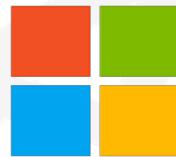


AI at the EDGE con .NET: cosa potrebbe andare storto?

Marco Dal Pino



Platinum Sponsor



Microsoft



Gold Sponsor



Technical Sponsor



AI on the EDGE... Why?



SECURITY



LATENCY



DATA
RESIDENCY



SPECIFIC
SCENARIOS



Just Play with AI on your PC



Usiamo la AI con .NET

Step 1: Setting Up the Development Environment

- Install .NET SDK:** Make sure you have the .NET SDK installed on your machine. You can download it from the [official .NET website](#).
- Install ML.NET:** ML.NET is a machine learning framework for .NET. Install it by adding the Microsoft.ML NuGet package to your project:

```
Bash
```

```
Copy
```

```
dotnet add package Microsoft.ML
```

Step 2: Create a New .NET Project

- Create a Console App:** Create a new .NET console application:

```
Bash
```

```
Copy
```

```
dotnet new console -n EdgeAIApp  
cd EdgeAIApp
```

Step 3: Build and Train Your AI Model

- Define Data Model:** Create a class to define the data schema for your model:

```
Csharp
```

```
Copy
```

```
public class ModelInput  
{  
    public float Feature1 { get; set; }  
    public float Feature2 { get; set; }  
    public string Label { get; set; }  
}
```

- Load Data:** Load your training data into the model:

```
Csharp
```

```
Copy
```

```
var mlContext = new MLContext();  
var data = mlContext.Data.LoadFromTextFile<ModelInput>("data
```

- Build and Train the Model:** Define and train the model:

```
Csharp
```

```
Copy
```

```
var pipeline = mlContext.Transforms.Conversion.MapValueToKey  
.Append(mlContext.Transforms.Concatenate("Fe  
.Append(mlContext.MulticlassClassification.T  
.Append(mlContext.Transforms.Conversion.MapK  
  
var model = pipeline.Fit(data);
```



Cont...

Step 4: Export the Model

1. **Save the Model:** Save the trained model to a file:

```
Csharp Copy  
mlContext.Model.Save(model, data.Schema, "model.zip");
```

Step 5: Deploying to Edge Device

1. **Install .NET on Edge Device:** Ensure .NET is installed on your edge device.
2. **Copy Model to Edge Device:** Transfer the `model.zip` file to your edge device.
3. **Run the Model:** Use a .NET application on the edge device to load and use the model:

```
Csharp Copy  
  
var loadedModel = mlContext.Model.Load("model.zip", out var  
var predictionEngine = mlContext.Model.CreatePredictionEngin  
  
var input = new ModelInput { Feature1 = 1.2f, Feature2 = 3.4  
var prediction = predictionEngine.Predict(input);  
Console.WriteLine($"Predicted label: {prediction.PredictedLa
```

Step 6: Testing and Optimization

1. **Test Model:** Test the model on the edge device with real-world data to ensure it works as expected.
2. **Optimize:** Make necessary adjustments to optimize performance and accuracy for your specific use case.

Step 7: Monitoring and Maintenance

1. **Monitor Performance:** Continuously monitor the model's performance and make updates as needed.
2. **Regular Updates:** Periodically update the model with new data and retrain it to maintain accuracy.

By following these steps, you'll be able to set up an AI on the edge project with .NET successfully. Let me know if you need any more details or if there's anything else I can help with!



Microsoft.Extensions.AI

Step 1: Install the Package

First, you need to install the Microsoft.Extensions.AI package:

Bash

Copy

```
dotnet add package Microsoft.Extensions.AI --prerelease
```

Step 2: Implement AI Services

You can implement various AI services using the abstractions provided by the library [1](#). For example, to use a chat client, you can implement the `IChatClient` interface:

Step 3: Use the AI Service

Once you have implemented the AI service, you can use it in your .NET application:



- 30+ years in IT (Developer, Architect, Consultant, PM, Trainer)
- Speaker, Community addicted
- IoT Influencer
- Microsoft Certified Trainer

Marco Dal Pino
Technical Consultant
Microsoft



 <https://www.linkedin.com/in/marcodalpino>

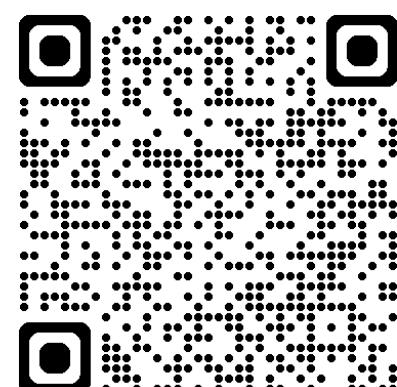
 <https://about.me/marcodalpino>

 <https://twitter.com/marcodalpino>

 info@contoso.blog

 <https://www.twitch.tv/dpcons>

<https://www.twitch.tv/techchat>





Feedback, Feedback, Feedback