OKIEM EKSPERTA

Architektura systemów AI

Projektowanie skalowalnego i niezawodnego oprogramowania

Richard D. Avila
Imran Ahmad

Helion
packt

Tytuł oryginału: Architecting AI Software Systems: Crafting robust
and scalable AI systems for modern software development

Tłumaczenie: Grzegorz Werner

ISBN: 978-83-289-3855-7

Drogi Czytelniku!
Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres
helion.pl/user/opinie/arsyai
Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Helion S.A.
ul. Kościuszki 1c, 44-100 Gliwice
tel. 32 230 98 63
e-mail: *helion@helion.pl*
WWW: *helion.pl* (księgarnia internetowa, katalog książek)

Printed in Poland.

Spis treści

CZĘŚĆ 2. Architektura systemów AI

Podstawy architektury systemów AI

Rozdział 1

Ostatni wzrost zainteresowania **sztuczną inteligencją** (ang. *Artificial Intelligence*, AI), szczególnie w kontekście generatywnej AI, wywołał falę entuzjazmu i zapotrzebowania na kompleksowe rozwiązania AI. Zainteresowanie to wykracza poza środowisko entuzjastów technologii i badaczy, obejmując firmy, rządy i osoby prywatne, które chcą wykorzystać potencjał AI do rozwiązywania rzeczywistych problemów i zwiększania możliwości. W tym kontekście architektura systemów AI, określająca ich strukturę, komponenty i interakcje, odgrywa kluczową rolę w kształtowaniu rozwoju i we wdrażaniu efektywnych rozwiązań AI.

AI stała się siłą transformacyjną, rewolucjonizując branże i zmieniając sposób, w jaki wchodzimy w interakcje z technologią i otaczającym nas światem. W swojej istocie AI odnosi się do modeli obliczeniowych, które naśladują ludzkie funkcje poznawcze, w tym uczenie się na podstawie danych, rozpoznawanie wzorców, podejmowanie decyzji, a nawet interakcję z otoczeniem. Ta rewolucyjna technologia obejmuje szerokie spektrum, od prostych systemów opartych na regułach po zaawansowane modele uczenia głębokiego, z których każdy ma unikatowe zastosowania i możliwości.

Kluczowym aspektem każdego systemu AI jest to, że wyniki przeprowadzanego wnioskowania muszą być trafne i godne zaufania. Aby zdobyć i utrzymać zaufanie użytkowników, trzeba zastosować solidną architekturę. Projektuje się nie tylko technologię, ale także sposób jej wykorzystania, zarządzania nią i oceniania jej przez różnych interesariuszy. Interesariusze muszą mieć możliwość precyzyjnego identyfikowania problemów, szybkiego korygowania parametrów modelu i wdrażania zmian w przemyślany i sprawny sposób. Mówiąc prościej: architekturę i procesy wspierające można nazwać „barierkami ochronnymi". Sposób wykorzystania tych barierek jest bardzo specyficzny dla danej dziedziny i przypadku użycia technologii AI. Można wyróżnić kilka klas barierek ochronnych, np.: wykorzystanie tzw. kanarków do oceny poprawności modelu według znanego złotego standardu, użycie miary czasu i przepływu danych do oceny wydajności modelu oraz zastosowanie filtrów i solidnej kontroli jakości danych, żeby do systemu trafiały tylko spójne i poprawne dane. Inną klasą barierek ochronnych są interfejsy między człowiekiem a systemem, takie jak platformy alarmowania do klasyfikowania i monitorowania błędów, narzędzia do radzenia sobie z problemami oraz ustalone protokoły obsługi nieoczekiwanych błędów. Pisemne procedury lub wytyczne dotyczące modelowania pozwalają na utrzymanie sprawności systemu bez proszenia twórcy modelu o rozwiązywanie problemów.

Zaufanie jest kluczowym czynnikiem sukcesu systemu, więc musi zostać uwzględnione w jego architekturze. Zagadnienia poruszane w tej książce pod wieloma względami dotyczą wzbudzenia zaufania do systemu AI.

W tym rozdziale podkreślimy kluczowe aspekty architektury AI, które decydują o udanym wdrożeniu AI. Omawiane tematy to:

- Wprowadzenie i kluczowe koncepcje AI.
- Komponenty systemu AI.
- Technologie AI i mikrousługi.
- Systemy AI i kwestie techniczne.
- Kwestie wdrożeniowe.

Wprowadzenie do systemów AI: projektowanie przyszłości inteligencji

Systemy AI są ucieleśnieniem sztucznej inteligencji i funkcjonują jako silniki napędzające inteligentne aplikacje i usługi. Są to złożone konstrukcje, starannie zaprojektowane do wykonywania różnorodnych zadań, od rozpoznawania obrazów i przetwarzania języka naturalnego po autonomiczne podejmowanie decyzji i eliminowanie skomplikowanych problemów.

Architektura systemu SI to szczegółowy plan techniczny określający jego strukturalną organizację i interakcje między różnymi komponentami. Do tych komponentów należą:

- **Infrastruktura sprzętowa.** Procesory główne (CPU) do ogólnego przetwarzania, procesory graficzne (GPU) do obliczeń równoległych, procesory tensorowe (TPU) do operacji na tensorach oraz wyspecjalizowane akceleratory AI.
- **Platformy programistyczne.** TensorFlow, PyTorch, JAX i inne biblioteki umożliwiające tworzenie modeli.
- **Implementacje algorytmów.** Algorytmy uczenia maszynowego, architektury sieci neuronowych i mechanizmy wnioskowania.
- **Potoki danych.** Procesy ETL, magazyny cech i systemy zarządzania danymi.

Wszystkie te elementy współpracują ze sobą, aby system mógł efektywnie i niezawodnie realizować zaprojektowane cele.

Dobrze zaprojektowany system SI spełnia kilka kluczowych wymagań technicznych:

- **Optymalna wydajność.** System maksymalizuje efektywność obliczeniową, aby dostarczać dokładne wyniki z minimalnymi opóźnieniami. Obejmuje to zoptymalizowany projekt modelu, efektywną alokację zasobów oraz implementacje uwzględniające specyfikę sprzętu, które w pełni wykorzystują dostępne możliwości obliczeniowe.
- **Skalowalność.** System radzi sobie z rosnącymi obciążeniami i powiększającymi się zbiorami danych poprzez skalowanie poziome (dodawanie większej liczby

maszyn) i pionowe (dodawanie wydajniejszych maszyn) bez pogorszenia wydajności. Nowoczesne architektury AI muszą dostosowywać się do rosnących ilości danych, liczby użytkowników i wymagań obliczeniowych.

- **Efektywność.** System zmniejsza zużycie zasobów obliczeniowych i energii oraz koszty operacyjne poprzez techniki takie jak kwantyzacja modeli, destylacja wiedzy i zoptymalizowane ścieżki wnioskowania. Efektywne systemy AI minimalizują wykorzystanie zasobów przy zachowaniu skuteczności funkcjonalnej.
- **Niezawodność.** System zapewnia spójne działanie i wysoką dostępność, nawet w obliczu nieoczekiwanych wzorców danych, zmian na wejściu czy awarii systemu. Wymaga to solidnej obsługi błędów, możliwości kontrolowanej degradacji i kompleksowych systemów monitorowania. Ponieważ technologie AI mogą być zarówno deterministyczne, jak i niedeterministyczne, należy uwzględnić możliwość interwencji człowieka. Interwencja ta powinna obejmować zakres od prostego monitorowania po pełny pakiet infrastruktury testowej.
- **Bezpieczeństwo.** System wdraża kompleksowe środki ochrony danych i broni się przed atakami adwersaryjnymi, zatruciem danych i lukami w zabezpieczeniach modeli. Systemy AI muszą zachowywać poufność i integralność danych oraz być odporne zarówno na tradycyjne cyberzagrożenia, jak i ataki specyficzne dla AI.
- **Wyjaśnialność.** System daje wgląd w procesy decyzyjne algorytmów, aby zapewnić zgodność z przepisami, zaufanie użytkowników i możliwość debugowania systemu. Nowoczesne architektury AI muszą równoważyć skuteczność z interpretowalnością, by sprostać rosnącym wymaganiom dotyczącym transparentności AI.

Dziedzina sztucznej inteligencji nieustannie ewoluuje, a nowe architektury i technologie pojawiają się w szybkim tempie. Zagłębiając się w tę fascynującą dziedzinę, będziemy badać różne typy systemów AI, ich podstawowe zasady oraz liczne zastosowania, które kształtują przyszłość technologii i społeczeństwa.

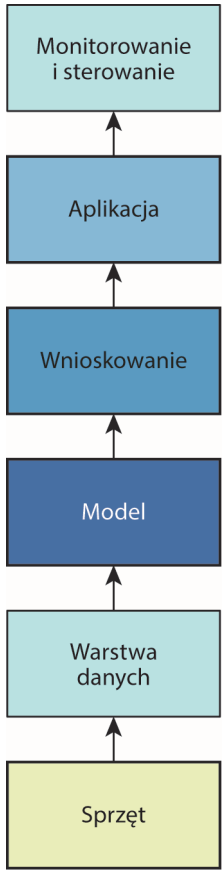
Czym jest system AI?

System AI to model obliczeniowy lub zbiór modeli zaprojektowanych do wykonywania zadań, które zazwyczaj wymagają ludzkiej inteligencji. Systemy te są napędzane przez algorytmy i dane, co umożliwia im uczenie się na podstawie doświadczeń, dostosowywanie do nowych informacji oraz podejmowanie decyzji lub prognozowanie.

Z perspektywy implementacyjnej systemy AI zazwyczaj składają się z kilku kluczowych warstw (rysunek 1.1):

1. **Sprzęt.** Obejmuje zasoby obliczeniowe, takie jak procesory CPU, GPU i TPU, pamięć masową i sieć.
2. **Warstwa danych.** Odpowiada za pobieranie, przechowywanie i wstępne przetwarzanie danych oraz za inżynierię cech.

3. **Warstwa modelu.** Zawiera wytrenowane modele uczenia maszynowego lub głębokiego.
4. **Warstwa wnioskowania.** Zarządza wykonywaniem modeli na nowych danych wejściowych.
5. **Warstwa aplikacji.** Integruje możliwości AI z aplikacjami dla użytkowników.
6. **Warstwa monitorowania.** Śledzi działanie systemu, dryf danych i kondycję modelu.

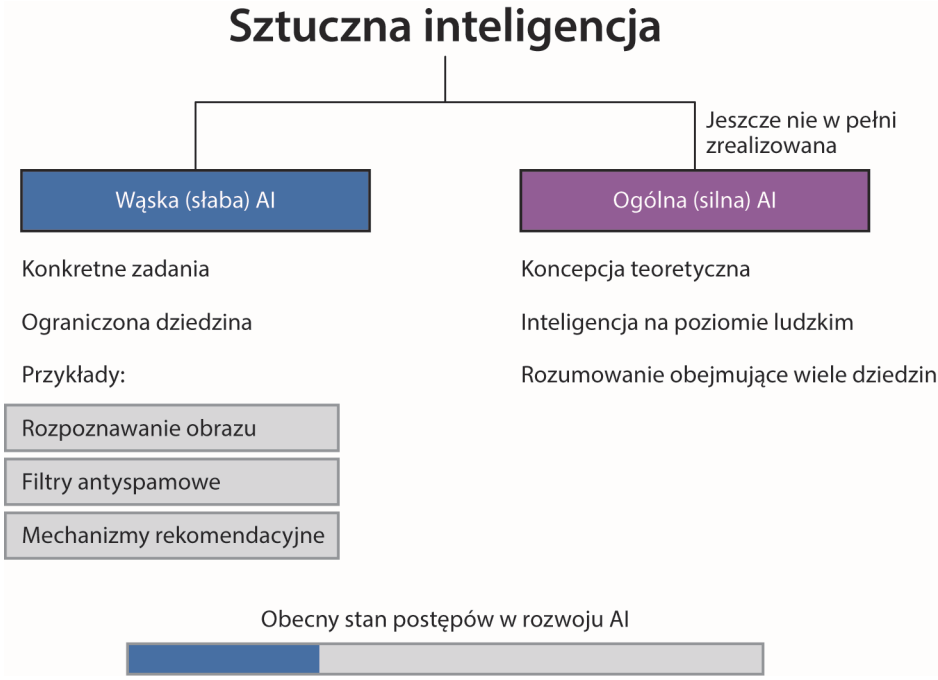


Rysunek 1.1. Stos technologii sztucznej inteligencji

Systemy AI można podzielić na dwie szerokie kategorie (rysunek 1.2):

- **Wąska sztuczna inteligencja (słaba AI).** Systemy te są zaprojektowane do idealnego wykonywania konkretnych zadań w ograniczonym zakresie. Przykłady obejmują oprogramowanie do rozpoznawania obrazów, filtry antyspamowe i mechanizmy rekomendacyjne. Choć systemy te często sprawdzają się doskonale w wyznaczonych zadaniach, nie potrafią uogólniać swojej wiedzy na inne dziedziny.

- **Ogólna sztuczna inteligencja (silna AI).** Jest to teoretyczna koncepcja systemu AI, który posiada inteligencję na poziomie ludzkim i może wykonywać dowolne zadania intelektualne. Taki system miałby zdolność rozumowania, planowania, rozwiązywania problemów, uczenia się na podstawie doświadczeń oraz rozumienia złożonych idei z różnych dziedzin. Choć ogólna AI pozostaje odległym celem, poczyniono znaczne postępy w rozwoju systemów o coraz bardziej zaawansowanych możliwościach.



Rysunek 1.2. Klasyfikacja systemów AI

Wpływ infrastruktury AI: podstawa inteligentnych rozwiązań w wielu różnych branżach

Dobrze zaprojektowana infrastruktura AI, obejmująca sprzęt, oprogramowanie i sieci wspierające aplikacje sztucznej inteligencji, jest motorem transformacji AI w różnych branżach. Infrastruktura ta umożliwia wdrażanie i skalowanie modeli, algorytmów i platform AI, odblokowując pełny potencjał likwidowania złożonych problemów i dostarczania innowacyjnych rozwiązań.

- **Opieka zdrowotna:**
  - **Szybsza analiza obrazów medycznych.** Wysokowydajne klastry obliczeniowe i specjalistyczne akceleratory sprzętowe umożliwiają szybkie przetwarzanie obrazów medycznych, co ułatwia szybszą i dokładniejszą diagnostykę.

- **Wnioski oparte na danych.** Skalowalna infrastruktura do przechowywania i przetwarzania danych wspomaga analizę ogromnych zbiorów danych pacjentów z wykorzystaniem sztucznej inteligencji, co prowadzi do tworzenia spersonalizowanych planów leczenia i poprawy wyników terapii.
- **Monitorowanie w czasie rzeczywistym.** Chmurowa infrastruktura AI umożliwia ciągłe monitorowanie parametrów życiowych pacjenta i innych danych zdrowotnych, co ułatwia szybkie interwencje i prewencję.

- **Finanse:**
  - **Zaawansowane wykrywanie oszustw.** Platformy przetwarzania rozproszonego i analityki w czasie rzeczywistym umożliwiają modelom AI dokładniejsze i szybsze wykrywanie nieuczciwych transakcji, chroniąc instytucje finansowe i konsumentów.
  - **Zoptymalizowane strategie handlowe.** Algorytmy handlu wysokiej częstotliwości wykorzystują sieci o niskich opóźnieniach i potężne zasoby obliczeniowe do precyzyjnego i wydajnego przeprowadzania transakcji, maksymalizując zyski.
  - **Spersonalizowane usługi finansowe.** Chmurowa infrastruktura AI umożliwia wdrażanie robo-doradców i innych narzędzi opartych na sztucznej inteligencji, które zapewniają indywidualne porady finansowe i usługi dla klientów.
- **Pojazdy autonomiczne:**
  - **Łączenie danych z czujników w czasie rzeczywistym.** Wysokowydajne potoki danych i infrastruktura przetwarzania brzegowego umożliwiają szybkie przetwarzanie danych z kamer, lidarów, radarów i innych czujników, pozwalając pojazdom autonomicznym natychmiast podejmować decyzje.
  - **Ulepszone rozpoznawanie obiektów.** Modele uczenia głębokiego trenowane na ogromnych zbiorach danych i wdrażane na specjalistycznych akceleratorach sprzętowych umożliwiają dokładną i niezawodną identyfikację obiektów w otoczeniu.
  - **Zoptymalizowana nawigacja.** Usługi mapowania i nawigacji oparte na chmurze, w połączeniu z pokładowymi systemami AI, dostarczają pojazdom autonomicznym informacji i wskazówek w czasie rzeczywistym, pozwalając na bezpieczną i efektywną jazdę.

Dalszy rozwój i optymalizacja infrastruktury AI będą odgrywać kluczową rolę w realizacji pełnego potencjału sztucznej inteligencji w rozmaitych branżach. Zapewniając podstawę dla wydajnych i skalowalnych rozwiązań AI, infrastruktura ta może zmienić sposób, w jaki żyjemy i pracujemy.

Kluczowe komponenty architektury systemów AI

Systemy AI to zasadniczo złożone struktury zaprojektowane do naśladowania ludzkich zdolności poznawczych, takich jak uczenie się, rozumowanie i rozwiązywanie problemów. Aby osiągnąć te możliwości, systemy AI opierają się na dobrze zdefiniowanej architekturze, składającej się z kilku powiązanych ze sobą komponentów, z których każdy odgrywa kluczową rolę w ogólnym funkcjonowaniu systemu. Zrozumienie tych komponentów jest podstawą do zrozumienia wewnętrznego działania i potencjału SI.

- **Komponenty danych.** Dane stanowią bazę każdego systemu SI, działając jak surowiec, na podstawie którego system uczy się i doskonali. Mogą występować w różnych formach:
- **Dane ustrukturyzowane.** Zorganizowane w predefiniowanych formatach, takich jak bazy danych i arkusze kalkulacyjne.
- **Dane częściowo ustrukturyzowane.** Częściowo uporządkowane informacje, np. pliki JSON lub XML.
- **Dane nieustrukturyzowane.** Surowe informacje, w tym dokumenty tekstowe, obrazy, nagrania dźwiękowe i pliki wideo.

Jakość, ilość i istotność danych mają duży wpływ na skuteczność systemu AI i jego zdolność do generalizacji, czyli poprawnego działania w nowych sytuacjach.

- **Rozwiązania algorytmiczne.** Algorytmy są silnikami napędzającymi systemy AI, dostarczającymi instrukcji i logiki przetwarzania danych i generowania inteligentnych wyników. Algorytmy uczenia maszynowego, będące podzbiorem algorytmów AI, umożliwiają systemom uczenie się wzorców i zależności na podstawie danych, co pozwala im na przewidywanie, klasyfikowanie lub podejmowanie decyzji. Do popularnych podejść algorytmicznych w produkcyjnych systemach AI należą:
  - **Tradycyjne uczenie maszynowe.** Regresja liniowa, lasy losowe, wzmacnianie gradientowe i maszyny wektorów nośnych.
  - **Uczenie głębokie.** Konwolucyjne sieci neuronowe (ang. *Convolutional Neural Network*, CNN), rekurencyjne sieci neuronowe (ang. *Recurrent Neural Network*, RNN), transformery i grafowe sieci neuronowe.
  - **Uczenie przez wzmacnianie.** Metoda Q-learningu, metody gradientu polityki i architektury aktor-krytyk.

Wybór odpowiednich algorytmów zależy od konkretnej dziedziny problemu, charakterystyki dostępnych danych i wymagań wydajnościowych.

- **Architektury modeli.** Model to zwieńczenie procesu uczenia w systemach sztucznej inteligencji. Są to matematyczne reprezentacje wiedzy wydobytej z danych, zawierające wzorce, zależności i spostrzeżenia odkryte przez algorytmy. Modele te mogą być proste lub złożone, zależnie od charakteru zadania i użytego algorytmu. Architektury modeli można podzielić na:
  - **Proste modele liniowe.** Łatwe do interpretacji, ale o ograniczonych możliwościach.
  - **Modele zespołowe.** Łączące wiele prostszych modeli w celu poprawy wydajności.

    - **Głębokie sieci neuronowe.** Złożone architektury z milionami lub miliardami parametrów.

Po wytrenowaniu modele są używane do przewidywania lub podejmowania decyzji na podstawie nowych, niewidzianych wcześniej danych.

- **Infrastruktura.** Komponent infrastruktury obejmuje zasoby sprzętowe i programowe, które zapewniają moc obliczeniową i środowisko niezbędne do działania systemów sztucznej inteligencji. Kluczowe elementy infrastruktury to:
    - **Zasoby obliczeniowe.** Wydajne serwery, specjalistyczne akceleratory AI (procesory graficzne, procesory tensorowe, układy FPGA) i rozproszone klastry obliczeniowe.
    - **Systemy pamięci masowej.** Skalowalne systemy o wysokiej przepustowości do przechowywania danych treningowych i artefaktów modeli.
    - **Komponenty sieciowe.** Połączenia o niskich opóźnieniach do rozproszonego treningu i wnioskowania.
- **Platformy programistyczne.** Biblioteki oprogramowania, takie jak TensorFlow, PyTorch i Hugging Face, które usprawniają tworzenie i wdrażanie rozwiązań AI.

Zrozumienie tych najważniejszych komponentów i ich interakcji stanowi solidną podstawę do poruszania się po złożonym krajobrazie architektur systemów AI. Poprzez staranne projektowanie i optymalizację każdego komponentu naukowcy i inżynierowie mogą budować systemy AI zdolne do rozwiązywania szerokiego zakresu zadań, od rozpoznawania obrazów i przetwarzania języka naturalnego po autonomiczną jazdę i odkrywanie nowych leków. Integracja możliwości AI z istniejącymi stosami oprogramowania wymaga przemyślanych rozwiązań architektonicznych, które skutecznie wykorzystują inteligencję, jednocześnie zaspokajając unikatowe wymagania stwarzane przez komponenty AI. Te specyficzne wymagania i podejścia architektoniczne stanowią główny temat książki. Ze względu na złożoność systemów AI kluczowe znaczenie ma charakter podejścia wdrożeniowego. W następnym podrozdziale omówimy wykorzystanie architektur mikrousługowych, które zapewniają równowagę między wydajnością a modularnością.

Architektura mikrousługowa: modularne podejście do budowania złożonych systemów AI

W miarę jak systemy sztucznej inteligencji stają się coraz bardziej złożone, tradycyjne monolityczne architektury okazują się nieporęczne i zaczynają ograniczać szybkość rozwoju i elastyczność. Atrakcyjną alternatywą jest architektura mikrousługowa, która rozbija te skomplikowane systemy na mniejsze, niezależne usługi. Każda mikrousługa koncentruje się na konkretnej funkcji i komunikuje się z innymi za pomocą precyzyjnie zdefiniowanych interfejsów API.

Korzyści z wykorzystania mikrousług w sztucznej inteligencji

- **Zwiększona elastyczność.** Zespoły mogą niezależnie rozwijać, wdrażać i aktualizować każdą mikrousługę, stosując najbardziej odpowiednie technologie i języki programowania do poszczególnych zadań. Przyspiesza to cykle rozwojowe i ułatwia eksperymentowanie oraz innowacje.
- **Wyższa skalowalność.** Mikrousługi można skalować poziomo, aby sprostać konkretnym wymaganiom przy optymalnym wykorzystaniu zasobów. Na przykład usługę przetwarzania obrazów można skalować niezależnie od usługi odpowiedzialnej za rozumienie języka naturalnego.
- **Większa odporność i izolacja awarii.** Ewentualna awaria mikrousługi ma ograniczony zasięg, co minimalizuje zakłócenia w całym systemie. Zwiększa to ogólną niezawodność i upraszcza rozwiązywanie problemów.
- **Różnorodność technologiczna.** Architektura mikrousług umożliwia zespołom wykorzystanie najlepszych narzędzi do każdego zadania, wspierając innowacyjność i pozwalając na stopniowe ulepszanie technologii.

Wyzwania związane z architekturą mikrousługową

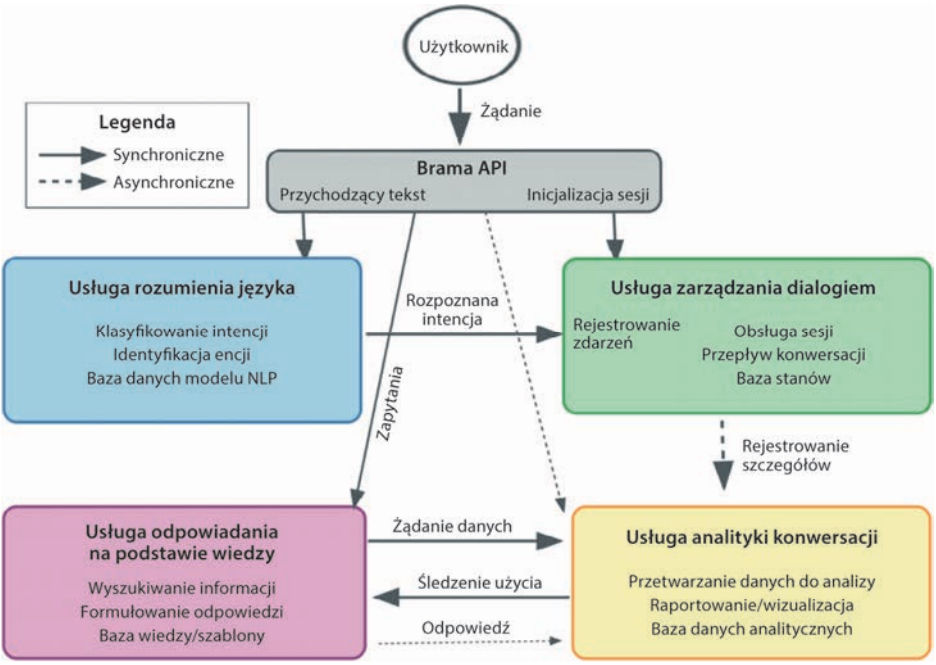
- **Zwiększona złożoność.** Zarządzanie wieloma usługami i ich interakcjami wymaga solidnych narzędzi do orkiestracji i monitorowania. Kluczowe stają się kwestie odkrywania usług, równoważenia obciążenia i obsługi awarii.
- **Narzut komunikacyjny.** Nadmierna komunikacja między usługami może wprowadzać opóźnienia i wpływać na ogólną wydajność systemu. Aby złagodzić ten problem, niezbędne jest staranne projektowanie interfejsów API i wzorców komunikacji.
- **Spójność danych.** Utrzymanie spójności danych w rozproszonych usługach bywa trudne. Zapewnienie integralności danych może wymagać takich strategii jak spójność końcowa lub transakcje rozproszone.

Przykład z życia: wdrażanie konwersacyjnych usług AI w architekturze mikrousługowej

Aby zilustrować, jak podejście oparte na mikrousługach może usprawnić rozwiązanie z zakresu konwersacyjnej AI, zobaczmy praktyczny przykład, który pokazuje, jak te zasady działają w rzeczywistości. W tym podrozdziale omówimy konwersacyjny system AI — taki jak czatbot lub wirtualny asystent — zbudowany z użyciem architektury składającej się z czterech mikrousług z bramą API.

Cztery główne mikrousługi

Ogólną architekturę konwersacyjnego systemu AI przedstawiono na rysunku 1.3.



Rysunek 1.3. Mikrousługi konwersacyjnego systemu AI

Architektura składa się z czterech podstawowych wyspecjalizowanych usług oraz bramy API.

1. Usługa rozumienia języka:
   - **Podstawowe funkcje.** Klasyfikacja intencji, identyfikacja i ekstrakcja encji oraz obsługa modeli NLP.
   - **Dane i modele.** Odwołuje się do baz danych jednego lub wielu modeli NLP (np. klasyfikatorów opartych na transformerach).
   - **Kluczowe interakcje.** Odbiera tekst użytkownika (przez bramę API), określa intencję użytkownika (np. „Sprawdź saldo konta") i wyodrębnia istotne encje (np.: „data", „lokalizacja", „nazwa produktu").
2. Usługa zarządzania dialogiem:
   - **Podstawowe funkcje.** Nadzoruje przebieg rozmowy, obsługuje stan sesji i koordynuje kolejne kroki dialogu.
   - **Dane i stan.** Przechowuje kontekst rozmowy w specjalnej bazie stanu.
   - **Kluczowe interakcje.** Rejestruje zdarzenia konwersacyjne (asynchronicznie) oraz aktualizuje lub wyszukuje szczegóły sesji w celu kierowania przebiegiem rozmowy (np.: „Powitanie", „Potwierdzenie", „Następny krok").

3. Usługa odpowiadania na podstawie wiedzy:
   - **Podstawowe funkcje.** Wyszukuje istotne informacje i formułuje odpowiedzi. Może to obejmować odpytywanie bazy wiedzy (np. FAQ, informacje o produktach) lub tworzenie odpowiedzi na podstawie szablonów.
   - **Dane i szablony.** Przechowuje dane dziedzinowe w bazie wiedzy i wykorzystuje szablony lub mechanizmy generatywne do tworzenia odpowiedzi.
   - **Kluczowe interakcje.** Otrzymuje zapytania od usługi zarządzania dialogiem, znajduje lub komponuje najlepszą odpowiedź i zwraca ją w celu ostatecznego dostarczenia użytkownikowi.
4. Usługa analityki konwersacji:
   - **Podstawowe funkcje.** Przetwarza dzienniki i miary użycia na potrzeby raportowania, wizualizacji i głębszej analizy (np. rozkład intencji, trendy satysfakcji użytkowników).
   - **Dane i raportowanie.** Utrzymuje w oddzielnej bazie danych dane analityczne do tworzenia pulpitów nawigacyjnych lub przetwarzania w trybie offline.
   - **Kluczowe interakcje.** Asynchronicznie gromadzi wpisy dziennika zdarzeń z usługi zarządzania dialogiem i innych komponentów w celu pomiaru wydajności, śledzenia zachowań użytkowników i dostarczania spostrzeżeń, które pomagają ulepszyć system.

Rola bramy API

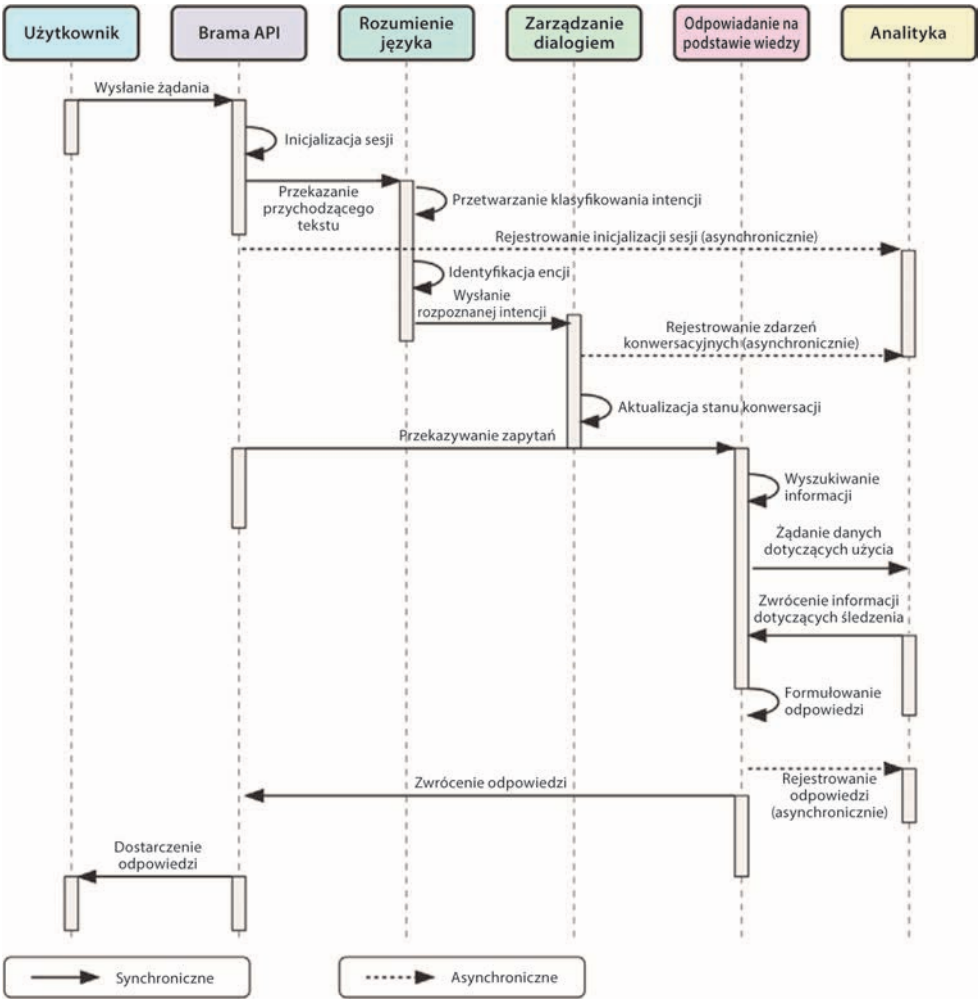
Brama API, choć nie jest zaliczana do czterech głównych mikrousług, stanowi istotny komponent architektury. Pełni ona następujące funkcje:

- Odbiera żądania od użytkownika (za pośrednictwem tekstu lub innych kanałów).
- Inicjalizuje sesję i kieruje przychodzące dane do usługi rozumienia języka.
- Przekazuje rozpoznane intencje i nowe informacje do usługi zarządzania dialogiem.
- Przesyła odpowiedzi z usług podrzędnych z powrotem do użytkownika.

Poprzez centralizację zarządzania ruchem brama API zapewnia spójne zasady bezpieczeństwa, ograniczania przepustowości i monitorowania, jednocześnie izolując każdą mikrousługę i umożliwiając jej niezależne skalowanie.

Sekwencja rozmowy

Aby zilustrować, jak te mikrousługi współdziałają podczas typowej podróży użytkownika, na rysunku 1.4 przedstawiono sekwencję ich wzajemnych wywołań w jednym cyklu konwersacji.



Rysunek 1.4. Diagram sekwencyjny interakcji między komponentami systemu

Sekwencja przebiega następująco:

1. **Użytkownik → brama API.** Użytkownik wysyła żądanie (np. wiadomość czatu). Brama API inicjalizuje sesję (jeśli to konieczne) i przekazuje wiadomość do usługi rozumienia języka.
2. **Usługa rozumienia języka:**
   - Klasyfikuje intencję i identyfikuje encje.
   - Zwraca rozpoznaną intencję (np. „Sprawdź pogodę") oraz wyodrębnione encje (np. data, lokalizacja) do bramy API.
3. **Usługa zarządzania dialogiem:**
   - Odbiera rozpoznaną intencję od bramy API.

- Rejestruje zdarzenia konwersacyjne (asynchronicznie) w usłudze analityki konwersacji.
- Aktualizuje lub wyszukuje **stan sesji** (np. lokalizację użytkownika lub kontekst ostatniej rozmowy).

4. **Usługa odpowiadania na podstawie wiedzy:**
   - Gdy usługa zarządzania dialogiem stwierdzi, że potrzebne są dodatkowe dane (np. informacje o pogodzie, szczegóły produktu), wysyła **zapytanie** do usługi odpowiadania na podstawie wiedzy.
   - Usługa ta wyszukuje niezbędne informacje lub tworzy szablon odpowiedzi (np. „Pogoda w twojej lokalizacji: słonecznie, temperatura 24 °C”).
5. **Usługa analityki konwersacji (rejestrowanie asynchroniczne):**
   - Nieustannie otrzymuje dane dotyczące użycia i wpisy dziennika konwersacji z usługi zarządzania dialogiem (oraz ewentualnie z usługi odpowiadania na podstawie wiedzy).
   - Przetwarza i przechowuje te wpisy dziennika do późniejszego raportowania (np. pulpity miesięcznego użytkowania, miary skuteczności modelu).
6. **Odpowiadanie użytkownikowi:**
   - Sformułowana odpowiedź z usługi odpowiadania na podstawie wiedzy jest przekazywana z powrotem przez usługę zarządzania dialogiem (jeśli to konieczne w celu ostatecznego zaktualizowania sesji), a następnie zwracana przez bramkę API.
   - Użytkownik otrzymuje **odpowiedź** i interakcja dobiega końca.

Kluczowe aspekty komunikacji między mikrousługami

1. Wywołania synchroniczne a asynchroniczne:
   - Żądania wymagające natychmiastowej odpowiedzi (np. generowanie odpowiedzi dla użytkownika) wykorzystują wywołania synchroniczne.
   - Operacje rejestrowania lub analizy danych zazwyczaj wykonuje się asynchronicznie, aby uniknąć spowolnienia głównej pętli konwersacji.
2. Komponenty stanowe a bezstanowe:
   - Zarządzanie dialogiem wymaga śledzenia stanu sesji, podczas gdy inne usługi (np. rozumienie języka) często korzystają ze wzorców bezstanowych, które ułatwiają skalowanie.
   - Usługa zarządzania dialogiem może wymagać solidnych rozwiązań do zarządzania stanem, takich jak rozproszona pamięć podręczna lub bazy danych.
3. Autonomia usług:
   - Każda mikrousługa może być aktualizowana lub wymieniana niezależnie, bez wpływu na resztę systemu.

- Modele NLP w usłudze rozumienia języka mogą wymagać częstego ponownego trenowania. Ponieważ jest to oddzielna usługa, takie aktualizacje można wdrażać bez zakłócania pracy innych usług.

4. Izolacja danych:
   - Usługi zarządzają własnymi danymi dziedzinowymi. Usługa zarządzania dialogiem przechowuje stan rozmowy, usługa odpowiadania na podstawie wiedzy zawiera fakty dotyczące dziedziny, a usługa analityki prowadzi rejestry interakcji.
   - Wrażliwe dane użytkowników nie powinny opuszczać magazynu stanu usługi zarządzania dialogiem, aby można było zminimalizować ich ekspozycję w skali całego systemu.

Kwestie wdrożeniowe związane z konwersacyjnymi mikrousługami AI

1. Niezależne skalowanie:
   - Usługę rozumienia języka można skalować w górę lub w dół zależnie od obciążenia przychodzącymi wiadomościami (np. automatyczne skalowanie poziome w czasie szczytowego ruchu).
   - Usługa zarządzania dialogiem utrzymuje stan rozmowy i może wymagać innych strategii skalowania.
   - Usługa odpowiadania na podstawie wiedzy często skaluje się zgodnie ze złożonością wyszukiwania informacji.
   - Usługa analityki może być skalowana oddzielnie, zwłaszcza jeśli obciążenia analityczne (np. generowanie raportów) występują w innych momentach niż zapytania użytkowników.
2. Zarządzanie opóźnieniami:
   - Konwersacyjne systemy AI dążą do interakcji w czasie zbliżonym do rzeczywistego. Kluczowe jest minimalizowanie liczby przeskoków sieciowych i narzutu komunikacyjnego między usługami. Stosowanie lekkich protokołów komunikacyjnych pomaga zapewnić dobrą wydajność systemu przy dużej skali.
3. Izolacja błędów:
   - Jeśli jedna z usług zawiedzie (np. usługa odpowiadania na podstawie wiedzy przestanie działać), reszta systemu nadal może obsługiwać inne zadania lub oferować rozwiązania awaryjne (np. odpowiedź z przeprosinami lub przekierowanie do człowieka).
4. Monitorowanie i obserwowalność:
   - Solidne praktyki w zakresie rejestrowania zdarzeń i obserwowalności są kluczowe do zapewnienia odporności systemu na awarie lub na spowolnienia usług. Usługa analityki konwersacji odgrywa kluczową rolę w śledzeniu stanu i wydajności systemu.

Dlaczego warto stosować mikrousługi w konwersacyjnych systemach AI?

Podział konwersacyjnego systemu AI na cztery wyspecjalizowane usługi przynosi duże korzyści w zakresie **łatwości utrzymania**, **skalowalności** i **elastyczności**. Każda usługa może ewoluować niezależnie, co umożliwia szybkie modyfikowanie modeli języka naturalnego, przepływów konwersacji i strategii wyszukiwania wiedzy bez ryzyka „wielkiego wybuchu" w całej aplikacji.

Jednocześnie ważne jest zwrócenie szczególnej uwagi na **komunikację między usługami**. Jak pokazuje diagram sekwencyjny, każde żądanie użytkownika wymaga wielu przeskoków. Wykorzystanie lekkich protokołów komunikacyjnych oraz rozróżnienie między operacjami synchronicznymi i asynchronicznymi pomaga utrzymać krótki czas reakcji systemu.

Przykład konwersacyjnego systemu AI pokazuje, że podejście oparte na mikrousługach umożliwia równoważenie elastyczności, odporności na błędy i stopniowego wprowadzania innowacji. Wnioski płynące z tego przykładu — takie jak niezależność skalowania kluczowych usług, izolacja danych w celu zapewnienia bezpieczeństwa oraz kontrolowane tryby awarii — mają szerokie zastosowanie w wielu rozwiązaniach opartych na sztucznej inteligencji.

Ten rzeczywisty wzorzec implementacyjny pokazuje, że choć mikrousługi zwiększają złożoność, to korzyści, jakie przynoszą systemom sztucznej inteligencji — szczególnie tym wymagającym częstych aktualizacji, zmiennego skalowania i innowacji na poziomie poszczególnych komponentów — często przewyższają wady, pod warunkiem że usługi są odpowiednio zaprojektowane i wdrożone.

Aspekty systemu AI

Opracowanie dobrze zaprojektowanej architektury systemu AI wymaga starannego rozważenia kilku głównych czynników. Zapewniają one, że system nie tylko skutecznie działa, lecz także dostosowuje się do przyszłych wymagań i wyzwań.

Skalowalność — radzenie sobie z rosnącą ilością danych i ze złożonością modeli

Systemy sztucznej inteligencji często muszą radzić sobie z rosnącą ilością danych i z coraz bardziej złożonymi modelami. Skalowalność to zdolność systemu do obsługi tego wzrostu bez spadku wydajności. Skuteczne strategie obejmują:

- **Skalowanie poziome.** Polega na dodawaniu większej liczby zasobów obliczeniowych w celu rozłożenia obciążenia. Na przykład w środowisku chmurowym można wdrożyć dodatkowe maszyny wirtualne lub kontenery, aby obsłużyć zwiększony ruch. Kubernetes może zarządzać tymi kontenerami, zapewniając równomierne rozłożenie obciążenia.

- **Skalowanie pionowe.** Ulepszanie istniejących zasobów poprzez zastosowanie wydajniejszego sprzętu. Na przykład: modernizacja procesora lub kart graficznych serwera, dodanie większej ilości pamięci RAM lub użycie SSD zamiast dysków twardych w celu zwiększenia wydajności operacji wejścia-wyjścia.
- **Przetwarzanie rozproszone.** Wykorzystanie platform takich jak Apache Spark lub Hadoop do przetwarzania danych na wielu węzłach. Podejście to polega na podziale dużych zbiorów danych na mniejsze fragmenty, które mogą być przetwarzane równolegle, co znacznie skraca czas przetwarzania. Na przykład funkcja **Resilient Distributed Datasets (RDD)** w Sparku umożliwia przetwarzanie w pamięci, które jest znacznie szybsze niż tradycyjne przetwarzanie oparte na dysku.

Wydajność — techniki optymalizacji

W wielu zastosowaniach AI kluczowe znaczenie ma przetwarzanie w czasie rzeczywistym lub zbliżonym do rzeczywistego. Do technik optymalizacji wydajności należą:

- **Akceleracja sprzętowa.** Wykorzystanie procesorów graficznych (GPU) lub tensorowych (TPU) do zadań intensywnych obliczeniowo — np. TensorFlow i PyTorch mogą stosować rdzenie CUDA w kartach NVIDIA do przyspieszenia treningu modeli uczenia głębokiego.
- **Przetwarzanie równoległe.** Podział zadań na mniejsze podzadania, które mogą być wykonywane jednocześnie. W Pythonie można wykorzystać biblioteki takie jak multiprocessing czy concurrent.futures do zrównoleglenia zadań — np. do jednoczesnego trenowania wielu modeli lub przetwarzania różnych partii danych równolegle.
- **Optymalizacja algorytmów.** Wybieranie lub projektowanie algorytmów o mniejszej złożoności obliczeniowej. Na przykład stosowanie przybliżonych algorytmów wyszukiwania najbliższych sąsiadów zamiast dokładnych metod, które są kosztowne obliczeniowo w przypadku wyszukiwania podobieństw na dużą skalę.

Niezawodność — odporność na awarie, obsługa błędów i redundancja

Niezawodność jest kluczowa, zwłaszcza w zastosowaniach krytycznych. Aby zapewnić ciągłość działania systemu i integralność danych, stosuje się strategie takie jak odporność na awarie, obsługa błędów i redundancja.

- **Odporność na awarie.** System może kontynuować pracę nawet wtedy, gdy niektóre komponenty ulegną awarii. Na przykład w architekturze mikrousługowej, jeśli jedna usługa zawiedzie, pozostałe mogą nadal funkcjonować. Można użyć narzędzi takich jak Hystrix firmy Netflix do implementacji „bezpieczników" zarządzających awariami.

- **Obsługa błędów.** Mechanizmy wykrywania i kontrolowanego korygowania błędów — np. użycie bloków `try-catch` w kodzie do obsługi wyjątków oraz rejestrowanie błędów do późniejszej analizy.
- **Redundancja.** Powielanie krytycznych komponentów w celu uniknięcia pojedynczych punktów podatności na awarię — np. stosowanie konfiguracji RAID dla pamięci dyskowej lub wdrażanie usług w wielu strefach dostępności w środowiskach chmurowych w celu zapewnienia wysokiej dostępności.

Bezpieczeństwo — prywatność danych i odporność modelu

Systemy sztucznej inteligencji często przetwarzają wrażliwe dane, dlatego bezpieczeństwo jest kwestią najwyższej wagi. Do kluczowych kwestii należą:

- **Szyfrowanie danych.** Ochrona danych w spoczynku i tranzycie — np. szyfrowanie algorytmem AES danych przechowywanych w bazach danych oraz zabezpieczanie protokołem TLS danych przesyłanych przez sieć. Należy dokładnie rozważyć i przetestować wpływ zastosowanych metod szyfrowania na wydajność modelu i całego systemu.
- **Kontrola dostępu.** Wdrożenie rygorystycznych mechanizmów uwierzytelniania i autoryzacji — np. wykorzystanie OAuth 2.0 do bezpiecznego dostępu do API oraz **kontroli dostępu opartej na rolach** (ang. *Role-Based Access Control*, RBAC) do zarządzania uprawnieniami.
- **Odporność modelu.** Zabezpieczenie przed atakami, które mogłyby zmanipulować system. Techniki takie jak trening adwersaryjny, w którym model uczy się zarówno na normalnych, jak i na adwersaryjnych przykładach, mogą zwiększyć jego odporność. Dodatkowo można wdrożyć systemy wykrywania anomalii, aby monitorować nietypowe wzorce w danych wejściowych.

Modelowanie danych — katalogi i ontologie

W dziedzinie sztucznej inteligencji dane nie są tylko cennym zasobem, lecz także fundamentem, na którym buduje się inteligentne systemy. Ponieważ modele AI wymagają ogromnych ilości danych do nauki i podejmowania decyzji, kluczowe znaczenie ma skuteczne zarządzanie tymi danymi. Tu na scenę wkraczają katalogi i ontologie danych — narzędzia niezbędne do skutecznego poruszania się po złożonych krajobrazach danych w architekturach AI.

Katalogi to scentralizowane repozytoria metadanych, które dostarczają kompleksowych informacji o zasobach danych w systemie AI. Działają jak wszechstronny indeks, oferując wgląd w lokalizację danych, schemat, pochodzenie, jakość i inne istotne atrybuty. Poprzez konsolidację tych informacji w ustrukturyzowanym i przystępnym formacie katalogi danych umożliwiają specjalistom data science, inżynierom i analitykom głębsze zrozumienie zasobów danych, usprawnienie procesów pracy i zapewnienie nadzoru nad danymi.

Ontologie oferują semantyczną reprezentację elementów danych w konkretnej dziedzinie. Mogą pomóc inżynierom danych w zrozumieniu, jak i dlaczego elementy danych są powiązane, oraz usprawnić procesy przetwarzania. Ontologie zapewniają też specjalistom data science kontekst do tworzenia i aktualizowania modeli.

Omówiliśmy techniczne i funkcjonalne atrybuty systemów AI. Kolejny podrozdział dotyczy różnych sposobów wdrażania systemów w nowoczesnym środowisku chmurowym. Wykorzystanie technologii chmurowej zapewnia elastyczność alokacji zasobów oraz możliwość łatwego skalowania systemu AI w zależności od rzeczywistego zapotrzebowania.

Współczesne paradygmaty wdrażania sztucznej inteligencji

W miarę rozwoju systemów sztucznej inteligencji pojawiły się nowe paradygmaty wdrożeniowe, które odpowiadają na konkretne wymagania i przypadki użycia. W tym podrozdziale omówimy dwa istotne podejścia: chmurową architekturę AI oraz brzegowe wdrożenia AI.

Chmurowe architektury sztucznej inteligencji

Rosnąca złożoność i skala aplikacji AI poskutkowały wprowadzeniem natywnych architektur chmurowych. Architektury te wykorzystują skalowalność, elastyczność i efektywność kosztową platform chmurowych, umożliwiając wydajne tworzenie systemów AI, wdrażanie ich i zarządzanie nimi. W architekturze chmurowej komponenty AI są zaprojektowane tak, aby bezproblemowo działały w środowiskach chmurowych, korzystając ze specjalistycznych usług do przechowywania danych, obliczeń i łączności sieciowej.

Kluczowe cechy architektur AI natywnych dla chmury to:

- **Konteneryzacja.** Aplikacje AI są pakowane w lekkie, przenośne kontenery z użyciem technologii takich jak Docker, co zapewnia spójność środowisk rozwojowego, testowego i produkcyjnego.
- **Orkiestracja.** Platformy do orkiestracji kontenerów, takie jak Kubernetes, zarządzają wdrażaniem, skalowaniem i działaniem kontenerów aplikacji w klastrach hostów.
- **Mikrousługi.** Jak już wspomniano, dzielenie systemów AI na mniejsze, niezależne usługi umożliwia efektywniejsze wykorzystanie zasobów i ułatwia skalowanie.
- **Przetwarzanie bezserwerowe.** Platformy takie jak AWS Lambda, Azure Functions i Google Cloud Functions pozwalają programistom skupić się na pisaniu kodu bez martwienia się o infrastrukturę, co jest szczególnie przydatne w przypadku zadań AI sterowanych zdarzeniami.

- **Usługi zarządzane.** Dostawcy chmury oferują specjalistyczne usługi AI, takie jak w pełni zarządzane platformy uczenia maszynowego (np.: Amazon SageMaker, Microsoft Azure ML, Google Vertex AI), które usprawniają proces rozwoju i wdrażania.
- **Natywne rozwiązania chmurowe a systemy przenoszone bez zmian.** Komponenty AI działające natywnie w chmurze są specjalnie zaprojektowane tak, aby wykorzystywały zalety środowisk chmurowych, takie jak automatyczne skalowanie, przetwarzanie bezserwerowe i usługi zarządzane. Podejście to oferuje większą elastyczność, skalowalność i efektywność kosztową w porównaniu ze zwykłym przeniesieniem istniejących lokalnych systemów AI do chmury bez modyfikacji architektonicznych.

Jeziora danych i hurtownie danych w architekturach AI — fundament inteligencji opartej na danych

W dziedzinie sztucznej inteligencji dane stanowią fundament innowacyjności i postępu. Modele AI bazują na ogromnych zbiorach danych, używając ich do rozpoznawania wzorców, tworzenia prognoz i generowania cennych spostrzeżeń. Jednak efektywne zarządzanie tak dużymi ilościami danych w projektach AI wymaga specjalistycznych rozwiązań do przechowywania i przetwarzania informacji. W tym kontekście powstały dwa kluczowe pojęcia: jeziora danych i hurtownie danych.

Jezioro danych — ogromny rezerwuar nieprzetworzonych informacji

Jeziora danych to obszerne repozytoria, w których przechowywane są surowe dane w ich pierwotnym formacie. Są one zaprojektowane do przechowywania danych ustrukturyzowanych, częściowo ustrukturyzowanych i nieustrukturyzowanych z różnorodnych źródeł. Elastyczność jezior danych sprawia, że idealnie nadają się one do przechowywania dużych ilości danych, które mogą nie mieć z góry określonego celu lub struktury.

- **Kluczowe cechy:**
  - **Schemat określany podczas odczytu (ang. *schema-on-read*).** Jeziora danych nie narzucają ścisłego schematu podczas wprowadzania danych, co zapewnia elastyczność w zakresie typów i struktur danych. Schemat jest definiowany podczas analizy lub przetwarzania, umożliwiając użytkownikom dostosowanie się do zmieniających się wymagań dotyczących danych.
  - **Efektywna kosztowo skalowalność.** Jeziora danych można łatwo skalować, aby pomieścić rosnące ilości danych, co czyni je ekonomicznym rozwiązaniem do przechowywania ogromnych zbiorów danych.

- **Obsługa różnorodnych danych.** Jeziora danych mogą obsługiwać szeroki zakres danych, w tym odczyty z czujników, kanały społecznościowe, pliki dzienników i wiele innych.
- **Idealne do analizy eksploracyjnej.** Jeziora danych zapewniają naukowcom i analitykom danych bogate środowisko do eksplorowania danych, identyfikowania wzorców i generowania hipotez.

- **Przykładowe zastosowania:**
  - Firma e-commerce może przechowywać w jeziorze danych dane o kliknięciach, opinie klientów i interakcje społecznościowe do późniejszej analizy i personalizacji.
  - Organizacja opieki zdrowotnej mogłaby wykorzystać jezioro danych do przechowywania obrazów medycznych, elektronicznej dokumentacji medycznej i danych genomowych do badań i rozwoju narzędzi diagnostycznych opartych na sztucznej inteligencji.

Hurtownie danych — ustrukturyzowane repozytoria do celów analitycznych

Hurtownie danych to ustrukturyzowane repozytoria przechowujące przetworzone i uporządkowane dane, przekształcone do spójnego formatu na potrzeby analiz i raportowania. Można tworzyć i rozwijać ontologie w celu organizowania danych wprowadzanych do systemu i nadawania im semantycznej struktury. Ontologie zapewniają również mechanizm lepszego zarządzania modelem i kontrolowania jego skuteczności poprzez jawne określanie relacji między elementami danych.

Hurtownie danych ułatwiają efektywne odpytywanie i analizowanie informacji, co czyni je niezbędnymi do zastosowań związanych z analityką biznesową i ze wspomaganiem podejmowania decyzji.

- **Kluczowe cechy:**
  - **Schemat określany podczas zapisu (ang. *schema-on-write*).** Hurtownie danych wymuszają predefiniowany schemat podczas wprowadzania danych, zapewniając ich spójność i integralność.
  - **Zoptymalizowane pod kątem zapytań.** Hurtownie danych wykorzystują zoptymalizowane struktury danych i techniki indeksowania, aby usprawnić wyszukiwanie i analizowanie informacji, co przyspiesza wyciąganie wniosków.
  - **Obsługa danych ustrukturyzowanych.** Hurtownie danych są zaprojektowane do pracy z danymi ustrukturyzowanymi, takimi jak dane transakcyjne, informacje o klientach i dokumenty finansowe.
  - **Idealne do analityki biznesowej.** Hurtownie danych umożliwiają organizacjom generowanie raportów, pulpitów nawigacyjnych i wizualizacji na potrzeby podejmowania decyzji.

- **Przykładowe przypadki użycia:**
  - Instytucja finansowa może wykorzystywać hurtownię danych do przechowywania informacji o transakcjach, danych klientów oraz trendów rynkowych na potrzeby analizy ryzyka i wykrywania oszustw.
  - Firma produkcyjna może korzystać z hurtowni danych do analizowania danych produkcyjnych, wskaźników łańcucha dostaw oraz opinii klientów w celu optymalizacji działań i poprawy jakości produktów.

Synergia jezior danych i hurtowni danych

W wielu architekturach AI jeziora danych i hurtownie danych uzupełniają się wzajemnie. Surowe dane są najpierw wprowadzane do jeziora danych, gdzie przechodzą proces czyszczenia, transformacji i wzbogacania. Następnie przetworzone dane są przenoszone do hurtowni danych w celu dalszych analiz i raportowania. To synergiczne podejście umożliwia organizacjom wykorzystanie elastyczności jezior danych do eksploracji danych oraz struktury hurtowni danych do wspomagania decyzji, tworząc solidny fundament pod aplikacje AI oparte na danych.

Sztuczna inteligencja w chmurze — przełom w dziedzinie AI

Konwergencja sztucznej inteligencji i przetwarzania w chmurze otworzyła nowe możliwości dla organizacji chcących wykorzystać potencjał AI. Chmura zapewnia skalowalną, elastyczną i efektywną kosztowo infrastrukturę do tworzenia, wdrażania i skalowania aplikacji AI. Dzięki jej możliwościom firmy mogą pokonać ograniczenia tradycyjnych, działających lokalnie rozwiązań AI i przyspieszyć wprowadzanie innowacji.

Zalety chmurowej sztucznej inteligencji

Chmurowa sztuczna inteligencja ma kilka kluczowych zalet, które czynią ją atrakcyjną opcją dla organizacji każdej wielkości:

- **Skalowalność.** Zasoby chmurowe można łatwo skalować w górę lub w dół, aby sprostać zmiennym wymaganiom obciążeń związanych z AI. Elastyczność ta pozwala organizacjom obsługiwać duże zbiory danych, trenować złożone modele i przetwarzać ogromne ilości informacji bez konieczności inwestowania w kosztowną infrastrukturę sprzętową i jej utrzymywanie.
- **Elastyczność.** Platformy chmurowe oferują szeroki zakres usług i narzędzi AI, dając organizacjom swobodę wyboru najlepszych opcji dla ich konkretnych potrzeb. Umożliwia to firmom eksperymentowanie z różnymi podejściami do AI, szybkie wypróbowywanie kolejnych wersji modeli i dostosowywanie się do zmiennych wymagań.

- **Efektywność kosztowa.** AI w chmurze może być bardziej opłacalna niż rozwiązania lokalne. Organizacje płacą tylko za wykorzystywane zasoby, co eliminuje potrzebę początkowych inwestycji kapitałowych w sprzęt i oprogramowanie. Poza tym dostawcy usług chmurowych często oferują modele „płatności w miarę użycia", co może jeszcze bardziej obniżyć koszty.

Wykorzystując moc sztucznej inteligencji w chmurze, organizacje mogą osiągnąć nowy poziom innowacyjności, wydajności i konkurencyjności.

Główne chmurowe platformy AI — przyspieszanie innowacji dzięki kompleksowym zestawom narzędzi

Najwięksi dostawcy usług chmurowych stali się kluczowymi graczami w obszarze sztucznej inteligencji, oferując całe zestawy narzędzi i usług AI, które zaspokajają szeroki zakres potrzeb. Platformy te stanowią kompleksowe rozwiązanie dla firm i programistów, którzy chcą zastosować możliwości sztucznej inteligencji w swoich aplikacjach i przepływach pracy.

Kluczowe chmurowe platformy AI

- **Google Cloud AI (Vertex AI).** Ta ujednolicona platforma usprawnia cały cykl **uczenia maszynowego** (ang. *Machine Learning*, ML), od budowania i trenowania modeli po ich wdrażanie i zarządzanie nimi w środowisku produkcyjnym. Funkcja AutoML w Verteksie AI upraszcza tworzenie modeli dla użytkowników o ograniczonej wiedzy z zakresu uczenia maszynowego, podczas gdy biblioteka modeli oferuje zbiór wstępnie wytrenowanych modeli gotowych do wdrożenia. Mechanizm Vertex AI Pipelines koordynuje złożone procesy ML, umożliwiając efektywne eksperymentowanie i automatyzację.
- **Amazon SageMaker.** W pełni zarządzana usługa, która umożliwia użytkownikom budowanie, trenowanie i wdrażanie modeli ML na dużą skalę. Oferuje szeroki wybór wbudowanych algorytmów i rozwiązań, dzięki czemu jest przydatna zarówno dla początkujących, jak i doświadczonych użytkowników. Skalowalność SageMakera i integracja z innymi usługami AWS sprawiają, że jest to popularny wybór dla rozwiązań AI klasy korporacyjnej.
- **Amazon Bedrock.** Ta nowoczesna usługa demokratyzuje dostęp do **modeli podstawowych** (ang. *Foundation Model*, FM), od czołowych start-upów AI i samego Amazona po prosty interfejs API. Bedrock pozwala programistom wykorzystać moc najnowocześniejszych możliwości generatywnej AI bez konieczności budowania i trenowania złożonych modeli od podstaw.
- **Microsoft Azure AI.** Ta platforma oferuje szeroki zakres usług AI, w tym gotowe modele AI do widzenia komputerowego, rozpoznawania mowy, przetwarzania języka naturalnego i podejmowania decyzji. Azure Machine Learning pozwala użytkownikom tworzyć i wdrażać własne modele AI, a ścisła integracja platformy z innymi usługami Azure czyni ją wszechstronnym wyborem dla różnorodnych zastosowań AI.

Te chmurowe platformy AI zapewniają organizacjom użyteczny i przystępny sposób na wykorzystanie AI w codziennej działalności, przyspieszając wprowadzanie innowacji i zwiększając wartość biznesową.

Podsumowanie

W tym rozdziale zbadaliśmy fundamentalne zasady architektury systemów sztucznej inteligencji, aby przygotować ramy pojęciowe niezbędne do zrozumienia elementów składowych inteligentnych systemów. Przyjrzeliśmy się kluczowym komponentom — danym jako krwiobiegowi systemu, strukturom algorytmicznym umożliwiającym uczenie się, modelom zawierającym inteligencję oraz infrastrukturze dostarczającej zasoby obliczeniowe — wraz z wzorcami architektonicznymi, takimi jak mikrousługi, które zapewniają modularność i elastyczność. Omówiliśmy krytyczne kwestie projektowe, w tym skalowalność, wydajność, niezawodność i bezpieczeństwo, jako niezbędne składniki solidnych systemów AI, które mogą rozwijać się wraz ze wzrostem wymagań, jednocześnie pozostając odporne i bezpieczne.

Szybko rozwijają się środowiska wdrożeniowe systemów AI, które wykorzystują natywną architekturę chmurową, konteneryzację, orkiestrację i przetwarzanie bezserwerowe do osiągnięcia niespotykanej dotąd efektywności. Synergia między jeziorami danych, hurtowniami danych i katalogami danych tworzy solidną podstawę dla analityki danych, podczas gdy największe platformy chmurowe demokratyzują dostęp do zaawansowanych możliwości AI. W przyszłości te fundamentalne zasady będą kierować rozwojem systemów AI, które staną się nie tylko potężne, lecz także skalowalne, niezawodne i bezpieczne, co pozwoli na wprowadzenie nowej generacji innowacji w różnych branżach.

Polecana lektura

- L. Bass, P. Clements, R. Kazman, *Architektura oprogramowania w praktyce. Wydanie IV*, Helion, 2022.
- D. Weyns, *Software Architecture: Principles and Practices*, MIT Press, 2021.
- K. Hazelwood i in., „Applied Machine Learning at Facebook: A Datacenter Infrastructure Perspective", *IEEE International Symposium on High Performance Computer Architecture (HPCA)*, 2018.
- D. Sculley i in., „Hidden Technical Debt in Machine Learning Systems", [w:] *Advances in Neural Information Processing Systems*, 2015.
- National Institute of Standards and Technology, *AI Risk Management Framework (AI RMF)*, NIST, 2023.
- P.R. Baheti, H. Gill, „Cyber-physical Systems", [w:] *The Impact of Control Technology*, 2011.
- D. Patterson i in., *Carbon Emissions and Large Neural Network Training*, preprint arXiv, arXiv:2104.10350, 2021.
- Y. LeCun, Y. Bengio, G. Hinton, *Deep Learning*, „Nature", 2015.
- H. Mao i in., „Resource Management with Deep Reinforcement Learning", [w:] *Proceedings of the 15th ACM Workshop on Hot Topics in Networks*, 2016.

Poznaj wzorce, strategie i koncepcje niezbędne do projektowania złożonej architektury AI!

Każdego dnia pojawiają się nowe zastosowania sztucznej inteligencji. Większość z nich opiera się na złożonych systemach programistycznych, których budowa wymaga zdyscyplinowanego, inżynierskiego podejścia. Dobre praktyki projektowania oprogramowania są doskonale znane — ale czy sprawdzają się w architekturze, w której kluczową rolę odgrywają modele AI?

W tej książce znajdziesz sposób na zapanowanie nad złożonością integracji AI. Poznasz koncepcje i procesy architektoniczne kluczowe dla budowania skalowalnych, solidnych systemów AI przy jednoczesnej minimalizacji ryzyka związanego z ich rozwojem i konserwacją. Poszczególne zagadnienia zostały zilustrowane rzeczywistymi przykładami i wzbogacone o praktyczne ćwiczenia, co pozwoli Ci pogłębić zrozumienie omawianych tematów. Krok po kroku nauczysz się budować kluczowe komponenty architektoniczne wspierające systemy AI. Książkę w szczególności docenią architekci i doświadczeni programiści, którzy chcą budować systemy AI w sposób uporządkowany, przewidywalny i zgodny z zasadami inżynierii oprogramowania.

Najciekawsze zagadnienia:

- wyzwania architektoniczne w systemach AI
- narzędzia ułatwiające projektowanie i integrację rozwiązań AI
- koncepcje AI/ML, takie jak wnioskowanie i podejmowanie decyzji
- prototypowanie i iteracyjne doskonalenie systemów
- korzystanie z wzorców i heurystyk
- integracja AI z większymi systemami

Richard D. Avila jest doświadczonym architektem oprogramowania i systemów informatycznych. Jego publikacje w recenzowanych czasopismach i wydawnictwach branżowych obejmują teorię dowodzenia, architekturę zapewniania jakości, modelowanie wieloagentowe oraz uczenie maszynowe.

Dr Imran Ahmad jest naukowcem zajmującym się danymi i autorem bestsellerowej książki *50 algorytmów, które powinien znać każdy programista.* Obecnie pracuje w rządowym Centrum Zaawansowanych Rozwiązań Analitycznych (A2SC), piastuje również stanowisko profesora wizytującego na Uniwersytecie Ottawskim.

Helion
helion.pl
HELION S.A.
ul. Kościuszki 1c
44-100 Gliwice
tel.: 32 230 98 63
helion@helion.pl
KOD KORZYŚCI
Sięgnij po więcej!
ISBN 978-83-289-3855-7
9 788328 938557
Cena: 69,00 zł
packt