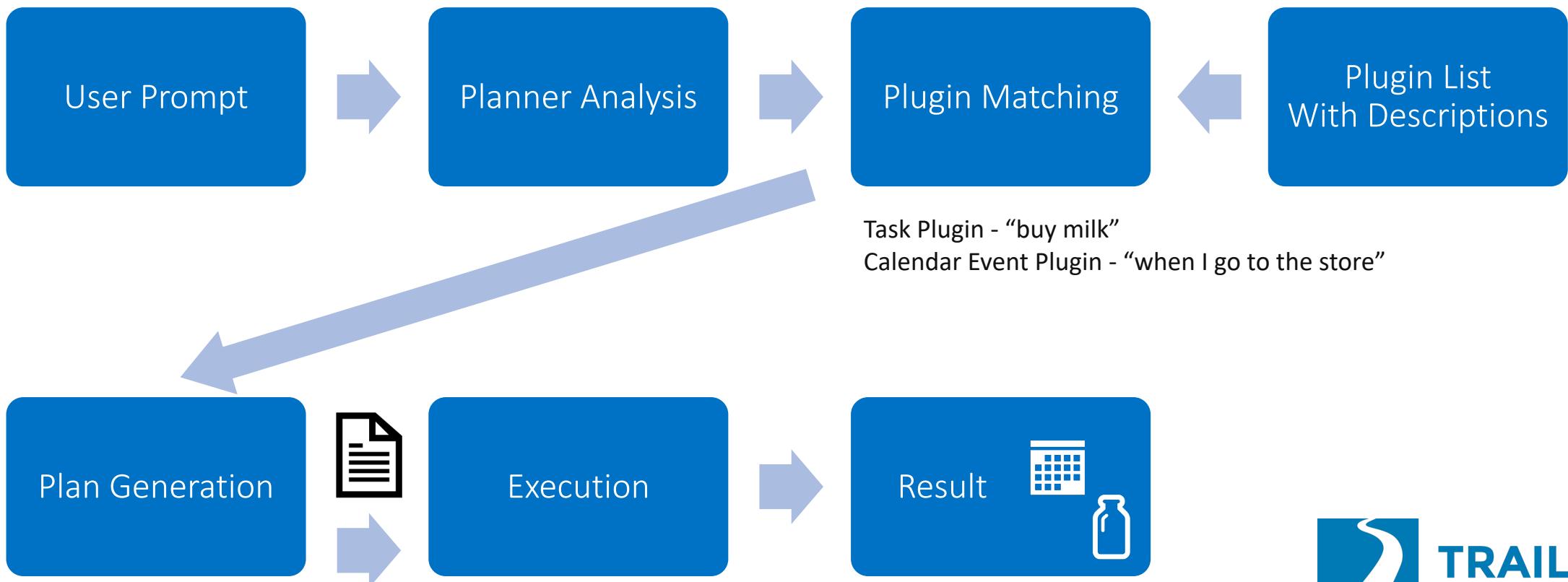
# Semantic Kernel



# Semantic Kernel

"Remind me to buy milk when I go to the store."

"buy milk" (task)  
"when I go to the store" (calendar event)



[AVAILABLE FUNCTIONS]

```
TaskPlugin.CreateTask
Description: Add two numbers
Inputs:
- task string - The task to be described (required)
Output: string
```

```
CalendarEventPlugin.CreateEvent
Description: Creates calendar event
...
```

Plugin List With Descriptions

# Semantic Kernel

```
dotnet add package Microsoft.SemanticKernel
```



# Semantic Kernel

```
var config = new ConfigurationBuilder()
    .AddJsonFile("appsettings.json")
    .Build();

var builder = Kernel.CreateBuilder();
builder.Services.AddAzureOpenAIChatCompletion(
    "DEPLOYMENT_MODEL",
    "AZURE_OPENAI_ENDPOINT",
    "AZURE_OPENAI_KEY");

var kernel = builder.Build();

var result = await kernel.InvoicePromptAsync("What are the best hikes in Utah?");

Console.WriteLine(result);
```



# Semantic Kernel

```
var config = new ConfigurationBuilder()
    .AddJsonFile("appsettings.json")
    .Build();

var builder = Kernel.CreateBuilder();
builder.Services.AddAzureOpenAIChatCompletion(
    "DEPLOYMENT_MODEL",
    "AZURE_OPENAI_ENDPOINT",
    "AZURE_OPENAI_KEY");

builder.Plugins.AddFromType<MyPluginClass>();

var kernel = builder.Build();

var result = await kernel.InvoicePromptAsync("What are the best hikes in Utah?");

Console.WriteLine(result);
```



# Semantic Kernel

```
public class LightPlugin
{
    public bool IsOn { get; set; } = false;

    [KernelFunction]
    [Description("Gets the state of the light.")]
    public string GetState() => IsOn ? "on" : "off";

    [KernelFunction]
    [Description("Changes the state of the light.'")]
    public string ChangeState([Description("The new state of the light.")] bool newState)
    {
        this.IsOn = newState;
        var state = GetState();
        return state;
    }
}
```



# Cognitive Toolkit

todo

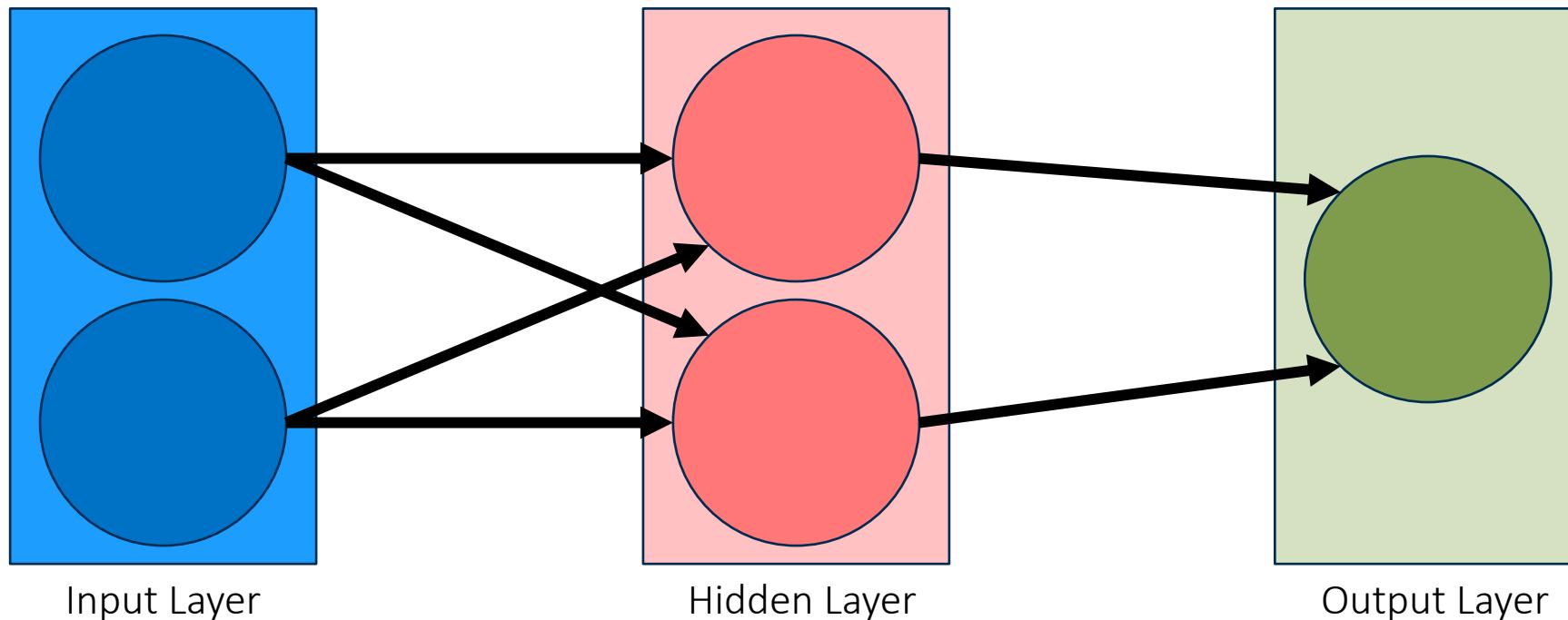
# Microsoft Cognitive Toolkit (CNTK)



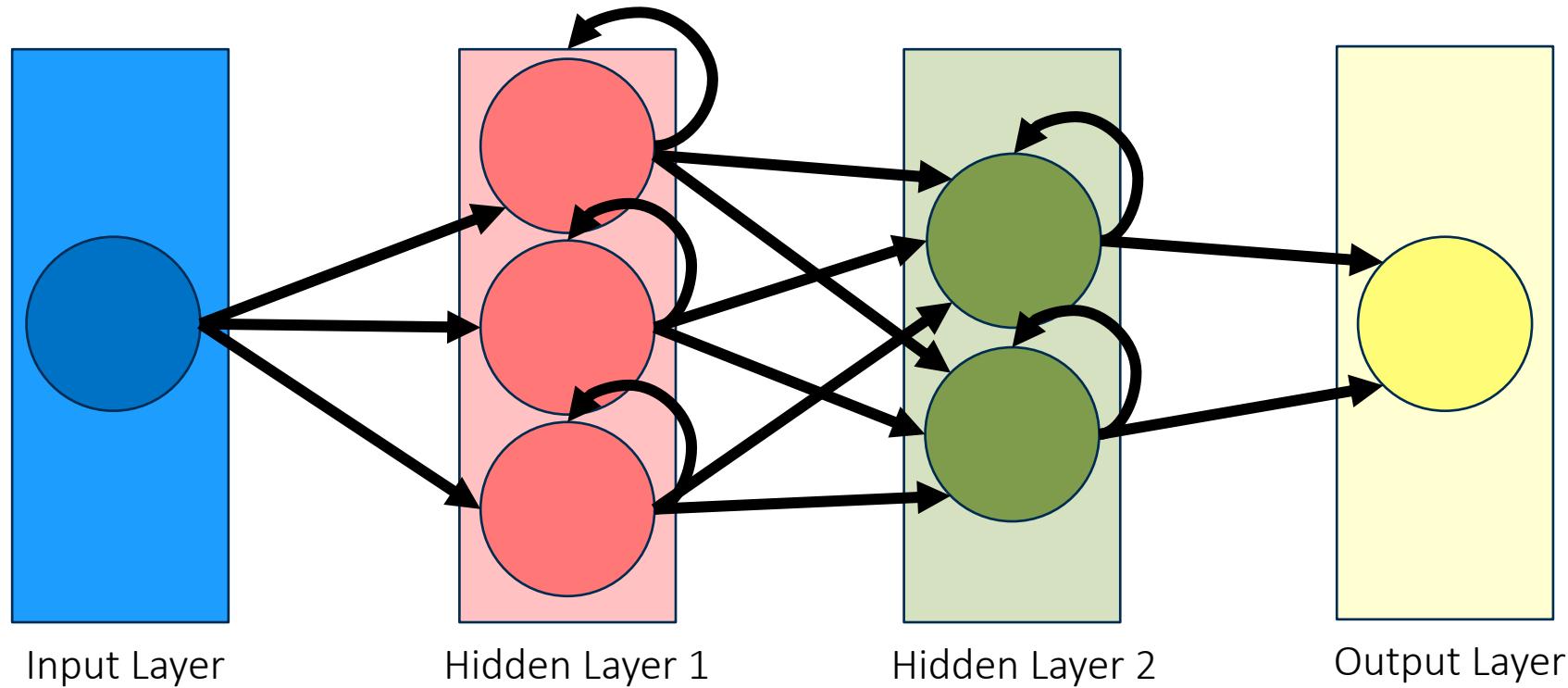
An open-source toolkit for commercial-grade deep learning

- Feed-forward DNNs
- Convolutional neural networks (CNNs)
- Recurrent neural networks (RNNs/LSTMs)

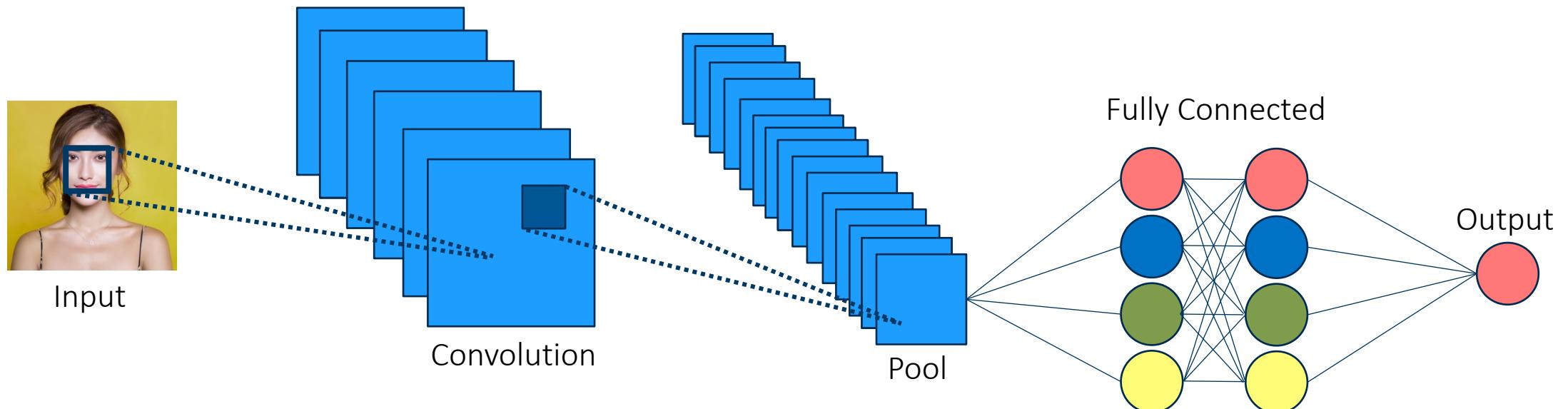
# Feed Forward Neural Network (FFNN)



# Recurrent Neural Network (RNN)



# Convolutional Neural Network (CNN)



# Microsoft Cognitive Toolkit (CNTK)

```
var convParams1 = new Parameter(
    new int[] { kernelWidth1, kernelHeight1, numInputChannels, outFeatureMapCount1 },
    DataType.Float, CNTKLib.GlorotUniformInitializer(convWScale, -1, 2), device);
var convFunction1 = CNTKLib.ReLU(CNTKLib.Convolution(
    convParams1, input,
    new int[] { 1, 1, numInputChannels } ));
var pooling1 = CNTKLib.Pooling(convFunction1, PoolingType.Max,
    new int[] { poolingWindowWidth1, poolingWindowHeight1 }, new int[] { hStride1, vStride1 }, new bool[] { true });
var convParams2 = new Parameter(
    new int[] { kernelWidth2, kernelHeight2, outFeatureMapCount1, outFeatureMapCount2 },
    DataType.Float, CNTKLib.GlorotUniformInitializer(convWScale, -1, 2), device);
var convFunction2 = CNTKLib.ReLU(CNTKLib.Convolution(
    convParams2, pooling1,
    new int[] { 1, 1, outFeatureMapCount1 } ));
var pooling2 = CNTKLib.Pooling(convFunction2, PoolingType.Max,
    new int[] { poolingWindowWidth2, poolingWindowHeight2 }, new int[] { hStride2, vStride2 }, new bool[] { true });
var imageClassifier = TestHelper.Dense(pooling2, numClasses, device, Activation.None, "ImageClassifier");
```

## Some Others Worth a Look

TensorFlow.NET

Accord.NET

ONNX

# Challenges & Considerations

# Challenges & Considerations



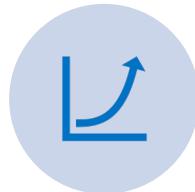
Data Quality &  
Quantity



Ethical Concerns



Black Box Nature



Scalability & Resource  
Requirements



Maintenance &  
Monitoring

# Summing Up

1. AI is important, can be integrated with traditional algorithms
2. AI is only good for some types of problems
3. Use Azure AI for packages services in cloud
4. Use Azure ML for MLaaS
5. Use ML.NET for .NET-hosted ML
6. Use Semantic Kernel for your own copilots
7. Use Cognitive Toolkit for building own neural networks
8. AI has some drawbacks to consider



# Thanks! Questions?

## Jonathan "J." Tower

🏆 Microsoft MVP in .NET

✉️ jtower@trailheadtechnology.com

🌐 trailheadtechnology.com/blog

🐦 jtowermi

linkedin jtower

# Free Consultation



[bit.ly/th-offer](http://bit.ly/th-offer)

[github.com/trailheadtechnology/ai-for-dotnet](https://github.com/trailheadtechnology/ai-for-dotnet)