Lab 7: Programowanie AI z Azure

Speech, Vision, Document Intelligence

Materiały na kolokwium

7 grudnia 2025

Spis treści

1 Azure AI Speech

1.1 Podstawy i Zasoby

Azure AI Speech - Konfiguracja

Tworzenie zasobu:

- Resource Group → Azure AI Speech
- **Region** - wybór wpływa na latency i dostępne języki
- **Endpoint** - URL do API
- **Keys** - primary/secondary (authentication)
- Best practice: przechowywanie w Key Vault

Pricing tiers:

- **Free (F0):** 5 audio hours/month STT, 0. 5M characters/month TTS
- **Standard (S0):** pay-as-you-go, unlimited, $1/hour STT, $4/1M chars TTS

1.2 Speech-to-Text (STT)

STT - Transkrypcja Mowy → Tekst

Realtime STT:

- Live transcription (mikrofon, streaming audio)
- Low latency (milisekundy)
- Partial results (intermediate recognition podczas mówienia)
- Use case: live captions, voice commands, dyktowanie

Batch STT:

- Asynchroniczne przetwarzanie plików audio
- Duże pliki (do kilku godzin)
- Higher accuracy (więcej czasu na processing)
- Use case: call center analytics, meeting transcripts, podcasts

Formaty Audio i Jakość

Formaty wspierane:

- **WAV** - PCM 16 kHz mono (RECOMMENDED - best quality)
- MP3, OGG, FLAC, OPUS

Sample rate:

- 8 kHz - telefonia (niższa jakość)
- 16 kHz - standard (recommended)
- 24 kHz, 48 kHz - higher quality

Mikrofony:

- Mono preferred (stereo może być używany, ale mono lepszy)
- Odległość 15-30 cm od ust
- Redukcja szumów tła (noise cancellation)

Modele jakości:

- **Neural** - najlepsza jakość (deep learning models)
- **Enhanced** - poprawiona akustyka dla specific scenarios
- Multilingual support - 100+ języków
- Automatic language detection

Speech Studio - Testowanie

Workflow:

1. Speech Studio → Transcribe
2. Nagraj z mikrofonu LUB upload WAV file (16 kHz mono)
3. Wybierz język rozpoznawania
4. Analyze - otrzymasz transkrypcję + confidence scores

SDK Integration (C#):

- NuGet: `Microsoft. CognitiveServices. Speech`
- SpeechConfig (endpoint, key, region)
- SpeechRecognizer (from mic / from file)
- RecognizeOnceAsync() - single utterance
- StartContinuousRecognitionAsync() - streaming

1.3 Text-to-Speech (TTS)

TTS - Synteza Mowy

Neural TTS:

- Naturalne brzmienie (deep neural networks)
- Human-like prosody (intonacja, rytm, akcent)
- Wiele głosów: męskie, żeńskie, dzieci
- Multilingual - 100+ języków
- Emocje: neutral, cheerful, sad, angry, excited, friendly

Use case:

- Virtual assistants (Alexa-like)
- Accessibility - czytniki ekranu dla niewidomych
- IVR systems (Interactive Voice Response - call centers)
- E-learning platforms
- Audiobooks
- Navigation systems

SSML - Speech Synthesis Markup Language

XML-based język do kontroli TTS
Kluczowe tagi:

- `<speak>` - root element
- `<voice name="en-US-JennyNeural»` - wybór głosu
- `<prosody rate="fast»` - prędkość (slow/medium/fast lub %)
- `<prosody pitch="high»` - wysokość głosu
- `<break time="500ms"/>` - pauza
- `<emphasis level="strong»` - nacisk na słowo
- `<phoneme ph="... »` - custom wymowa
- `<mstts:express-as style="cheerful»` - emocje

Output formats:

- RAW PCM, WAV
- MP3 (różne bitrate: 48k, 128k, 192k)
- OGG Opus

1.4 Speech Translation

Tłumaczenie Mowy Real-time

Funkcjonalność:

- Mowa (język A) → tekst (język B)
- Mowa (język A) → mowa (język B)
- Multi-language support (tłumaczenie na kilka języków jednocześnie)

Use case:

- International meetings/conferences
- Customer support (wielojęzyczny)
- Travel apps
- Real-time interpretation

Latency:

- Near real-time (1-2 sekundy opóźnienia)
- Partial translation results during speaking

1.5 Customizacja

Custom Speech

Dostosowanie modelu STT do specific scenarios
Custom Acoustic Model:

- Adaptacja do środowiska (hałas, echo, akcent)
- Training data: audio recordings + transkrypcje
- Minimum kilka godzin nagrań
- Use case: call centers z hałasem tła, nietypowe akcenty

Custom Language Model:

- Specjalistyczne słownictwo (medyczne, techniczne, prawnicze)
- Training data: teksty z domenową terminologią
- Improve recognition of domain-specific terms

Proces:

1. Upload training data (audio + transcripts / text)
2. Train custom model (kilka godzin)
3. Evaluate accuracy
4. Deploy custom endpoint

Custom Neural Voice

Własny głos syntezowany (TTS)
Wymagania:

- **Limited Access** - wymaga zatwierdzenia przez Microsoft
- Ethical use case approval (zapobieganie deepfake abuse)
- Training data: 300-2000 utterances (10-30 min nagrań)
- Professional recording environment

Proces aplikacji:

1. Submit use case description
2. Microsoft review (compliance, ethical use)
3. Approval (może zająć tygodnie)
4. Training z professional recordings

Use case:

- Brand voice (company assistant)
- Celebrity voice (za zgodą)
- Personalization (custom voice dla accessibility)

2 Azure AI Document Intelligence

2.1 Podstawy

Document Intelligence (Form Recognizer)

AI-powered extraction z dokumentów
Funkcje:

- **OCR** - Optical Character Recognition (tekst z obrazów)
- **Layout analysis** - struktura dokumentu (paragraphs, tables)
- **Key-value pairs** - pola formularzy
- **Table extraction** - dane tabelaryczne
- **Prebuilt models** - gotowe modele dla common documents
- **Custom models** - trenowanie własnych

Wspierane formaty:

- PDF, JPG, PNG, TIFF, BMP
- Handwriting i print text

2.2 Prebuilt Models

Gotowe Modele w Studio

Invoices (Faktury):

- Vendor name, date, invoice number
- Total amount, tax, line items
- Currency detection
- Confidence scores dla każdego pola

Receipts (Paragony):

- Merchant name, transaction date, total
- Line items (products, prices)
- Payment method, tip amount

ID/Passport:

- Full name, date of birth, document number
- Expiration date, nationality, address
- Face detection (photo location)

Business Cards:

- Name, job title, company
- Phone, email, address, website

W-2 (US tax form):

- Employee/employer info
- Wages, tax withheld, social security

Read (OCR):

- Plain text extraction
- Line-by-line output
- Handwriting recognition
- Language detection

Layout:

- Document structure (headers, paragraphs, tables)
- Bounding boxes dla text regions
- Reading order detection
- Selection marks (checkboxes)

Document Intelligence Studio

Testowanie prebuilt models:

1. Document Intelligence Studio → Prebuilt models
2. Wybierz model (np. Invoices)
3. Upload 2-3 sample documents (PDF/JPG)
4. Analyze - otrzymasz JSON output
5. Przejrzyj extracted fields:
    - Pola (vendor, date, total)
    - Tabele (line items)
    - Confidence scores (0. 0-1.0)
    - Bounding boxes (locations)

JSON output structure:

- `analyzeResult. documents[]` - extracted data
- `fields` - key-value pairs z confidence
- `tables` - row/column structure
- `pages` - page-level info

2.3 Custom Extraction

Custom Model - Trenowanie

Kiedy używać:

- Własny format dokumentu (nie covered by prebuilt)
- Specific fields potrzebne
- Multiple variants tego samego formularza

Proces:

1. **Przygotuj dane** - 5-15 dokumentów jednego typu (ideally 15-50)
2. **Upload do Blob Storage** - Store w Azure Storage
3. **Label fields w Studio**:
    - Draw bounding boxes wokół pól
    - Assign labels (np. "InvoiceDate", "Total")
    - Label tables (rows/columns)
4. **Train model** - kilka minut do godziny
5. **Evaluate metrics** - Precision, Recall, F1 score
6. **Test on new documents**
7. **Deploy** - use przez REST API / SDK

Compose - Łączenie Modeli

Problem: Różne warianty tego samego typu formularza
Rozwiązanie: Compose
Jak działa:

- Trenuj osobny model dla każdego variantu
- Combine models w jeden composed model
- Single endpoint - Azure automatycznie wybiera właściwy model

Use case:

- Faktury od różnych vendorów (różne layouts)
- Formularze w różnych językach
- Historical vs nowe wersje formularza

Max: 100 models w jednym composed model

Metryki Oceny

Precision (Dokładność):

- % poprawnie wyekstraktowanych wartości spośród wszystkich extracted
- High precision = mało false positives

Recall (Pokrycie):

- % znalezionych pól spośród wszystkich możliwych
- High recall = mało false negatives

F1 Score:

- Harmonic mean Precision i Recall
- Balansuje obie metryki
- F1 = 2 * (Precision * Recall) / (Precision + Recall)

Target:

- Precision > 0.90 (90%)
- Recall > 0. 90
- F1 > 0.90

3 Azure AI Vision

3.1 Image Analysis

Computer Vision - Analiza Obrazów

Funkcje:

- **Tags** - słowa kluczowe opisujące obraz
- **Descriptions** - pełne zdania opisujące scenę
- **Dense Captions** - szczegółowe opisy różnych regionów obrazu
- **OCR (Read API)** - ekstrakcja tekstu z obrazów
- **Object Detection** - lokalizacja obiektów (bounding boxes)
- **Face Detection** - wykrywanie twarzy, wiek, emocje
- **Adult Content Detection** - klasyfikacja treści (adult/racy/gory)
- **Color Analysis** - dominujące kolory, accent color
- **Image Type** - clipart vs photo, line drawing

Wspierane formaty:

- JPG, PNG, GIF, BMP
- Max size: 4 MB (lub 50 MB dla Read API)

Tags vs Descriptions vs Dense Captions

Tags (pojedyncze słowa):

- Lista keywords: "dog", "outdoor", "grass", "playing"
- Confidence score dla każdego tagu
- Use case: search indexing, kategoryzacja

Descriptions (całe zdania):

- Pełne zdanie: "A dog playing with a ball in a grassy field"
- 1-3 candidate descriptions z confidence
- Use case: alt text, accessibility

Dense Captions (regionalne):

- Szczegółowe opisy różnych części obrazu
- Bounding box + opis dla każdego regionu
- Przykład: "a red car"(bbox: left door), "blue sky"(bbox: upper part)
- Use case: detailed image understanding, scene composition

Multilingual support:

- English (default), Polish, Spanish, French, German, etc.
- Language parameter w API call

OCR - Read API

Ekstrakcja tekstu z obrazów
Charakterystyka:

- Printed i handwriting recognition
- Multi-language support (140+ języków)
- Orientation detection (auto-rotation)
- Bounding boxes dla words i lines
- Confidence scores

Use case:

- Document digitization
- Receipt/invoice scanning
- License plate recognition
- Street sign reading
- Handwritten notes extraction

Output structure:

- Pages → Lines → Words
- Każdy element ma: text, bbox, confidence
- Preserves layout structure

Vision Studio - Testowanie

Workflow:

1. Vision Studio → Image Analysis
2. Upload images (różne jakości - test robustness)
3. Przejrzyj results:
    - Tags (keywords)
    - Descriptions (sentences)
    - Dense Captions (regional descriptions)
    - OCR output (extracted text)
4. Zmień język opisu - compare quality
5. Test różne typy obrazów (outdoor, indoor, people, objects)

3.2 Integracja z . NET

REST API vs SDK

REST API:

- HttpClient do wywołania endpoint
- POST request z obrazem (binary lub URL)
- Headers: Ocp-Apim-Subscription-Key
- JSON response z results

SDK (NuGet):

- `Azure.AI.Vision.ImageAnalysis`
- ImageAnalysisClient (endpoint, credential)
- Strongly-typed methods
- Easier error handling

Batch vs Single:

- Single - analyze jeden obraz na raz
- Batch - multiple images (async processing, lower cost per image)

Minimal API Example - . NET 9

Endpoint: `POST /analyze-image`
Configuration:

- IOptions<VisionSettings> - endpoint, key z appsettings
- Key Vault integration dla production

Flow:

1. Client uploads image (multipart/form-data)
2. API wywołuje Vision REST endpoint
3. Przetwarza JSON response
4. Zwraca formatted result (tags, descriptions, OCR)

Error handling:

- Invalid image format → 400 Bad Request
- API quota exceeded → 429 Too Many Requests
- Unauthorized → 401 (check key)

4 Custom Vision

4.1 Podstawy

Custom Vision Service

Trenowanie własnych modeli computer vision
2 typy projektów:

- **Classification** - klasyfikacja obrazu (jedna lub wiele etykiet)
- **Object Detection** - lokalizacja i klasyfikacja obiektów

Portal: customvision.ai
Zasoby Azure:

- Custom Vision Training - do trenowania modeli
- Custom Vision Prediction - do wdrożenia i inference
- Możliwe combined resource (training + prediction)

4.2 Classification

Klasyfikacja Obrazów

Multiclass:

- Jedna etykieta na obraz
- Przykład: "dog"LUB "cat"LUB "bird"(nie może być kilka jednocześnie)
- Model wybiera najbardziej prawdopodobną klasę

Multilabel:

- Wiele etykiet na obraz
- Przykład: "beach"+ "sunset"+ "people"(wszystkie mogą być true)
- Independent probability dla każdego tagu

Kiedy który:

- Multiclass - mutually exclusive categories
- Multilabel - multiple attributes możliwe jednocześnie

Training Data - Classification

Minimum:

- 5 obrazów na tag (absolute minimum)
- Recommended: 30-50 obrazów na tag
- Ideally: 100+ obrazów na tag dla production

Balans klas:

- Podobna liczba obrazów dla każdego tagu
- Imbalance prowadzi do bias (model preferuje częstsze klasy)
- Jeśli imbalance nieunikniony: use class weights

Różnorodność:

- Różne kąty (front, side, top views)
- Różne oświetlenie (day, night, indoor, outdoor)
- Różne tła (isolated vs cluttered)
- Różne odległości (close-up, far away)

Augmentacje:

- Custom Vision automatycznie augmentuje
- Rotation, flip, color adjustments, zoom
- Zwiększa robustness bez dodatkowych obrazów

4.3 Object Detection

Detekcja Obiektów

Lokalizacja + klasyfikacja obiektów
Proces tagowania:

- Rysuj bounding boxes wokół obiektów
- Assign tag do każdego boxa
- Możliwe wiele obiektów tego samego typu na jednym obrazie
- Tight bounding boxes (bez zbędnego marginesu)

Training data:

- Minimum: 15 obrazów na tag (z labeled objects)
- Recommended: 50+ obrazów
- Ideally: 200+ obrazów dla production

Różnorodność:

- Różne rozmiary obiektów (small, medium, large)
- Occlusion scenarios (częściowo zakryte)
- Overlapping objects
- Cluttered backgrounds

4.4 Trening

Quick Training vs Advanced Training

Quick Training:

- Czas: kilka minut
- Mniejsza dokładność
- Fewer epochs
- Use case: prototyping, testing data quality, iteracja szybka

Advanced Training:

- Czas: do godziny (zależnie od data size)
- Wyższa dokładność
- More epochs, hyperparameter tuning
- Use case: production models, final deployment

Iteracje:

- Każdy trening tworzy nową iterację
- Wersjonowanie modeli (iteration 1, 2, 3...)
- Możesz compare performance między iteracjami
- Możesz revert do previous iteration
- Publish konkretną iterację do prediction endpoint

Metryki - Classification

Precision (Dokładność):

- % poprawnych positive predictions
- Precision = TP / (TP + FP)
- High precision = few false positives

Recall (Pokrycie):

- % znalezionych positive cases
- Recall = TP / (TP + FN)
- High recall = few false negatives

Trade-off:

- Increasing precision często decreases recall
- Adjust threshold aby balansować

Target values:

- Precision > 0.85 (85%)
- Recall > 0.85
- Depends on use case (medical = high recall, spam = high precision)

Metryki - Object Detection

mAP (mean Average Precision):

- Primary metric dla object detection
- Combines precision i recall across all classes
- IoU threshold (Intersection over Union) - typically 0. 5

IoU (Intersection over Union):

- Miara overlap between predicted i ground truth bbox
- IoU = Area of Overlap / Area of Union
- IoU $>$ 0. 5 = good detection (często used threshold)
- IoU > 0.75 = excellent detection

Target:

- mAP > 0.70 (70%) - good model
- mAP > 0.85 - excellent model
- Depends on complexity (simple objects easier than complex)

4.5 Wdrożenie

Prediction Endpoint (SaaS)

Cloud-hosted inference:

- Publish iteration → creates REST endpoint
- Automatic scaling
- Pay per prediction
- Easy updates (deploy new iteration)

API calls:

- POST request z obrazem (binary lub URL)
- Headers: Prediction-Key
- JSON response: predictions z probabilities

Threshold adjustment:

- Default: 0.5 (50% confidence)
- Higher threshold → more precision, less recall
- Lower threshold → more recall, less precision
- Adjust based on use case requirements

Export "On-Edge"

Offline inference (bez cloud connectivity)
Supported formats:

- **ONNX** - Open Neural Network Exchange (universal)
- **TensorFlow** - pełny model
- **TensorFlow Lite** - mobilne (Android/iOS, optimized)
- **CoreML** - iOS/macOS native
- **NCNN** - Android optimized (bardzo szybki)

Use case:

- IoT devices (offline capability)
- Mobile apps (MAUI, native iOS/Android)
- Edge computing (low latency critical)
- Privacy concerns (data stays on device)
- Cost optimization (no per-prediction charge)

Trade-offs:

- (+) Offline, low latency, no recurring cost
- (-) Manual model updates, limited device resources

4.6 Operacje i Maintenance

Zarządzanie Modelem

Iteracje:

- Train new iteration gdy dodajesz więcej danych
- Compare metrics między iteracjami
- Unpublish old iteration, publish new
- Keep historical iterations (for rollback)

Threshold tuning:

- Test różne progi confidence
- Monitor false positives vs false negatives
- A/B testing w production (compare thresholds)

Data drift:

- Zmiana danych w czasie (nowe scenariusze, oświetlenie, obiekty)
- Performance degradation
- Rozwiązanie: regularny retraining z nowymi danymi
- Monitoring: track prediction confidence over time

Adding new data:

- Upload nowe obrazy (especially edge cases)
- Retrain model (Quick Training dla fast iteration)
- Evaluate improvement
- Deploy jeśli better performance

5 Pytania kontrolne

1. Czym różni się Realtime STT od Batch STT?
2. Jaki format audio jest recommended dla STT i dlaczego?
3. Co to jest Neural TTS i jakie ma zastosowania?
4. Wymień 3 tagi SSML i ich funkcje.
5. Co to jest Custom Speech i kiedy go używać?
6. Wymień 5 prebuilt models w Document Intelligence.
7. Co to jest Compose w Document Intelligence?
8. Czym różnią się Tags, Descriptions i Dense Captions?
9. Co to jest Read API i do czego służy?
10. Czym różni się Classification Multiclass od Multilabel?

11. Ile minimum obrazów potrzeba do treningu Classification?
12. Co to jest mAP w Object Detection?
13. Czym różni się Quick Training od Advanced Training?
14. Jakie formaty eksportu są dostępne w Custom Vision?
15. Co to jest data drift i jak sobie z nim radzić?