

# Programowanie usług w chmurze

## Sprawozdanie z laboratorium 2 i 3

Bartosz Sieracki 304480

Ze względu na to, że ciężko przechodziłem covid od dnia po wstawieniu treści ćwiczenia drugiego do czasu zamknięcia terminu oddania go, zamieszczam też tutaj sprawozdanie z ćwiczenia drugiego.

### Ćwiczenie: Integracja z Azure Bing Search Service

Celem tego ćwiczenia jest zapoznanie się z usługą Azure Bing Search i praktyczne zastosowanie jej w aplikacji. Będziesz tworzyć aplikację, która wykorzystuje Bing Search API do przeprowadzania złożonych zapytań wyszukiwania i analizy wyników.

Krok 1: Zainstalowanie środowiska pracy.

Przygotowane wcześniej środowisko .NET do tworzenia aplikacji w języku C#.

Krok 2: Utworzenie zasobu Azure Cognitive Services

Poniżej podano wszystkie wybrane opcje.

Create Azure AI services

Podstawowe informacje Sieć Identity Tagi Przejdź i utwórz

[Pobierz szablon do automatyzacji](#)

POSTANOWIENIA

Klikając przycisk „Utwórz”: (a) akceptuję postanowienia prawne i oświadczenie o ochronie prywatności skierowane z wymienionymi powyżej ofertami w trybie Marketplace; (b) upoważniam firmę Microsoft do naliczania opłat za korzystanie z ofert (zgodnie z obecnie wybraną formą płatności) z taką samą częstotliwością rozliczeń jak w przypadku subskrypcji platformy Azure; (c) zgadzam się na udostępnianie przez firmę Microsoft moich informacji kontaktowych oraz informacji dotyczących użycia i transakcji dostawcom tych ofert na potrzeby pomocy technicznej, rozliczeń i innych działań związanych z transakcjami. Firma Microsoft nie przyznaje praw do ofert innych firm. Więcej szczegółów można znaleźć na stronie [Warunki korzystania z usługi Azure Marketplace](#).

**Podstawowe informacje**

Subskrypcja	Azure for Students
Grupa zasobów	uslugi_chmurowe
Region	North Europe
Nazwa	CognitiveServices967
Warianta cenowa	Standard S0

**Sieć**

Typ	Dostęp do tego zasobu można uzyskiwać ze wszystkich sieci, w tym internetu.
-----	-----------------------------------------------------------------------------

**Identity**

Typ tożsamości	None
----------------	------

### Krok 3: Konfiguracja Bing Search w Azure Cognitive Services

Utwórz zasób wyszukiwania Bing

Podstawowe Tagi Przejdź i utwórz

[Pobierz szablon do automatyzacji](#)

POSTANOWIENIA

Klikając przycisk „Utwórz”: (a) akceptuję postanowienia prawne i oświadczenie o ochronie prywatności skierowane z wymienionymi powyżej ofertami w trybie Marketplace; (b) upoważniam firmę Microsoft do naliczania opłat za korzystanie z ofert (zgodnie z obecnie wybraną formą płatności) z taką samą częstotliwością rozliczeń jak w przypadku subskrypcji platformy Azure; (c) zgadzam się na udostępnianie przez firmę Microsoft moich informacji kontaktowych oraz informacji dotyczących użycia i transakcji dostawcom tych ofert na potrzeby pomocy technicznej, rozliczeń i innych działań związanych z transakcjami. Firma Microsoft nie przyznaje praw do ofert innych firm. Więcej szczegółów można znaleźć na stronie [Warunki korzystania z usługi Azure Marketplace](#).

**Podstawowe**

Subskrypcja	Azure for Students
Grupa zasobów	uslugi_chmurowe
Nazwa	bing_search
Lokalizacja grupy zasobów	East US
Warianta cenowa	F1 (1 Calls per second, 1k Calls per month)

**Zarządzanie zasobami**

- Klucze i punkt końcowy
- Warianta cenowa
- Sieć
- Tożsamość
- Analiza kosztów
- Własności
- Blockady

**Klucze i punkt końcowy**

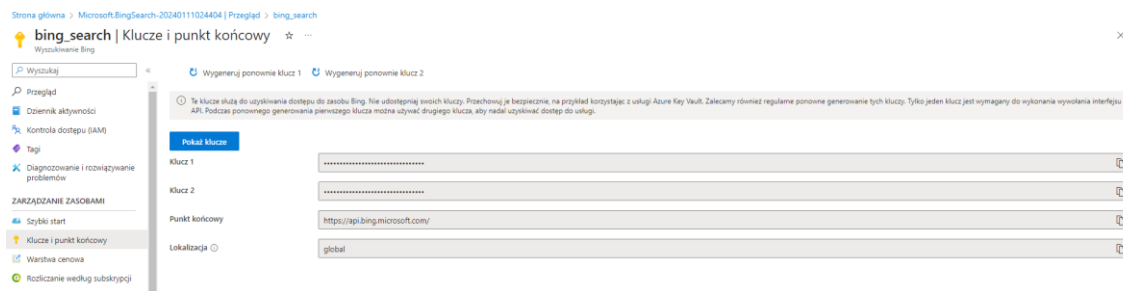
Wyszukaj

Wygeneruj ponownie klucz 1 Wygeneruj ponownie klucz 2

These keys are used to access your Azure AI service API. Do not share your keys. Store them securely— for example, using Azure Key Vault. We also recommend regenerating these keys regularly. Only one key is necessary to make an API call. When regenerating the first key, you can use the second key for continued access to the service.

**Pokaż klucze**

KLUCZ 1	.....
KLUCZ 2	.....
Lokalizacja/Region	northeurope
Punkt końcowy	https://cognitiveservices967.cognitiveservices.azure.com/



## Kroki 4-7: Tworzenie aplikacji

Poniżej podano kod aplikacji, która implementuje funkcję wyszukiwania poprzez Bing API. Skonfigurowano ją tak, aby używała klucza dostępu i punktu końcowego Bing Search API. Umożliwia ona wpisanie frazy do wyszukania oraz zwraca 3 najlepsze wyniki, następnie można ponownie wyszukać następną frazę lub wyjść z aplikacji.

```
using Newtonsoft.Json;
using System;
using System.Collections.Generic;
using System.IO;
using System.Net;
using System.Text;

namespace BingWebSearch
{
    class Program
    {
        static string subscriptionKey = "8677703b991e41869b92ed868c284abf";
        static string endpoint = "https://api.bing.microsoft.com/" +
"/v7.0/search";

        static void Main()
        {
            while (true)
            {
                Console.WriteLine("Wprowadź zapytanie do wyszukiwania (lub wpisz 'exit' aby zakończyć:");
                string query = Console.ReadLine();

                if (query.ToLower() == "exit")
                    break;

                PerformSearch(query);
            }
        }

        static void PerformSearch(string searchQuery)
        {
```

```

        Dictionary<String, String> relevantHeaders = new
Dictionary<String, String>();

        Console.OutputEncoding = Encoding.UTF8;
        Console.WriteLine($"Searching the Web for: {searchQuery}");
        var uriQuery = endpoint + "?q=" +
Uri.EscapeDataString(searchQuery);

        WebRequest request = HttpWebRequest.Create(uriQuery);
        request.Headers["Ocp-Apim-Subscription-Key"] = subscriptionKey;
        HttpWebResponse response =
(HttpWebResponse)request.GetResponseAsync().Result;
        string json = new
StreamReader(response.GetResponseStream()).ReadToEnd();

        foreach (String header in response.Headers)
        {
            if (header.StartsWith("BingAPIs-") || header.StartsWith("X-
MSEdge-"))
                relevantHeaders[header] = response.Headers[header];
        }

        Console.WriteLine("Relevant HTTP Headers:");
        foreach (var header in relevantHeaders)
            Console.WriteLine(header.Key + ": " + header.Value);

        Console.WriteLine("Top 3 Results:");
        dynamic parsedJson = JsonConvert.DeserializeObject(json);

        int resultCount = Math.Min(3, parsedJson.webPages.value.Count);
        for (int i = 0; i < resultCount; i++)
        {
            var result = parsedJson.webPages.value[i];
            Console.WriteLine($"{i + 1}. {result.name} - {result.url}");
        }
    }
}

```

Wyniki:

```
Wprowadź zapytanie do wyszukiwania (lub wpisz 'exit' aby zakończyć):
marchewka
Searching the Web for: marchewka
Relevant HTTP Headers:
BingAPIs-TraceId: 659f612fbb3a42ad9db39ebae9d45318
BingAPIs-SessionId: 022CD212EA344AE886F6A3209405AE68
X-MSEdge-ClientID: 2A960409E6CB696634BA100AE7C368DE
BingAPIs-Market: pl-PL
BingAPIs-RGUID: f026236db2d04742b9aa7c5cef5c1788
X-MSEdge-Ref: Ref A: 7C8E635635634CE5B881F2363FAEDF49 Ref B: AMS04EDGE2917 Ref C: 2024-01-11T03:31:59Z
Top 3 Results:
1. Marchew - smaczna i pełna witamin. Co warto o niej wiedzieć? - Medonet.pl - https://www.medonet.pl/zdrowie,marchew
sklad,-wlasosci,-przygotowanie,-na-co-uwazac,artykul,1730551.html
2. Marchewka - właściwości zdrowotne i zastosowanie w kosmetyce - https://zywienie.abczdrowie.pl/marchew-i-jej-leczni
-wlasosci
3. Marchew zwyczajna - Wikipedia, wolna encyklopedia - https://pl.wikipedia.org/wiki/Marchew_zwyczajna
Wprowadź zapytanie do wyszukiwania (lub wpisz 'exit' aby zakończyć):
```

```
Wprowadź zapytanie do wyszukiwania (lub wpisz 'exit' aby zakończyć):
polska
Searching the Web for: polska
Relevant HTTP Headers:
BingAPIs-TraceId: 659f6249062a4b7d8a30eba8a3a3fdc4
BingAPIs-SessionId: 642E1C6E8CAE40AE9915ADA4FDA9C480
X-MSEdge-ClientID: 24BEA4F6964165812D45B0F597AC64C2
BingAPIs-Market: pl-PL
BingAPIs-RGUID: 8f4700620650490ab060b73c225f153c
X-MSEdge-Ref: Ref A: F7FBD4F6649341C7AA7D069454A5EA02 Ref B: AMS04EDGE3322 Ref C: 2024-01-11T03:36:41Z
Top 3 Results:
1. Polska - Wikipedia, wolna encyklopedia - https://pl.wikipedia.org/wiki/Polska
2. Portal:Polska - Wikipedia, wolna encyklopedia - https://pl.wikipedia.org/wiki/Portal:Polska
3. Polska - najświeższe wiadomości - Polsat News - https://www.polsatnews.pl/wiadomosci/polska/
Wprowadź zapytanie do wyszukiwania (lub wpisz 'exit' aby zakończyć):
```

```
Wprowadź zapytanie do wyszukiwania (lub wpisz 'exit' aby zakończyć):
panele fotowoltaiczne
Searching the Web for: panele fotowoltaiczne
Relevant HTTP Headers:
BingAPIs-TraceId: 659f6271ccb7487b82848adc913d21b3
BingAPIs-SessionId: 1C446A4D20C748B99B707B20048A95B5
X-MSEdge-ClientID: 27560A2966D0699524631E2A67D868A8
BingAPIs-Market: pl-PL
BingAPIs-RGUID: ac9d156e3a124a92adf0d5ca9b2d430a
X-MSEdge-Ref: Ref A: 7ADB9F9B332DE4244AF6B968E228330F0 Ref B: AMS04EDGE3107 Ref C: 2024-01-11T03:37:21Z
Top 3 Results:
1. Panele fotowoltaiczne - Sklep Soltech - https://sklepsoltech.pl/pl/c/Panele-fotowoltaiczne/13
2. Panele fotowoltaiczne - Jakie wybrać? kompletny przewodnik! - https://fotowoltaikaonline.pl/panele-fotowoltaiczne
3. Panele fotowoltaiczne - Niska cena na Allegro.pl - https://allegro.pl/kategoria/ogrzewanie-kolektory-sloneczne-i-pa
le-fotowoltaiczne-258713?rodzaj=panel
Wprowadź zapytanie do wyszukiwania (lub wpisz 'exit' aby zakończyć):
```

## Ćwiczenie: Praca z Azure Speech Service

Celem tego ćwiczenia jest zaznajomienie się z Azure Speech Service oraz praktyczne wykorzystanie tej usługi do rozpoznawania mowy, konwersji tekstu na mowę i tłumaczenia mowy. Będziesz tworzyć aplikację, która wykorzystuje Speech Service API do przeprowadzania tych operacji.

### Krok 1 i 2: Utworzenie zasobu Azure Speech Service

Poniżej podano wszystkie wybrane opcje.

## Tworzenie usług mowy

Basics Sieć Identity Tagi Przejrzyj i utwórz

🔗 Pobierz szablon do automatyzacji

### POSTANOWIENIA

Klikając przycisk „Utwórz” (a) akceptuję postanowienia prawne i obowiązkujące o ochronie prywatności składowane z wymienionym powyżej ofertą usługi w Marketplace; (b) upoważniam firmę Microsoft do naliczania opłat za korzystanie z ofert (zgodnie z obecnie wybraną formą płatności) z taką samą częstotliwością rozliczeń jak w przypadku subskrypcji platformy Azure; (c) zgadzam się na udostępnianie przez firmę Microsoft moich informacji kontaktowych oraz informacji dotyczących użycia transakcji dostawcom tych ofert na potrzeby pomocy technicznej, roszczeń i innych działań związanych z transakcjami. Firma Microsoft nie przyznaje praw do ofert innych firm. Więcej szczegółów można znaleźć na stronie [Warunki korzystania z usługi Azure Marketplace](#).

### Basics

Subskrypcja	Azure for Students
Grupa zasobów	uslugi_chmurowe
Region	North Europe
Nazwa	uslugimowy454
Warstwa cenowa	Free F0

### Sieć

Typ	Dostęp do tego zasobu można uzyskiwać ze wszystkich sieci, w tym internetu.
-----	-----------------------------------------------------------------------------

### Identity

Typ tożsamości	None
----------------	------

◀ Wstecz **Utwórz**

## Krok 3: Konfiguracja Speech Service w Azure

uslugimowy454 | Klucze i punkt końcowy ☆

Wyszukaj Wygeneruj ponownie klucz 1 Wygeneruj ponownie klucz 2

Przebieg  
Dziennik aktywności  
Kontrola dostępu (IAM)  
Tagi  
Diagnostowanie i rozwiązywanie problemów

Zarządzanie zasobami  
**Klucze i punkt końcowy**  
Szyfrowanie  
Warstwa cenowa  
Sieć  
Tożsamość  
Analiza kosztów  
Właściwości  
Blokady

These keys are used to access your Azure AI service API. Do not share your keys. Store them securely— for example, using Azure Key Vault. We also recommend regenerating these keys regularly. Only one key is necessary to make an API call. When regenerating the first key, you can use the second key for continued access to the service.

**Pokaż klucze**

KLUCZ 1	.....
KLUCZ 2	.....
Lokalizacja/Region	northeurope
Punkt końcowy	<a href="https://northeurope.api.cognitive.microsoft.com/">https://northeurope.api.cognitive.microsoft.com/</a>

## Kroki 4-5: Tworzenie aplikacji i implementacja funkcji rozpoznawania mowy

Na początku dołączono odpowiednią bibliotekę za pomocą komendy: `dotnet add package Microsoft.CognitiveServices.Speech`. Utworzono aplikację służącą do rozpoznawania mowy, skonfigurowano ją tak, aby używała klucza dostępu i punktu końcowego Speech Service API. Polega ona na ciągłym przechwytywaniu dźwięku z mikrofonu i następnie wypisaniu rozpoznanego tekstu w konsoli.

```
using System;
using System.IO;
using System.Threading.Tasks;
using Microsoft.CognitiveServices.Speech;
using Microsoft.CognitiveServices.Speech.Audio;

class Program
{
    static string speechKey = "cb5fc29167d7499d804cf3d57c9412ee";
    static string speechRegion = "northeurope";

    static void OutputSpeechRecognitionResult(SpeechRecognitionResult
speechRecognitionResult)
    {
        switch (speechRecognitionResult.Reason)
        {
```

```

        case ResultReason.RecognizedSpeech:
            Console.WriteLine($"RECOGNIZED:
Text={speechRecognitionResult.Text}");
            break;
        case ResultReason.NoMatch:
            Console.WriteLine($"NOMATCH: Speech could not be
recognized.");
            break;
        case ResultReason.Canceled:
            var cancellation =
CancellationDetails.FromResult(speechRecognitionResult);
            Console.WriteLine($"CANCELED: Reason={cancellation.Reason}");

            if (cancellation.Reason == CancellationReason.Error)
            {
                Console.WriteLine($"CANCELED:
ErrorCode={cancellation.ErrorCode}");
                Console.WriteLine($"CANCELED:
ErrorDetails={cancellation.ErrorDetails}");
                Console.WriteLine($"CANCELED: Did you set the speech
resource key and region values?");
            }
            break;
    }
}

async static Task Main(string[] args)
{
    var speechConfig = SpeechConfig.FromSubscription(speechKey,
speechRegion);
    speechConfig.SpeechRecognitionLanguage = "pl-PL";

    using var audioConfig = AudioConfig.FromDefaultMicrophoneInput();
    using var speechRecognizer = new SpeechRecognizer(speechConfig,
audioConfig);

    Console.WriteLine("Speak into your microphone.");
    var speechRecognitionResult = await
speechRecognizer.RecognizeOnceAsync();
    OutputSpeechRecognitionResult(speechRecognitionResult);
}
}

```

Wyniki:

```

Speak into your microphone.
RECOGNIZED: Text=Na straganie w dzień targowy takie słyszy się rozmowy. Może pan się o mnie oprze pan? Tak wiednie, pani
e koprze, coś się dziwić? Mój szczypiorku leżę tutaj już od wtorku.

```

Krok 6: Implementacja konwersji tekstu na mowę

Kolejną wersją aplikacji jest implementacja konwersji tekstu pisanego na mowę. Analogicznie użyto tego samego klucza oraz endpointu

```
using System;
using System.IO;
using System.Threading.Tasks;
using Microsoft.CognitiveServices.Speech;
using Microsoft.CognitiveServices.Speech.Audio;

class Program
{
    static string speechKey = "cb5fc29167d7499d804cf3d57c9412ee";
    static string speechRegion = "northeurope";

    static void OutputSpeechSynthesisResult(SpeechSynthesisResult
speechSynthesisResult, string text)
    {
        switch (speechSynthesisResult.Reason)
        {
            case ResultReason.SynthesizingAudioCompleted:
                Console.WriteLine($"Speech synthesized for text: [{text}]");
                break;
            case ResultReason.Canceled:
                var cancellation =
SpeechSynthesisCancellationDetails.FromResult(speechSynthesisResult);
                Console.WriteLine($"CANCELED: Reason={cancellation.Reason}");

                if (cancellation.Reason == CancellationReason.Error)
                {
                    Console.WriteLine($"CANCELED:
ErrorCode={cancellation.ErrorCode}");
                    Console.WriteLine($"CANCELED:
ErrorDetails=[{cancellation.ErrorDetails}]");
                    Console.WriteLine($"CANCELED: Did you set the speech
resource key and region values?");
                }
                break;
            default:
                break;
        }
    }

    async static Task Main(string[] args)
    {
        var speechConfig = SpeechConfig.FromSubscription(speechKey,
speechRegion);
        speechConfig.SpeechSynthesisVoiceName = "pl-PL-ZofiaNeural";

        using (var speechSynthesizer = new SpeechSynthesizer(speechConfig))
```

```

    {
        Console.WriteLine("Enter some text that you want to speak >");
        string text = Console.ReadLine();

        var speechSynthesisResult = await
speechSynthesizer.SpeakTextAsync(text);
        OutputSpeechSynthesisResult(speechSynthesisResult, text);
    }

    Console.WriteLine("Press any key to exit...");
    Console.ReadKey();
}
}

```

Wyniki:

```

Enter some text that you want to speak >
Na straganie w dzień targowy takie słyszy się rozmowy
Speech synthesized for text: [Na straganie w dzie targowy takie s yszy si rozmowy]
Press any key to exit...

```

Następnie wprowadzono możliwość dostosowania parametrów głosu (szybkość mówienia, głośność, wysokość dźwięku) poprzez ssml.

```

using System;
using System.IO;
using System.Threading.Tasks;
using Microsoft.CognitiveServices.Speech;
using Microsoft.CognitiveServices.Speech.Audio;

class Program
{
    static string speechKey = "cb5fc29167d7499d804cf3d57c9412ee";
    static string speechRegion = "northeurope";

    async static Task Main(string[] args)
    {
        var speechConfig = SpeechConfig.FromSubscription(speechKey,
speechRegion);

        var speechSynthesisVoiceName = "pl-PL-ZofiaNeural";
        var ssml = @$"<speak xmlns='http://www.w3.org/2001/10/synthesis'
xmlns:mstts='http://www.w3.org/2001/mstts'
xmlns:emo='http://www.w3.org/2009/10/emotionml' version='1.0' xml:lang='pl-
PL'>
        <voice name='{speechSynthesisVoiceName}'><prosody rate='+100.00%'
volume='-50.00%' pitch='+50.00%'>Na straganie w dzień targowy takie słyszy się
rozmowy.</prosody></voice></speak>";

        using (var speechSynthesizer = new SpeechSynthesizer(speechConfig))

```



```

{
    Console.WriteLine($"SSML to synthesize: \r\n{ssml}");
    var speechSynthesisResult = await
speechSynthesizer.SpeakSsmlAsync(ssml);

    switch (speechSynthesisResult.Reason)
    {
        case ResultReason.SynthesizingAudioCompleted:
            Console.WriteLine("SynthesizingAudioCompleted result");
            break;
        case ResultReason.Canceled:
            var cancellation =
SpeechSynthesisCancellationDetails.FromResult(speechSynthesisResult);
            Console.WriteLine($"CANCELED:
Reason={cancellation.Reason}");

            if (cancellation.Reason == CancellationReason.Error)
            {
                Console.WriteLine($"CANCELED:
ErrorCode={cancellation.ErrorCode}");
                Console.WriteLine($"CANCELED:
ErrorDetails=[{cancellation.ErrorDetails}]");
                Console.WriteLine($"CANCELED: Did you set the speech
resource key and region values?");
            }
            break;
        default:
            break;
    }
}

Console.WriteLine("Press any key to exit...");
Console.ReadKey();
}
}

```

Wyniki:

```

SSML to synthesize:
<speak xmlns='http://www.w3.org/2001/10/synthesis' xmlns:mstts='http://www.w3.org/2001/mstts' xmlns:emo='http://www.w3.org/2009/10/emotionml' version='1.0' xml:lang='pl-PL'>
  <voice name='pl-PL-ZofiaNeural'><prosody rate='+100.00%' volume='-50.00%' pitch='+50.00%'>Na straganie w dzień targowy takie słyszy się rozmowy.</prosody></voice></speak>
SynthesizingAudioCompleted result
Press any key to exit...

```

Słychać poprawne zastosowanie tej konfiguracji.

Krok 7: Dodatkowe funkcjonalności – tłumaczenie

Utworzono aplikację korzystającą z tego samego klucza oraz endpointu Speech Service API.

Dźwięk przechwytywany jest z mikrofonu oraz następnie wyświetlany jest przetłumaczony tekst.

```

using System;
using System.IO;
using System.Threading.Tasks;
using Microsoft.CognitiveServices.Speech;
using Microsoft.CognitiveServices.Speech.Audio;
using Microsoft.CognitiveServices.Speech.Translation;

class Program
{
    static string speechKey = "cb5fc29167d7499d804cf3d57c9412ee";
    static string speechRegion = "northeurope";

    static void OutputSpeechRecognitionResult(TranslationRecognitionResult
translationRecognitionResult)
    {
        switch (translationRecognitionResult.Reason)
        {
            case ResultReason.TranslatedSpeech:
                Console.WriteLine($"RECOGNIZED:
Text={translationRecognitionResult.Text}");
                foreach (var element in
translationRecognitionResult.Translations)
                {
                    Console.WriteLine($"TRANSLATED into '{element.Key}':
{element.Value}");
                }
                break;
            case ResultReason.NoMatch:
                Console.WriteLine($"NOMATCH: Speech could not be
recognized.");
                break;
            case ResultReason.Canceled:
                var cancellation =
CancellationDetails.FromResult(translationRecognitionResult);
                Console.WriteLine($"CANCELED: Reason={cancellation.Reason}");

                if (cancellation.Reason == CancellationReason.Error)
                {
                    Console.WriteLine($"CANCELED:
ErrorCode={cancellation.ErrorCode}");
                    Console.WriteLine($"CANCELED:
ErrorDetails={cancellation.ErrorDetails}");
                    Console.WriteLine($"CANCELED: Did you set the speech
resource key and region values?");
                }
                break;
        }
    }
}

```

```

async static Task Main(string[] args)
{
    var speechTranslationConfig =
SpeechTranslationConfig.FromSubscription(speechKey, speechRegion);
    speechTranslationConfig.SpeechRecognitionLanguage = "pl-PL";
    speechTranslationConfig.AddTargetLanguage("es");

    using var audioConfig = AudioConfig.FromDefaultMicrophoneInput();
    using var translationRecognizer = new
TranslationRecognizer(speechTranslationConfig, audioConfig);

    Console.WriteLine("Speak into your microphone.");
    var translationRecognitionResult = await
translationRecognizer.RecognizeOnceAsync();
    OutputSpeechRecognitionResult(translationRecognitionResult);
}
}

```

Wyniki:

```

Speak into your microphone.
RECOGNIZED: Text=Na straganie w dzień targowy takie słyszy się rozmowy.
TRANSLATED into 'es': En un puesto de mercado en un día de mercado, se pueden escuchar este tipo de conversaciones.

```

## Ćwiczenie: Wprowadzenie do Azure Form Recognizer

Celem tego ćwiczenia jest zapoznanie się z Azure Form Recognizer i jego zastosowaniem do analizowania i przetwarzania dokumentów. Będziesz tworzyć aplikację, która wykorzystuje Form Recognizer API do wydobywania danych z obrazów dokumentów i formularzy.

### Krok 1 i 2: Utworzenie zasobu Azure Form Recognizer

Poniżej podano wszystkie wybrane opcje.

Create Document Intelligence ...

Basics Network Identity Tags **Review + create**

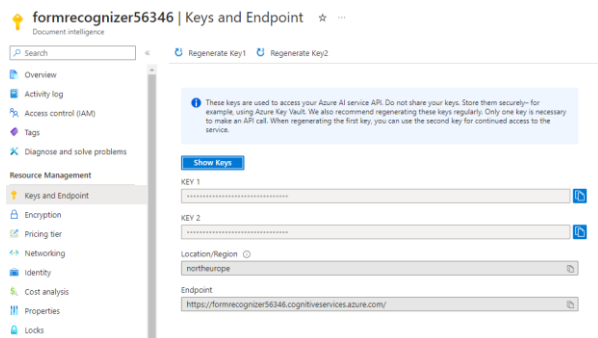
[View automation template](#)

TERMS

By clicking "Create", I (a) agree to the legal terms and privacy statement(s) associated with the Marketplace offering(s) listed above, (b) authorize Microsoft to bill my current payment method for the fees associated with the offering(s), with the same billing frequency as my Azure subscription; and (c) agree that Microsoft may share my contact, usage and transactional information with the provider(s) of the offering(s) for support, billing and other transactional activities. Microsoft does not provide rights for third-party offerings. See the [Azure Marketplace Terms](#) for additional details.

<b>Basics</b>	
Subscription	Azure for Students
Resource group	uslugi_chmurowe
Region	North Europe
Name	formrecognizer56346
Pricing tier	Free F0 (200 Pages per month, 20 Calls per minute for recognizer API, 1 Call ...
<b>Network</b>	
Type	All networks, including the internet, can access this resource.
<b>Identity</b>	
Identity type	None

### Krok 3: Konfiguracja Form Recognizer w Azure



## Krok 4 i 5: Tworzenie aplikacji

Utworzono aplikację do analizowania i przetwarzania dokumentów. Skonfigurowano ją, aby korzystała z odpowiedniego klucza dostępu oraz punktu końcowego. Następnie przystosowano ją do analizy przykładowego dokumentu dostępnego w Azure.

```
using Azure;
using Azure.AI.FormRecognizer.DocumentAnalysis;

string endpoint = "0502371944e848acac650a51f7c059b0";
string key = "https://formrecognizer56346.cognitiveservices.azure.com/";
AzureKeyCredential credential = new AzureKeyCredential(key);
DocumentAnalysisClient client = new DocumentAnalysisClient(new Uri(endpoint),
credential);

Uri fileUri = new Uri ("https://raw.githubusercontent.com/Azure-
Samples/cognitive-services-REST-api-samples/master/curl/form-
recognizer/sample-layout.pdf");

AnalyzeDocumentOperation operation = await
client.AnalyzeDocumentFromUriAsync(WaitUntil.Completed, "prebuilt-layout",
fileUri);

AnalyzeResult result = operation.Value;

foreach (DocumentPage page in result.Pages)
{
    Console.WriteLine($"Document Page {page.PageNumber} has {page.Lines.Count}
line(s), {page.Words.Count} word(s),");
    Console.WriteLine($"and {page.SelectionMarks.Count} selection mark(s).");

    for (int i = 0; i < page.Lines.Count; i++)
    {
        DocumentLine line = page.Lines[i];
        Console.WriteLine($" Line {i} has content: '{line.Content}'.");

        Console.WriteLine($"    Its bounding box is:");
        Console.WriteLine($"        Upper left => X:
{line.BoundingPolygon[0].X}, Y= {line.BoundingPolygon[0].Y}");
    }
}
```

```

        Console.WriteLine($"        Upper right => X:
{line.BoundingPolygon[1].X}, Y= {line.BoundingPolygon[1].Y}");
        Console.WriteLine($"        Lower right => X:
{line.BoundingPolygon[2].X}, Y= {line.BoundingPolygon[2].Y}");
        Console.WriteLine($"        Lower left => X:
{line.BoundingPolygon[3].X}, Y= {line.BoundingPolygon[3].Y}");
    }

    for (int i = 0; i < page.SelectionMarks.Count; i++)
    {
        DocumentSelectionMark selectionMark = page.SelectionMarks[i];

        Console.WriteLine($" Selection Mark {i} is {selectionMark.State}.");
        Console.WriteLine($" Its bounding box is:");
        Console.WriteLine($"        Upper left => X:
{selectionMark.BoundingPolygon[0].X}, Y=
{selectionMark.BoundingPolygon[0].Y}");
        Console.WriteLine($"        Upper right => X:
{selectionMark.BoundingPolygon[1].X}, Y=
{selectionMark.BoundingPolygon[1].Y}");
        Console.WriteLine($"        Lower right => X:
{selectionMark.BoundingPolygon[2].X}, Y=
{selectionMark.BoundingPolygon[2].Y}");
        Console.WriteLine($"        Lower left => X:
{selectionMark.BoundingPolygon[3].X}, Y=
{selectionMark.BoundingPolygon[3].Y}");
    }
}

foreach (DocumentStyle style in result.Styles)
{
    bool isHandwritten = style.IsHandwritten.HasValue && style.IsHandwritten
== true;

    if (isHandwritten && style.Confidence > 0.8)
    {
        Console.WriteLine($"Handwritten content found:");

        foreach (DocumentSpan span in style.Spans)
        {
            Console.WriteLine($" Content:
{result.Content.Substring(span.Index, span.Length)}");
        }
    }
}

Console.WriteLine("The following tables were extracted:");

for (int i = 0; i < result.Tables.Count; i++)

```

```
{
    DocumentTable table = result.Tables[i];
    Console.WriteLine($" Table {i} has {table.RowCount} rows and
{table.ColumnCount} columns.");

    foreach (DocumentTableCell cell in table.Cells)
    {
        Console.WriteLine($" Cell ({cell.RowIndex}, {cell.ColumnIndex}) has
kind '{cell.Kind}' and content: '{cell.Content}'");
    }
}
```

Wyniki:

```
Unhandled exception. System.UriFormatException: Invalid URI: The format of the URI could not be determined.
at System.Uri.CreateThis(String uri, Boolean dontEscape, UriKind uriKind, UriCreationOptions& creationOptions)
at System.Uri..ctor(String uriString)
at Program.<Main>$(String[] args) in C:\Users\plok\Lab1\Program.cs:line 705
at Program.<Main>(String[] args)
```

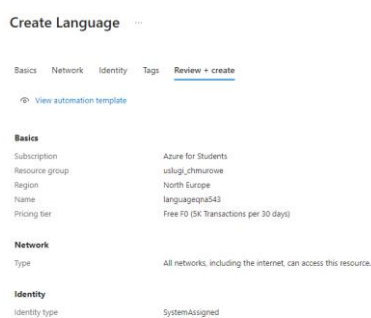
Niewyjaśnionym problemem jest jednak błąd URI, przez co nie otrzymano wyniku tej analizy.

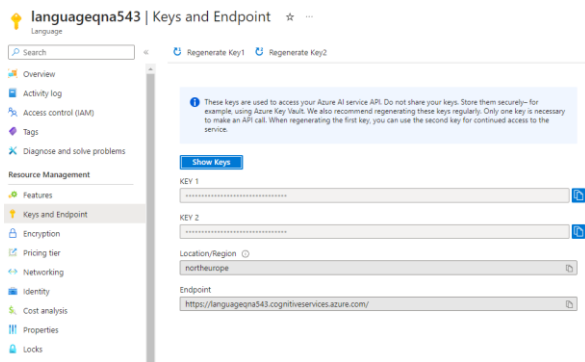
## Ćwiczenie: Tworzenie i Implementacja Chatbota z Azure QnA Maker

Celem tego ćwiczenia jest nauka korzystania z Azure QnA Maker do tworzenia, szkolenia i publikowania inteligentnej bazy wiedzy, która może być wykorzystana do obsługi chatbota. Ćwiczenie to umożliwi praktyczne zrozumienie, jak tworzyć i integrować boty oparte na AI, które mogą odpowiadać na pytania w sposób zautomatyzowany.

### Krok 1 i 2: Utworzenie zasobu Azure QnA Maker

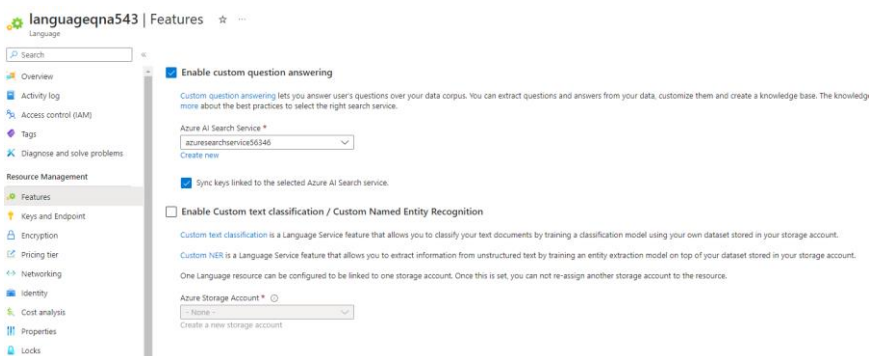
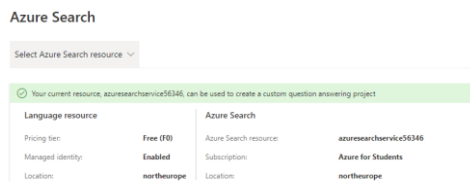
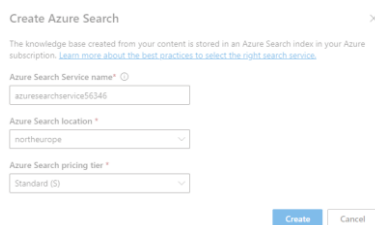
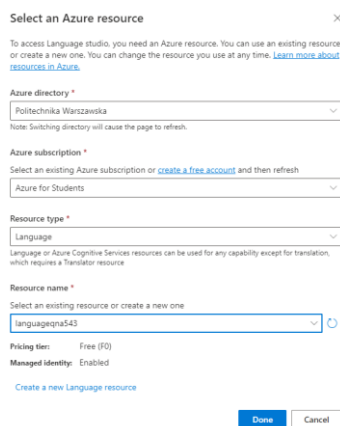
Poniżej przedstawiono wybrane opcje.





### Krok 3: Tworzenie bazy wiedzy w QnA Maker

Poniżej pokazano proces tworzenia bazy w QnA Maker Portal korzystającego z zasobu Azure Language i Azure Search Service.



Utworzono projekt oraz dodano przykładowy dokument z którego zaczytano pytanie oraz odpowiedzi na nie.

Create a project

Choose language setting

Enter basic information

Review and finish

Review and finish

Review the configurations you set for your project in the previous steps.

Projects in multiple languages allowed?  
No

Language resource  
languageqna543

Azure Search resource  
azuresearchservice56346

Project name  
project1

Description  
Opis

Source language  
English

Default answer when no response is returned  
No answer found

Back

Next

Create project

Cancel

## Manage sources

+ Add source Edit name Refresh URL Delete

1 items in list Filter

Source	Source name	Unstructured	Source type
structured.docx	word document	No	file

Language Studio > Custom question answering > project1 - Edit knowledge base

Question answer pairs (11) Synonyms (0)

+ Add

- Remove

↕ Swap

↺ Previous

↻ Refresh

✖ Close

🔍 Search pairs

Sources: word document Clear all

Why did Microsoft develop the Bot Framework?

While the Conversation User Interface (CUI) is upon us, at this point few...

What is the v4 SDK?

Bot Builder v4 SDK builds on the feedback and learnings from the prior...

When will you add more conversation experiences to the Bot Framework?

We plan on making continuous improvements to the Bot Framework...

I have a communication channel I'd like to be configurable with Bot Framework. Can I work with Microsoft to do that?

We have not provided a general mechanism for developers to add new...

If I want to create a bot for Skype, what tools and services should I use?

The Bot Framework is designed to build...

Edit knowledge base

0 unstructured sources and 1 structured sources. View sources

Why did Microsoft develop the Bot Framework?

Enable rich text Show context tree

Source: structured.docx

Answer

Alternate questions (1)

Follow up prompts (0)

Metadata (0)

## Krok 4: Szkolenie i publikowanie bazy wiedzy

Test

Response options

☐ Include short answer response

☐ Use deployed knowledge base

Show advanced options

Clear

Can I create my bot?

Inspect

The Bot Framework is designed to build, connect, and deploy high quality, responsive, performant and scalable bots for Skype and many other channels. The SDK can be used to create text/ims, image, button and card-capable bots (which constitute the majority of bot interactions today across conversation experiences) as well as bot interactions which are Skype-specific such as rich audio and video experiences.

If you already have a great bot and would like to reach the Skype audience, your bot can easily be connected to Skype (or any supported channel) via the Bot Builder for REST API (provided it has an internet-accessible REST endpoint).

Type your message and press enter

Question

Can I create my bot?

Top answers

Select the most appropriate answer:

Save this query as alternate question for this answer

Answer

Edit question answer pair

The Bot Framework is designed to build, connect, and deploy high quality, responsive, performant and scalable bots for Skype and many other channels. The SDK can be used to create text/ims, image, button and card-capable bots (which constitute the majority of bot interactions today across conversation experiences) as well as bot interactions which are Skype-specific such as rich audio and video experiences.

If you already have a great bot and would like to reach the Skype audience, your bot can easily be connected to Skype (or any supported channel) via the Bot Builder for REST API (provided it has an internet-accessible REST endpoint).

Confidence score: 0.55

See additional information

Answer

Edit question answer pair

While the Conversation User Interface (CUI) is upon us, at this point few developers have the expertise and tools needed to create new conversational experiences or enable existing applications and services with a conversational interface their users can enjoy. We have created the Bot Framework to make it easier for developers to build and connect great bots to users, wherever they converse, including on Microsoft's premier channels.

Confidence score: n/a

## Krok 5: Tworzenie i integracja chatbota



Poniżej podano proces tworzenia chatbota - wszystkie wykorzystywane zasoby oraz konfiguracje (problematyczne było jednak skorzystanie z App Service Plan, którego nie mogłem w tym miejscu w ogóle utworzyć).

#### Deploy knowledge base

Deploy knowledge base and create a bot in a few clicks.

[Deploy](#) [Get prediction URL](#)

✔ Your knowledge base is now deployed. You can get your prediction URL or create a bot.			
<strong>Knowledge base status</strong>			
State:	Deployed		Resource: languagegn543
Deployment Date:	1/11/2024		Location: northeurope
Deployment Time:	8:07:34 AM		Tier: Free (F0)

#### Next steps: Create a bot

Step 1: [Read the documentation](#) to learn more about creating bots.

Step 2: Go to Azure to create a bot.

[Create a bot](#)

#### Create App Service Plan

Basics Tags **Review + create**

##### Summary

**App Service Plan**  
by Microsoft

##### Details

Subscription	Azure for Students
Resource Group	uslugi_chmurowe
Name	AppServicePlan54
Operating System	Windows
Region	North Europe
SKU	Free
ACU	Shared infrastructure
Memory	1 GB memory

##### Basics

Subscription	Azure for Students
Resource group	uslugi_chmurowe
Resource group location	East US
Bot handle	languagegn543-bot
Pricing tier	Standard
Creation type	Create new User-assigned managed identity

##### Web App

App name	languagegn543-bot-eba7
SDK language selection	C#
Creation type	Use existing app service plan
App service plan	-
Language resource key	project1
Language project name	https://languagegn543.cognitiveservices.azure.com
Language service endpoint hostname	/subscriptions/8d5ceb22-44c7-4367-bf6b-0778096e055
Subscription ID	uslugi_chmurowe
Resource Group Name	languagegn543
Account Name	

## Ćwiczenie: Tworzenie i Wdrażanie Modelu ML do Rozpoznawania Obrazów z Azure Machine Learning i Integracja z Aplikacją C# MAUI

Celem tego ćwiczenia jest zapoznanie się z możliwością tworzenia, szkolenia, wdrażania modeli ML do rozpoznawania obrazów z Azure Machine Learning Studio i integracji z aplikacją mobilną C# MAUI.

### Krok 1 i 2. Przygotowanie Danych:

Ze względu na problemy z dodaniem danych innym sposobem niż przez URI open-source, dodano jeden z datasetów dostępnych w platformie Azure, które występowały w formie tabelarycznej (wykonano trenowanie modelu na jednym z takich zbiorów danych – sample-diabetes). Wybrano ten zbiór danych do eksperymentu oraz ustawiono zmienną wyjściową (przewidywaną) Y.

**Utwórz nowe zadanie zautomatyzowanego uczenia maszynowego**

Wybierz zasób danych

**Konfiguruj zadanie**

Wybierz zadanie i ustawienia

Konfiguracja hiperparametru (tylko przetwarzanie obrazów)

Zweryfikuj i przetestuj

AGE (Integer)

SEX (Integer)

BMI (Decimal)

BP (Decimal)

S1 (Integer)

S2 (Decimal)

S3 (Decimal)

S4 (Decimal)

S5 (Decimal)

S6 (Integer)

Y (Integer)

Y (Integer)

Wybierz typ obliczeniowy

Klaster obliczeniowy

Wybierz typ klastra obliczeniowego usługi Azure ML \*

Nie znaleziono żadnego typu klastra obliczeniowego

+ Nowy

Odśwież środowiska obliczeniowe

Wstecz

Dalej

Kroki 2 i 3. Tworzenie i szkolenie modelu ML.

Jako zadanie modelu wybrano regresję, walidację poprzez krzyżowe sprawdzanie poprawności, a dane testowe wyszczególniono przed uczeniem ze zbioru testowego.

**Utwórz nowe zadanie zautomatyzowanego uczenia maszynowego**

Wybierz zasób danych

**Konfiguruj zadanie**

Wybierz zadanie i ustawienia

Konfiguracja hiperparametru (tylko przetwarzanie obrazów)

Zweryfikuj i przetestuj

**Konfiguruj zadanie**

Wybierz jeden z istniejących eksperymentów lub utwórz nowy, a następnie wybierz kolumnę docelową i treningowe środowisko obliczeniowe.

[Dowiedz się więcej o konfigurowaniu eksperymentu.](#)

Zasób danych

danedouczenia754 (Wyświetl zasób danych)

Nazwa eksperymentu

Utwórz nowy

Nazwa nowego eksperymentu \*

experiment7568

Kolumna docelowa \*

Y (Integer)

Wybierz typ obliczeniowy

Klaster obliczeniowy

Wybierz typ klastra obliczeniowego usługi Azure ML \*

Nie znaleziono żadnego typu klastra obliczeniowego

+ Nowy

Odśwież środowiska obliczeniowe

Wstecz

Dalej

Utwórz nowe zadanie zautomatyzowanego uczenia maszynowego

- ✓ Wybierz zasób danych
- ✓ Konfiguruj zadanie
- 1 Wybierz zadanie i ustawienia
- 2 Konfiguracja hiperparametru (tylko przetwarzanie obrazów)
- 3 Zweryfikuj i przetestuj

## Wybierz zadanie i ustawienia

Wybierz typ zadania uczenia maszynowego dla eksperymentu. Aby dokładniej dostosować eksperyment, wybierz dodatkowe ustawienia konfiguracji lub dołóż cech.



 **Klasyfikacja**

W celu przewidywania jednej z kilku kategorii w kolumnie docelowej. Tak/nie, niebieski, czerwony, zielony



 Regresja

Przewidywanie ciągłych wartości liczbowych



## Prognozowanie szeregów czasowych

Przewidywanie wartości na podstawie czasu



### 6.2 Przetwarzanie języka naturalnego

Przewidywanie tylko na podstawie typów danych tekstu przy użyciu klasyfikacji wieloklasowej lub wieloetykietowej



## Przetwarzanie obrazów

Klasyfikacja obrazów wieloklasowych lub wieloetykietowych, wykrywanie obiektów i segmentacja wystąpień

[Wyświetl dodatkowe ustawienia konfiguracji](#) [Wyświetl ustawienia doboru cech](#)

[Utwórz nowe zadanie zautomatyzowanego uczenia maszynowego](#)

- ✓ Wybierz zasób danych
- ✓ Konfiguruj zadanie
- ✓ Wybierz zadanie i ustawienia
- ✓ Konfiguracja hiperparametru (tylko przetwarzanie obrazów)
- ❌ Zweryfikuj i przetestuj

### Wybór typu walidacji i testowania

Możesz wybrać typ walidacji i zasób danych testowych jako krok opcjonalny. Udostępnianie własnych zasobów sprawdzania poprawności i testowania danych to obecnie funkcje w wersji zapoznawczej.

Typ weryfikacji ⓘ

k-krotne krzyżowe sprawdzanie poprawności

Liczba krzyżowych sprawdzeń poprawności \*

5

Zasób danych testowych (wersja zapoznawcza) ⓘ

Podział testu (wybierz procent danych trenowania)

Procentowy test danych \*

10

Zautomatyzowane uczenie maszynowe zaleca, aby od 10 do 30 procent danych wstrzymać na potrzeby test.

[illegible]

Najlepsze podsumowanie modelu

Nazwa algorytmu

VotingEnsemble

Szczegóły zespołu

[Widz szczegóły zespołu](#)

Znormalizowany pierwiastek błędu średniokwadratowego

0.16950 [Widz wszystkie inne metryki](#)

Probkowanie

100.00 % [Widz](#)

Zarejestrowane modele

Nie ma jeszcze rejestracji

Stan wdrażania

Nie ma jeszcze wdrożenia

Dane wejściowe

Nazwa wejścia: training\_data

Zestaw danych: daneDouczenia754.1 [Widz](#)

Podsumowanie przebiegu

Typ zadania

Regresja [Widz ustawienia konfiguracji](#)

Dobór cech

Automatycznie

Metryka podstawowa

Znormalizowany pierwiastek błędu średniokwadratowego

Nazwa eksperymentu

experiment7568

Dane wyjściowe

Nazwa wyjścia: best\_model

Model: azureml\_AutoML\_9dc6b7ca-9f78-438d-a63c-9bccc3724ab\_40\_output\_mflow\_log\_model\_1888071586.1

Identyfikator URI zasobu: [azuremlazureml\\_AutoML\\_9dc6b7ca-9f78-438d-a63c-9bccc3724ab\\_40\\_output\\_mflow\\_log\\_model\\_1888071586.1](#)

Nazwa wyjścia: full\_training\_dataset

Zestaw danych: E508377-9ef9-4cad-ac94-d6e6adfb4ba [Widz](#)

Rejestrowanie modelu na podstawie danych wyjściowych zadania

Wybierz zadanie

Wybierz wyjście

Ustawienia modelu

Przegląd

Przegląd

Przejrzyj lub wprowadź zmiany w wybranych opcjach.

Wybierz zadanie

Zadanie

purple\_leather\_tdctr0w2

Wybierz wyjście

Typ modelu

MLflow

Nazwa dane wyjściowe modelu

best\_model (azureml\_AutoML\_9dc6b7ca-9f78-438d-a63c-9bccc3724ab\_40\_output\_mflow\_log\_model\_1888071586.1)

Ustawienia modelu

Nazwa

nazwaModelum756

Opis

--

Wersja

--

Tagi

[Brak tag](#)

Typ zadania

Regresja

Metryka podstawowa

Znormalizowany pierwiastek błędu średniokwadratowego

Wyjajni najlepszy model

Włączono

Dozwolone modele

--

Zablokowane modele

TensorFlowDNN, TensorFlowLinearRegressor

Liczba krzyżowych sprawdzeń poprawności

5

Głębokie uczenie

Wyłączono

Kryterium wyjścia

Czas treningu (godziny)

6

Próg wyniku metryki

--

Walidacja

Typ weryfikacji

k-krotne krzyżowe sprawdzanie poprawności

Współbieżność

Maksymalna współbieżna iteracja

1

Metryki przebiegu

Wyjajniowane odchylenie

0.49438

Średni bład bezwzględny

44.320

Średni bezwzględny bład procentowy

39.413

Mediana błędu bezwzględnego

40.524

Znormalizowany średni bład bezwzględny

0.13807

Znormalizowana mediana błędu bezwzględnego

0.12624

Znormalizowany pierwiastek błędu średniokwadratowego

0.16950

Znormalizowany pierwiastek logarytmu błędu średniokwadratowego

0.16041

Wynik R2

0.48658

Pierwiastek błędu średniokwadratowego

54.410

Pierwiastek logarytmu błędu średniokwadratowego

0.41567

Korelacja Spearmana

0.68571