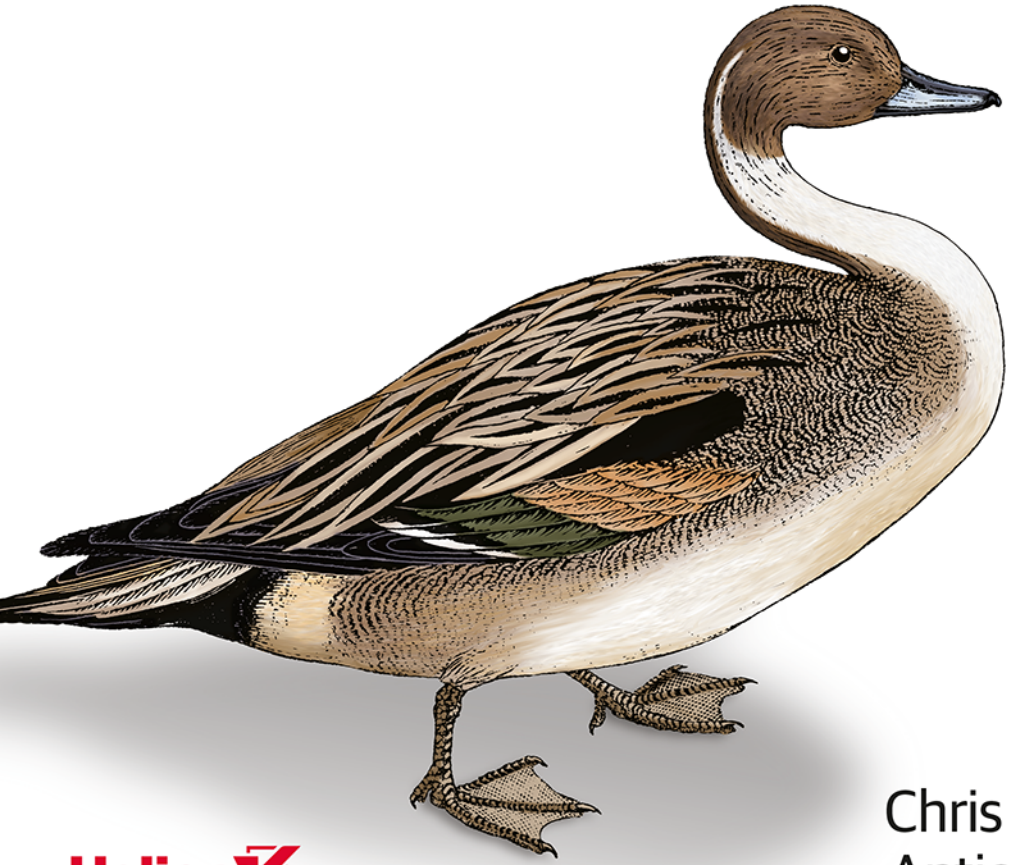
O'REILLY®

Inżynieria danych na platformie AWS

Jak tworzyć kompletne potoki uczenia maszynowego



Chris Fregly
Antje Barth

Helion

Tytuł oryginału: Data Science on AWS: Implementing End-to-End, Continuous AI and Machine Learning Pipelines

Tłumaczenie: Tomasz Walczak

ISBN: 978-83-283-9128-4

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

Helion S.A.
ul. Kościuszki 1c, 44-100 Gliwice
tel. 32 231 22 19, 32 230 98 63
e-mail: *helion@helion.pl*
WWW: *https://helion.pl* (księgarnia internetowa, katalog książek)

Printed in Poland.

Spis treści

ROZDZIAŁ 2.

Zastosowania danologii

W tym rozdziale pokazujemy, że sztuczna inteligencja i uczenie maszynowe zmieniły prawie każdą branżę — i nadal będą to robić w przyszłości. Omawiamy tu znane zastosowania tych podejść w różnych sektorach takich jak media, reklama, internet rzeczy i przemysł wytwórczy. Wraz z pojawianiem się nowych komponentów możliwe staje się rozwiązywanie coraz to nowych problemów. Programiści tworzący rozwiązania w chmurze mają dostęp do tych komponentów dzięki gotowym do użycia usługom SI takim jak Amazon Rekognition, personalizowanym usługom UM takim jak Amazon SageMaker i łatwym w dostępie komputerom kwantowym w usłudze Amazon Braket.

Sztuczna inteligencja i uczenie maszynowe stały się wszechobecne dzięki ostatnim innowacjom w przetwarzaniu chmurowym, wzrostowi mocy obliczeniowych i rejestrowaniu coraz większej ilości danych. „Demokratyzacja" SI i UM jest napędzana przez pojawienie się wielu usług SI, które są łatwe do zintegrowania z aplikacjami, wymagają bardzo niewiele konserwacji i są oferowane w modelu „płać za to, czego używasz".

Nie trzeba mieć doktoratu z danologii, żeby dodać rekomendacje produktów i zadowolić klientów, implementować wysoce skuteczne modele prognozowania i usprawnić łańcuchy dostaw lub tworzyć wirtualnych asystentów ułatwiających obsługę klienta, a wszystko to za pomocą jednego wywołania API! Tego rodzaju usługi SI zwalniają cenne zasoby ludzkie, dzięki czemu można się skupić na funkcjach specyficznych dla dziedziny i pozwalających wyróżnić produkty spośród innych.

Innowacje w każdej branży

Wiele zastosowań SI i UM można przypisać do jednej z dwóch kategorii: usprawnianie działalności biznesowej lub tworzenie nowych doświadczeń klientów. Znanymi przykładami usprawniania działalności biznesowej są prognozowanie zapotrzebowania, optymalizowanie wykorzystania zasobów i wykrywanie oszustw z wykorzystaniem SI. Przykładami tworzenia nowych doświadczeń klientów są spersonalizowane rekomendacje produktów i wzbogacone strumieniowanie wideo.

Nie ma wątpliwości, że SI i UM są źródłem innowacji w każdej branży. Oto kilka przykładów z różnych sektorów:

Media i rozrywka

Firmy zachwycają klientów wysoce angażującymi, spersonalizowanymi treściami. SI umożliwia ponadto wysoce wydajne i skuteczne pobieranie metadanych, dzięki czemu treści multimedialne są łatwiejsze do odkrywania i przeszukiwania dla klientów oraz osób z branży produkcji mediów.

Nauki przyrodnicze

Firmy stosują SI i UM do odkrywania nowych leków, zarządzania próbami klinicznymi, produkcji leków, cyfrowego rozwoju terapii i wspomagania decyzji klinicznych.

Usługi finansowe

SI i UM ułatwiają przestrzeganie regulacji, nadzór i wykrywanie oszustw. Pomagają przyspieszyć przetwarzanie dokumentów, ustalanie spersonalizowanych cen i rekomendacji produktów finansowych, a także wspomagają podejmowanie decyzji w handlu instrumentami finansowymi.

Branża motoryzacyjna

SI i UM są podstawą autonomicznej jazdy i nawigacji oraz samochodów podłączonych do sieci.

Przemysł wytwórczy

SI i UM wspomagają projektowanie techniczne i zarządzanie łańcuchami dostaw, a także optymalizację konserwacji, napraw i działań operacyjnych. Usprawniają też pracę linii montażowych i są podstawą inteligentnych produktów oraz fabryk.

Gry

Branża gier wykorzystuje SI i UM do wprowadzania inteligentnego automatycznego skalowania zasobów serwerów gry wraz ze zmianami obciążenia w ciągu dnia.

Dalej szczegółowo omawiamy najbardziej znane zastosowania SI i pokazujemy, jak można zacząć stosować podobne rozwiązania za pomocą gotowych usług SI z platformy AWS.

Spersonalizowane rekomendacje produktów

W ostatnich dziesięcioleciach klienci otrzymują coraz więcej spersonalizowanych rekomendacji produktów i treści. Rekomendacje są wszechobecne, także w serwisach Amazon.com, który sugeruje następne produkty do kupienia, i Amazon Prime Video, gdzie polecane są następne programy do obejrzenia.

Wiele systemów rekomendacji wykrywa podobieństwa na podstawie tego, jak klienci wchodzą w interakcje z pozycjami z katalogu. Wczesna wersja tego rodzaju „filtrowania kolaboratywnego" została opisana w opublikowanym przez serwis Amazon.com artykule z 2003 roku „Amazon.com Recommendations: Item-to-Item Collaborative Filtering" (*https://oreil.ly/LbrdC*).

Obecnie zaawansowane techniki uczenia głębokiego pomagają zrozumieć potrzeby klientów w odpowiednim momencie i we właściwym kontekście. Niezależnie od tego, czy robisz zakupy w serwisie Amazon.com, słuchasz muzyki w usłudze Prime Music, oglądasz programy w usłudze Prime

Video, czytasz e-booki na urządzeniu Amazon Kindle, czy słuchasz audiobooków za pomocą usługi Audible, będziesz otrzymywać nowe spersonalizowane rekomendacje.

Proste systemy rekomendacji często mają początkowo postać systemów opartych na regułach. Gdy liczba użytkowników i produktów w systemie rośnie, coraz trudniej jest definiować reguły na tyle dokładne, aby móc generować sensowne rekomendacje. Ogólne reguły przeważnie nie dają wystarczająco dobrych wyników, aby zachęcić klientów do powrotu.

Nawet systemy rekomendacji oparte na UM są narażone na problemy. Trzeba radzić sobie z nowymi użytkownikami i nowymi pozycjami w systemie, dla których nie ma danych pozwalających generować rekomendacje. Jest to klasyczny problem „zimnego rozruchu", który powinien być uwzględniony przez każdego, kto w dzisiejszych czasach tworzy systemy rekomendacji. Zimny rozruch ma miejsce, gdy brakuje informacji o wcześniejszych zdarzeniach, na podstawie których można byłoby zbudować model rekomendacji dla danego użytkownika lub produktu.

W rekomendacjach należy też unikać „pułapki popularności", która prowadzi do rekomendowania tylko popularnych produktów, co grozi pominięciem świetnych rekomendacji mniej znanych pozycji. Ten problem można rozwiązać za pomocą systemów rekomendacji, które eksplorują nowe pozycje za pomocą algorytmów takich jak wieloręki bandyta, co omawiamy w rozdziale 9.

Warto też uwzględniać zachodzące w czasie rzeczywistym zmiany w intencjach użytkowników aplikacji. To wymaga działającego w czasie rzeczywistym dynamicznego systemu rekomendacji zamiast tradycyjnych, wstępnie generowanych statycznych rekomendacji przesyłanych z bazy danych.

Dzięki dynamicznemu systemowi klienci docenią adekwatne i szybko przesyłane treści, a firma odniesie wymienione poniżej korzyści z bardziej spersonalizowanych doświadczeń klientów:

Większe zaangażowanie użytkowników

Dzięki rekomendowaniu użytkownikom adekwatnych treści firma zwiększa ich przywiązanie do witryny, zachęca do częstych powrotów i daje powód do dłuższych odwiedzin.

Wyższy poziom konwersji

Użytkownicy z większym prawdopodobieństwem kupują bardziej adekwatne produkty.

Wyższy współczynnik kliknięć

Dzięki spersonalizowanym propozycjom produktów dostosowanym do konkretnego użytkownika współczynnik kliknięć zapewne wzrośnie.

Wzrost przychodów

Gdy klienci otrzymują odpowiednie rekomendacje we właściwym czasie, firmy odnotowują wzrost przychodów.

Zmniejszenie poziomu utraty klientów

Można zmniejszyć poziom utraty klientów i ograniczyć rezygnacje z ciekawych kampanii e-mailowych.

W ciągu ostatnich 20 lat Amazon stale prowadził skoncentrowane na personalizacji badania nad uczeniem maszynowym. Artykuł „Two Decades of Recommender Systems at Amazon.com” (*https://oreil.ly/iXEXk*) Smitha i Lindena (2017) zawiera świetne podsumowanie tej drogi.

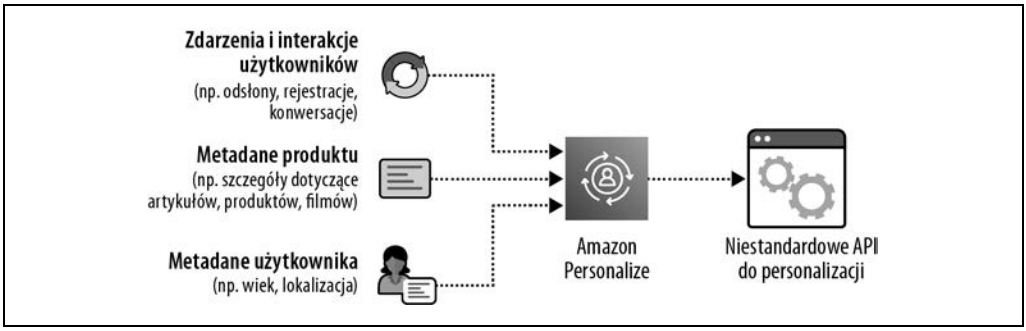
Więcej informacji o badaniach i publikacjach naukowych Amazona znajdziesz na stronie *https://www.amazon.science*.

Rekomendowanie produktów za pomocą usługi Amazon Personalize

Podobnie jak w wielu innych obszarach uczenia maszynowego, nie istnieje jeden algorytm, który rozwiązuje wszystkie problemy z personalizacją. Czy nie byłoby wspaniale, gdyby ktoś potrafił ująć bogate doświadczenia firmy Amazon.com z obszaru tworzenia spersonalizowanych rekomendacji produktów i treści oraz dodać analogiczne możliwości do naszych aplikacji? Usługa Amazon Personalize zapewnia właśnie takie funkcje.

Ta usługa jest odzwierciedleniem dziesięcioleci doświadczenia firmy Amazon.com w tworzeniu, skalowaniu i obsłudze technologii personalizacji. Amazon Personalize umożliwia programistom łatwe generowanie zindywidualizowanych rekomendacji produktów, a także tworzenie targetowanych promocji marketingowych. Ta usługa SI umożliwia programistom budowanie niestandardowych modeli personalizacji bez konieczności skomplikowanego zarządzania własną infrastrukturą do uczenia maszynowego.

Aby rozpocząć generowanie rekomendacji, wystarczy przekazać do usługi Amazon Personalize ciągły strumień aktywności z aplikacji (kliknięć, odsłon stron, rejestracji, zakupów itd.) wraz z zestawem produktów uwzględnianych w rekomendacjach (zobacz rysunek 2.1).



Rysunek 2.1. Aby zacząć generować rekomendacje, udostępnij zbiór danych o aktywnościach i zestaw produktów do usługi Amazon Personalize

Dane o aktywności obejmują informacje o zdarzeniach dotyczących interakcji użytkownika z systemem. Niektóre z tych aktywności to kliknięcia, dodawanie produktów do koszyka, kupowanie produktów lub oglądanie wideo. Tego rodzaju aktywności są wartościowymi sygnałami pomocnymi przy budowaniu skutecznych modeli rekomendacji.

Można też przekazywać dodatkowe metadane na temat użytkowników i produktów z interakcji, na przykład kategorię produktu, cenę produktu, wiek użytkownika, lokalizację użytkownika itd. Choć takie dodatkowe metadane są opcjonalne, pomagają w scenariuszu „zimnego rozruchu", kiedy dostępna jest niewielka lub zerowa ilość informacji o historycznych aktywnościach, które można byłoby wykorzystać do zbudowania modelu rekomendacji.

Twórcy usługi Amazon Personalize poinformowali niedawno o nowym algorytmie „zimnego rozruchu", który łączy sieci neuronowe i uczenie przez wzmacnianie, aby generować bardziej adekwatne rekomendacje, gdy dostępnych jest niewiele danych o użytkowniku.

Dzięki informacjom o aktywnościach i metadanym Amazon Personalize uczy, dostraja i instaluje niestandardowy model rekomendacji uwzględniający użytkowników i produkty. Usługa ta wykonuje wszystkie kroki procesu uczenia maszynowego, w tym inżynierię cech, wybór algorytmów, dostrajanie modelu i instalowanie modelu. Po tym, jak usługa wybierze, wyuczy i zainstaluje najlepszy model dla używanego zbioru danych, wystarczy użyć API `get_recommendations()`, aby zacząć generować w czasie rzeczywistym rekomendacje dla użytkowników:

```
get_recommendations_response = personalize_runtime.get_recommendations(
        campaignArn = campaign_arn,
        userId = user_id
)

item_list = get_recommendations_response['itemList']
recommendation_list = []
for item in item_list:
    item_id = get_movie_by_id(item['itemId'])
    recommendation_list.append(item_id)
```

Usługa Amazon Personalize wyuczona na podstawie znanego zbioru danych MovieLens z milionami ocen filmów generuje następujące rekomendacje dla przykładowego użytkownika:

```
Shrek
Amelie
Lord of the Rings: The Two Towers
Toy Story 2
Good Will Hunting
Eternal Sunshine of the Spotless Mind
Spirited Away
Lord of the Rings: The Return of the King
Schindler's List
Leon: The Professional
```

Generowanie rekomendacji za pomocą usługi Amazon SageMaker i biblioteki TensorFlow

Wielozadaniowe systemy rekomendacji tworzą model, który optymalizuje wyniki na podstawie przynajmniej dwóch celów jednocześnie. Taki model uczy się z wykorzystaniem transferu wiedzy, współdzieląc zmienne między zadaniami na etapie uczenia modelu.

W poniższym przykładzie opartym na bibliotece TensorFlow używamy biblioteki TFRS (ang. *Tensor ↪Flow Recommenders*; *https://oreil.ly/XdDIl*), a celem jest znalezienie modelu, który uczy system rekomendacji prognozować oceny (zadanie `Ranking`), a także liczbę wyświetleń wideo (zadanie `Retrieval`):

```
user_model = tf.keras.Sequential([
  tf.keras.layers.experimental.preprocessing.StringLookup(
      vocabulary=unique_user_ids),
  # Dodajemy 2, aby uwzględnić nieznane i ukryte tokeny.
  tf.keras.layers.Embedding(len(unique_user_ids) + 2, embedding_dimension)
])

movie_model = tf.keras.Sequential([
  tf.keras.layers.experimental.preprocessing.StringLookup(
      vocabulary=unique_movie_titles),
  tf.keras.layers.Embedding(len(unique_movie_titles) + 2, embedding_dimension)
])

rating_task = tfrs.tasks.Ranking(
    loss=tf.keras.losses.MeanSquaredError(),
    metrics=[tf.keras.metrics.RootMeanSquaredError()],
)

retrieval_task = tfrs.tasks.Retrieval(
        metrics=tfrs.metrics.FactorizedTopK(
            candidates=movies.batch(128).map(self.movie_model)
        )
)
```

Generowanie rekomendacji za pomocą usługi Amazon SageMaker i platformy Apache Spark

Usługa Amazon SageMaker obsługuje bezserwerową platformę Apache Spark (można na niej używać Pythona i Scali) za pomocą zadań SageMaker Processing. W książce używamy zadań SageMaker Processing do sprawdzania jakości danych i przekształcania cech. Jednak w tym punkcie generujemy rekomendacje za pomocą zadań SageMaker Processing i algorytmu filtrowania kolaboratywnego *Alternating Least Squares* (ALS) z pakietu Spark ML. Ten algorytm można zastosować, jeśli już używany jest potok danych oparty na platformie Spark i ma on posłużyć do generowania rekomendacji.

Oto plik *train_spark.py* z kodem generującym rekomendacje z użyciem pakietu Spark ML i algorytmu ALS:

```
import pyspark
from pyspark.sql import SparkSession
from pyspark.sql.functions import *
```

```
from pyspark.ml.evaluation import RegressionEvaluator
from pyspark.ml.recommendation import ALS
from pyspark.sql import Row

def main():
    ...
    lines = spark.read.text(s3_input_data).rdd
    parts = lines.map(lambda row: row.value.split("::"))
    ratingsRDD = parts.map(lambda p: Row(userId=int(p[0]),
                                         movieId=int(p[1]),
                                         rating=float(p[2]),
                                         timestamp=int(p[3])))
    ratings = spark.createDataFrame(ratingsRDD)
    (training, test) = ratings.randomSplit([0.8, 0.2])

    # Tworzenie modelu rekomendacji przez zastosowanie algorytmu ALS do danych treningowych.
    als = ALS(maxIter=5,
              regParam=0.01,
              userCol="userId",
              itemCol="itemId",
              ratingCol="rating",
              coldStartStrategy="drop")
    model = als.fit(training)

    # Ocena modelu na podstawie błędu średniokwadratowego obliczonego dla danych testowych.
    predictions = model.transform(test)
    evaluator = RegressionEvaluator(metricName="rmse",
                                    labelCol="rating",
                                    predictionCol="prediction")
    rmse = evaluator.evaluate(predictions)

    # Generowanie dziesięciu pierwszych rekomendacji dla każdego użytkownika.
    userRecs = model.recommendForAllUsers(10)
    userRecs.show()

    # Wyświetlanie pierwszych dziesięciu rekomendacji dla każdego użytkownika.
    userRecs.repartition(1).write.mode("overwrite")\
    .option("header", True).option("delimiter", "\t")\
    .csv(f"{s3_output_data}/recommendations")
```

Teraz uruchom ten skrypt PySpark w bezserwerowym środowisku platformy Apache Spark, używając zadania SageMaker Processing:

```
from sagemaker.spark.processing import PySparkProcessor
from sagemaker.processing import ProcessingOutput

processor = PySparkProcessor(base_job_name='spark-als',
                             role=role,
                             instance_count=1,
                             instance_type='ml.r5.2xlarge',
                             max_runtime_in_seconds=1200)

processor.run(submit_app='train_spark_als.py',
              arguments=['s3_input_data', s3_input_data,
                         's3_output_data', s3_output_data,
              ],
              logs=True,
              wait=False
)
```

W danych wyjściowych podawany jest identyfikator użytkownika i lista rekomendacji (identyfikator produktu i ocena) posortowanych według ocen od najbardziej do najmniej rekomendowanego filmu:

```
|userId|     recommendations|
+------+--------------------+
|    12|[[46, 6.146928], ...|
|     1|[[46, 4.963598], ...|
|     6|[[25, 4.5243497],...|
+------+--------------------+
```

Wykrywanie niestosownych materiałów wideo za pomocą usługi Amazon Rekognition

Wykrywanie obrazów jest przydatne w wielu zastosowaniach, w tym do moderowania treści generowanych przez użytkowników, cyfrowego weryfikowania tożsamości w ramach bezpiecznego logowania i wykrywania zagrożeń przez samochody autonomiczne.

Amazon Rekognition to usługa SI wysokiego poziomu, która identyfikuje obiekty (w tym ludzi), tekst i aktywności na zdjęciach i w materiałach wideo. Usługa ta korzysta ze zautomatyzowanego UM do uczenia niestandardowych modeli rozpoznawania obiektów specyficznych dla danej sytuacji i działalności biznesowej.

Tu zastosujemy usługę Amazon Rekognition do wykrywania scen przemocy w filmach przesłanych przez użytkowników aplikacji. Załóżmy, że za pomocą API Content Moderation z tej usługi chcemy odrzucać materiały wideo, w których pojawia się broń:

```
startModerationLabelDetection = rekognition.start_content_moderation(
    Video={
        'S3Object': {
            'Bucket': bucket,
            'Name': videoName,
        }
    },
)

moderationJobId = startModerationLabelDetection['JobId']

getContentModeration = rekognition.get_content_moderation(
    JobId=moderationJobId,
    SortBy='TIMESTAMP'
)

while(getContentModeration['JobStatus'] == 'IN_PROGRESS'):
    time.sleep(5)
    print('.', end='')

    getContentModeration = rekognition.get_content_moderation(
     JobId=moderationJobId,
     SortBy='TIMESTAMP')

display(getContentModeration['JobStatus'])
```

Oto dane wyjściowe z przypisanymi etykietami. Zwróć uwagę na pole `Timestamp`, które określa czas od początku filmu, oraz pole `Confidence`, określające poziom pewności co do etykiety przypisanej przez usługę Amazon Rekognition :

```
{'JobStatus': 'SUCCEEDED',
 'VideoMetadata': {'Codec': 'h264',
  'DurationMillis': 6033,
  'Format': 'QuickTime / MOV',
  'FrameRate': 30.0,
  'FrameHeight': 1080,
  'FrameWidth': 1920},
 'ModerationLabels': [{'Timestamp': 1999,
  'ModerationLabel': {'Confidence': 75.15272521972656,
   'Name': 'Violence',
   'ParentName': ''}},
 {'Timestamp': 1999,
  'ModerationLabel': {'Confidence': 75.15272521972656,
   'Name': 'Weapons',
   'ParentName': 'Violence'}},
 {'Timestamp': 2500,
  'ModerationLabel': {'Confidence': 87.55487060546875,
   'Name': 'Violence',
   'ParentName': ''}}]

Moderation labels in video
==========================================
At 1999 ms: Violence (Confidence: 75.15)
At 1999 ms: Weapons (Confidence: 75.15)
At 2500 ms: Violence (Confidence: 87.55)
```

Można dodatkowo zwiększyć pewność co do etykiet przypisywanych przez usługę Amazon Rekognition, przeprowadzając uczenie z użyciem własnego zbioru danych. Umożliwia to funkcja *custom labels*, dostępna w wielu usługach SI Amazona.

Prognozowanie zapotrzebowania

Prognozowanie zapotrzebowania jest stosowane w wielu dziedzinach do szacowania zapotrzebowania w obszarach takich jak: zużycie energii elektrycznej, produkty z łańcucha dostaw, personel w centrum obsługi telefonicznej, planowanie przepływów pieniężnych, wykorzystanie łóżek w szpitalach itd. Prognozowanie to problem z obszaru szeregów czasowych. Istnieje wiele znanych algorytmów do jego rozwiązywania, między innymi: Auto-Regressive Integrated Moving Average, Error Trend Seasonality, Non-Parametric Time Series, Prophet i DeepAR++.

Firmy używają różnych narzędzi, od prostych arkuszy kalkulacyjnych po złożone oprogramowanie do przetwarzania szeregów czasowych, aby prognozować przyszłe wartości takie jak popyt na produkt, zapotrzebowanie na zasoby lub wyniki finansowe. W tym podejściu modele prognostyczne są zwykle budowane na podstawie historycznych szeregów czasowych z założeniem, że przyszłe zapotrzebowanie jest determinowane przez dawną aktywność. Techniki oparte wyłącznie na szeregach czasowych nie generują trafnych prognoz, gdy trendy i wzorce są mało regularne. Ponadto w takich podejściach nie są uwzględniane wpływające na prognozy ważne metadane takie jak cena produktu, kategoria produktu lub lokalizacja sklepu.

Prognozowanie zbyt wysokiego zapotrzebowania zmniejsza efektywność i zwiększa koszty, ponieważ pozyskiwanych jest zbyt dużo zasobów, które nie są w pełni wykorzystywane. Prognozowanie zbyt niskiego zapotrzebowania może prowadzić do spadku satysfakcji klientów, niższych przychodów, ponieważ systemowi brakuje niezbędnych zasobów, i ponoszenia wyższych kosztów, na przykład płacenia pracownikom za nadgodziny.

Skuteczny system prognozowania zapotrzebowania powinien mieć następujące cechy:

Analizowanie złożonych zależności, a nie tylko danych z szeregów czasowych

Należy łączyć dane z szeregów czasowych z innymi metadanymi takimi jak cechy produktów i lokalizacje sklepów.

Skrócenie czasu generowania predykcji z miesięcy do godzin

Po automatycznym wczytaniu i zbadaniu zbioru danych oraz po zidentyfikowaniu w nim kluczowych atrybutów system powinien szybko uczyć, optymalizować i instalować niestandardowy model dopasowany do tego zbioru danych.

Generowanie prognoz dla wielu różnych zastosowań

Należy generować prognozy dla wszystkich możliwych zastosowań, w tym na potrzeby łańcucha dostaw, logistyki i finansów, używając rozbudowanej biblioteki algorytmów do automatycznego dostosowywania wyników do konkretnego zastosowania.

Zapewnianie bezpieczeństwa danych

Każdy punkt danych musi być chroniony za pomocą szyfrowania w spoczynku i w tranzycie, aby zapewnić bezpieczeństwo i poufność wrażliwych danych.

Automatyczne ponowne uczenie i instalowanie modeli, gdy jest to konieczne

Gdy pojawiają się nowe dane (albo gdy wyniki modelu spadają poniżej ustalonego poziomu progowego), system powinien ponownie uczyć i instalować model, aby poprawić predykcje.

Prognozowanie zużycia energii elektrycznej za pomocą usługi Amazon Forecast

Amazon Forecast to w pełni zarządzana usługa oparta na technologii, służąca do prognozowania zapotrzebowania w serwisie Amazon.com, aby umożliwić na przykład wydajne zarządzanie stanami magazynowymi, błyskawiczną realizację zamówień i dostawy tego samego dnia. Usługa ta wykorzystuje uczenie maszynowe w celu automatycznego uczenia, dostrajania i optymalizowania wysoce wyspecjalizowanych modeli prognozowania zapotrzebowania na podstawie wskazanego zbioru danych. Wystarczy zarejestrować historyczne zbiory danych i powiązane z nimi metadane w usłudze Forecast, aby rozpocząć generowanie predykcji zapotrzebowania. Prognozy zapotrzebowania można eksportować w formacie CSV, wyświetlać w panelu konsoli platformy AWS lub integrować z własnymi aplikacjami za pomocą API usług Forecast.

Zobacz teraz, jak wyuczyć model prognozowania zapotrzebowania pojedynczych gospodarstw domowych na energię elektryczną w następnych 24 godzinach za pomocą algorytmu DeepAR++ z usługi Forecast i publicznego zbioru danych z repozytorium UCI Machine Learning *https://oreil.ly/DYLJ7*).

Oto fragment tego zbioru danych z informacjami o zużyciu energii przez poszczególnych klientów:

Znacznik czasu	Wartość	Odbiorca
2014-01-01 01:00:00	38.34991708126038	client_12
2014-01-01 02:00:00	33.5820895522388	client_12
2014-01-01 03:00:00	34.41127694859037	client_12

Oto schemat zdefiniowany w usłudze Forecast do reprezentowania tego publicznego zbioru danych:

```
forecast_schema ={
   "Attributes":[
      {
         "AttributeName":"timestamp",
         "AttributeType":"timestamp"
      },
      {
         "AttributeName":"target_value",
         "AttributeType":"float
      },
      {
         "AttributeName":"item_id",
         "AttributeType":"string"
      }
   ]
}
```

Zarejestruj ten zbiór danych w usłudze Forecast:

```
response=forecast.create_dataset(
                    Domain="CUSTOM",
                    DatasetType='TARGET_TIME_SERIES',
                    DatasetName=forecast_dataset_name,
                    DataFrequency=DATASET_FREQUENCY,
                    Schema = forecast_schema
)
```

Teraz zobacz, jak wyuczyć model prognozowania zapotrzebowania w usłudze Forecast:

```
forecast_horizon = 24 # Liczba godzin.

algorithm_arn = 'arn:aws:forecast:::algorithm/Deep_AR_Plus'

create_predictor_response = \
    forecast.create_predictor(PredictorName=predictor_name,
                              AlgorithmArn=algorithm_arn,
                              ForecastHorizon=forecast_horizon,
                              PerformAutoML= False,
                              PerformHPO=False,
                              EvaluationParameters= {
                                  "NumberOfBacktestWindows": 1,
                                  "BackTestWindowOffset": 24
                              },
                              InputDataConfig= {
                                  "DatasetGroupArn": forecast_dataset_group_arn
                              },
                              FeaturizationConfig= {
                                  "ForecastFrequency": "H",
```

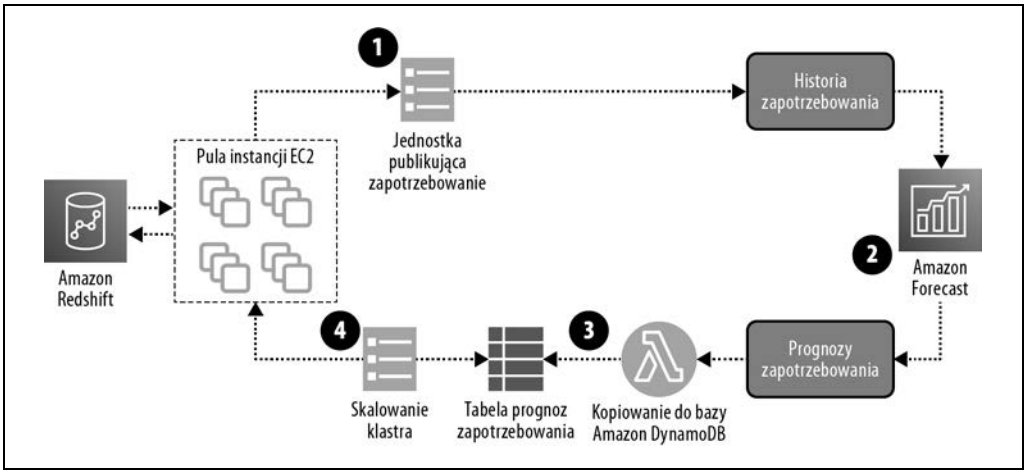
```
            "Featurizations": [{
                "AttributeName": "target_value",
                "FeaturizationPipeline":
                [{
                    "FeaturizationMethodName": "filling",
                    "FeaturizationMethodParameters": {
                        "frontfill": "none",
                        "middlefill": "zero",
                        "backfill": "zero"
                    }
                }]
            }]
        })
```

Teraz wygeneruje predykcje:

```
forecastResponse = forecastquery.query_forecast(
    ForecastArn=forecast_arn,
    Filters={"item_id":"client_12"}
)
```

Prognozowanie zapotrzebowania na instancje Amazon EC2 za pomocą usługi Amazon Forecast

AWS używa usługi Forecast do prognozowania zapotrzebowania na instancje Amazon EC2 w klastrach z hurtownią Amazon Redshift. Gdy nowe dane są pobierane do usługi Forecast, warstwa sterowania z hurtowni Amazon Redshift żąda od usługi Forecast dostosowania wielkości puli wstępnie zainicjowanych instancji Amazon EC2 przeznaczonych dla tej bazy (zobacz rysunek 2.2).



Rysunek 2.2. Warstwa sterowania z hurtowni Amazon Redshift dostosowuje pulę wstępnie zainicjowanych instancji Amazon EC2, używając usługi Forecast

Oto omówienie etapów z rysunku 2.2:

1. Zmiany zapotrzebowania na pulę wstępnie zainicjowanych instancji Amazon EC2 są publikowane w S3.

2. Usługa Forecast pobiera dane o zapotrzebowaniu z S3, a następnie tworzy nowe prognozy.
3. Funkcja Lambda kopiuje nowe prognozy do bazy Amazon DynamoDB.
4. Mechanizm skalowania klastra instancji Amazon EC2 wczytuje prognozy z bazy DynamoDB i dostosowuje wielkość puli wstępnie zainicjowanych instancji na podstawie prognozowanego zapotrzebowania.

Identyfikowanie fałszywych kont za pomocą usługi Amazon Fraud Detector

Każdego roku globalne straty z powodu oszustw internetowych wynoszą dziesiątki miliardów dolarów. Firmy internetowe są szczególnie narażone na ataki ze strony napastników, którzy próbują oszukać system, tworząc fałszywe konta użytkownika i kupując produkty za pomocą skradzionych kart kredytowych. Typowe systemy wykrywania oszustw, które identyfikują napastników, często opierają się na regułach biznesowych wolno dostosowujących się do najnowszych technik stosowanych przez przestępców.

Skuteczne systemy wykrywania oszustw i zapobiegania wyciekaniu danych powinny mieć następujące cechy:

Powstrzymywanie napastników zanim wyrządzą firmie szkody

Należy oznaczać podejrzaną aktywność, zanim napastnik zdąży wyrządzić poważne szkody.

Wysokiej jakości modele wykrywania oszustw niewymagające dużej ilości danych

Wstępnie wyuczone algorytmy potrafią analizować nawet bardzo niewielkie ilości danych historycznych i na tej podstawie budować wysokiej jakości modele wykrywania oszustw.

Umożliwianie zespołom ds. oszustw szybszą pracę i zapewnianie im większej kontroli

System powinien automatycznie wykonywać złożone zadania takie jak budowanie, uczenie, dostrajanie, instalowanie i aktualizowanie modeli wykrywania oszustw, gdy pojawiają się nowe dane.

Amazon Fraud Detector to w pełni zarządzana usługa, która identyfikuje potencjalne oszustwa związane na przykład z płatnościami w internecie i fałszywymi kontami. Usługa ta jest oparta na uczeniu maszynowym i 20 latach doświadczenia w wykrywaniu oszustw przez firmy AWS i Amazon.com.

Usługa Amazon Fraud Detector umożliwia utworzenie modelu wykrywania oszustw za pomocą kilku kliknięć, stosunkowo niewielkiej ilości danych historycznych i minimalnej ilości kodu. Wystarczy przesłać dane historyczne o zdarzeniach internetowych (na przykład o transakcjach internetowych i rejestracjach kont), a Amazon Fraud Detector zajmie się resztą, w tym uczeniem, dostrajaniem i instalowaniem niestandardowego modelu wykrywania oszustw.

Oto kod używany do uczenia usługi Amazon Fraud Detector na podstawie zbioru danych o transakcjach:

```
response = client.create_model_version(
    modelId    = MODEL_NAME,
```

```
        modelType  = 'ONLINE_FRAUD_INSIGHTS',
        trainingDataSource = 'EXTERNAL_EVENTS',
        trainingDataSchema = trainingDataSchema,
        externalEventsDetail = {
            'dataLocation'     : S3_FILE_LOC,
            'dataAccessRoleArn': ARN_ROLE
        }
    )
```

A oto kod do oceny, czy dana transakcja jest nielegalna:

```
    pred = client.get_event_prediction(
        detectorId = DETECTOR_NAME,
        detectorVersionId = DETECTOR_VER,
        eventId = str(eventId),
        eventTypeName = EVENT_TYPE,
        eventTimestamp = timestampStr,
        entities = [{'entityType': ENTITY_TYPE,
        'entityId':str(eventId.int)}],
        eventVariables = record)

    record["score"] = pred['modelScores'][0]['scores']\
                        ["{0}_insightscore".format(MODEL_NAME)]
```

Oto dane wyjściowe z predykcjami usługi Amazon Fraud Detector, obejmujące datę, ocenę transakcji i poziom pewności:

ip_address	email_address	state	postal	name	phone_number	score	outcomes
84.138.6.238	synth1@yahoo.com	LA	32733	Brandon Moran	(555)784 - 5238	5.0	[approve]
194.147.250.63	synth2@yahoo.com	MN	34319	Dominic Murray	(555)114 - 6133	4.0	[approve]
192.54.60.50	synth3@gmail.com	WA	32436	Anthony Abbott	(555)780 - 7652	5.0	[approve]
169.120.193.154	synth4@gmail.com	AL	34399.0	Kimberly Webb	(555)588 - 4426	938.0	[review]
192.175.55.43	synth5@hotmail.com	IL	33690.0	Renee James	(555)785 - 8274	16.0	[approve]

Używanie usługi Amazon Macie do wykrywania wycieków wrażliwych danych

Dobrze skonfigurowana aplikacja generuje wiele dzienników i wskaźników, aby zwiększyć wgląd w jej działanie i zapewnić wysoką bezawaryjność, co pomaga uniknąć niezadowolenia klientów. Jednak czasem dzienniki zawierają wrażliwe informacje o kontach, na przykład kody pocztowe lub numery kart kredytowych. Potrzebny jest więc system, który monitoruje dane pod kątem wrażliwych informacji, wykrywa dostęp do takich danych i przesyła powiadomienia po wykryciu nieuprawnionego dostępu lub wycieku danych.

Skuteczny system wykrywania i monitorowania dostępu do wrażliwych informacji powinien mieć następujące cechy:

Ciągła ocena mechanizmów kontroli dostępu i wrażliwości danych

Rachunek możliwych zysków i strat przeprowadzony przez napastnika podpowiada mu, że komora S3 z wrażliwymi danymi o klientach i słabo skonfigurowanymi rolami IAM jest atrakcyjnym celem. Należy więc uprzedzić działania napastników, stale monitorując całe środowisko S3 i generując informacje pozwalające na szybkie reagowanie, gdy jest to konieczne.

Obsługa wielu źródeł danych

Możliwa powinna być ocena wrażliwości danych i mechanizmów kontroli dostępu w wielu różnych źródłach danych takich jak S3, Amazon Relational Database Service (Amazon RDS), Amazon Aurora, poczta elektroniczna, serwery wymiany plików, narzędzia do współpracy itd.

Utrzymywanie zgodności z regulacjami

Obok monitorowania i ochrony danych wrażliwych zespoły ds. zgodności z regulacjami powinny udowodnić, że zapewniają bezpieczeństwo i prywatność danych, by spełnić wymogi regulacyjne.

Identyfikowanie danych wrażliwych w trakcie przenoszenia danych

W czasie przenoszenia dużych ilości danych do platformy AWS należy sprawdzić, czy występują w nich dane wrażliwe. Jeśli tak jest, w procesie przenoszenia danych zwykle trzeba zmodyfikować mechanizmy kontroli dostępu, ustawienia szyfrowania i tagi zasobów.

Amazon Macie to w pełni zarządzana usługa z obszaru zabezpieczeń, która wykorzystuje uczenie maszynowe do identyfikowania danych wrażliwych takich jak dane osobowe w źródłach danych z platformy AWS (na przykład w S3). Usługa Macie zapewnia wgląd w to, gdzie dane są składowane i kto ich używa. Dzięki monitorowaniu dostępu do wrażliwych danych usługa Macie może przesyłać alerty po wykryciu wycieku danych (lub ryzyka jego wystąpienia).

Macie stale identyfikuje dane wrażliwe oraz ocenia zabezpieczenia i mechanizm kontroli dostępu do takich danych. Usługa pomaga zachować prywatność i bezpieczeństwo wszystkich danych oraz zapewnia rozbudowane funkcje planowania analiz wrażliwości danych i mechanizmów kontroli dostępu, aby zachować prywatność danych i zgodność z regulacjami.

Można zaplanować codzienne, cotygodniowe lub comiesięczne zadania, które generują ocenę danych, znaczniki czasu i historyczne zapisy wszystkich komór oraz obiektów przeskanowanych pod kątem danych wrażliwych. Te odkrycia są podsumowywane w standardowym raporcie dostosowanym do audytów z zakresu prywatności i ochrony danych, dzięki czemu długoterminowe przechowywanie danych jest łatwiejsze. Jeśli chodzi o migrację danych, usługa Macie automatyzuje konfigurację polityki ochrony danych i dostępu opartego na rolach, gdy dane są przenoszone na platformę AWS.

Urządzenia konwersacyjne i asystenci głosowi

Zarówno Alexa, jak i inni słynni asystenci głosowi do użytku domowego korzystają z najnowszych technologii uczenia głębokiego z dziedzin automatycznego rozpoznawania mowy i rozumienia języka naturalnego, aby rozpoznawać znaczenie mówionego tekstu.

Rozpoznawanie mowy za pomocą usługi Amazon Lex

Gdy używasz usługi Amazon Lex do budowania głosowych i tekstowych interfejsów konwersacyjnych, masz dostęp do tych samych technologii uczenia głębokiego, na których oparty jest asystent Amazon Alexa. Amazon Lex to w pełni zarządzana usługa, która używa mechanizmów automatycznego rozpoznawania mowy do przekształcania mowy na tekst. Amazon Lex stosuje też mechanizmy rozumienia języka naturalnego, aby odkryć znaczenie tekstu. Można przygotować właściwe reakcje na wiele pytań głosowych i poleceń tekstowych takich jak „Gdzie w tym biurze znajduje się dział pomocy technicznej?" lub „Zarezerwuj tę salę na następne 30 minut".

Konwersja tekstu na mowę za pomocą usługi Amazon Polly

Amazon Polly jest zautomatyzowaną usługą przekształcania tekstu na mowę z dziesiątkami ludzkich głosów reprezentujących różne języki, dialekty i płcie. Za pomocą tej usługi możesz tworzyć aplikacje głosowe, które przekształcają tekst na mowę zbliżoną do ludzkiej, na przykład w celu pomocy niepełnosprawnym.

Konwersja mowy na tekst za pomocą usługi Amazon Transcribe

Amazon Transcribe to usługa automatycznego rozpoznawania mowy, która ułatwia programistom dodawanie mechanizmów przekształcania mowy na tekst w aplikacjach działających w czasie rzeczywistym i wsadowo. Amazon Transcribe przekształca mowę na tekst, przetwarzając dźwięk w porcjach lub w czasie rzeczywistym. Znane zastosowania tej usługi to tworzenie podpisów pod zdjęciami i napisów do materiałów wideo.

Analiza tekstu i NLP

NLP (ang. *natural language processing*) to dziedzina sztucznej inteligencji skupiona na rozwoju zdolności maszyn do czytania i rozumienia języków ludzkich oraz wydobywania znaczenia z tekstu. Badania w tym obszarze są prowadzone już od bardzo dawna. Pierwsze publikacje na ten temat pojawiły się już na początku XX wieku.

Wróćmy szybko do 2021 roku, kiedy to nadal prowadzone są przełomowe badania nad NLP, a niemal w każdym miesiącu pojawiają się nowe modele języka. W dalszych rozdziałach omawiamy ewolucję algorytmów NLP, opisujemy nowatorską architekturę sieci neuronowych Transformer i zagłębiamy się w rodzinę algorytmów BERT z obszaru NLP.

Skuteczny system analizy tekstu i wyszukiwania kognitywnego powinien mieć następujące cechy:

Szybki czas działania

Pożądane jest, aby nowe dokumenty można było szybko przeszukiwać i by nie pojawiały się błędy wymagające poprawek ze strony człowieka.

Wydajne procesy przetwarzania

Procesy przetwarzania dokumentów powinny być zautomatyzowane, aby zwiększyć szybkość i jakość, a jednocześnie zmniejszyć koszty oraz ilość ludzkiej pracy i niestandardowego kodu.

Tłumaczenie tekstów za pomocą usługi Amazon Translate

W dzisiejszej globalnej ekonomii trzeba dostosować się do użytkowników z innych państw, tłumacząc materiały na wiele specyficznych dla regionów wersji językowych. Częste scenariusze to tłumaczenie na żądanie treści wygenerowanych przez użytkowników, tłumaczenie w czasie rzeczywistym tekstu w komunikatorach i analiza sentymentu na podstawie wielojęzycznych treści w mediach społecznościowych.

Amazon Translate to usługa tłumaczenia wykorzystująca sieci neuronowe, która generuje dokładniejsze i płynniejsze tłumaczenia niż tradycyjne modele tłumaczeń oparte na statystyce i regułach.

Klasyfikowanie wiadomości do działu obsługi klienta za pomocą usługi Amazon Comprehend

„Obsesja na punkcie klientów" to jedna z głównych zasad Amazona. Koncentracja na użytkownikach jest ważna dla każdej firmy i branży. W wielu sytuacjach doświadczenia konsumentów są w dużym stopniu zależne od jakości obsługi klienta. W tym punkcie pokażemy, jak użyć usługi Amazon Comprehend do klasyfikowania sentymentu na podstawie zestawu wiadomości do działu obsługi klienta.

Klasyfikowanie tekstu to często wykonywane zadanie w dziedzinie NLP. Wcześniej wspomnieliśmy, że można użyć Amazon Comprehend jako w pełni zarządzanej usługi z obszaru NLP do klasyfikowania tekstu. Nie wymaga to dużego doświadczenia z zakresu uczenia maszynowego.

Na bardziej ogólnym poziomie Amazon potrafi rozpoznawać ważne encje, kluczowe wyrażenia, sentyment, język i temat na podstawie dokumentów tekstowych. Do ważnych encji należą nazwy, miejsca, rzeczy i daty. Kluczowe wyrażenia to na przykład „good morning", „thank you" i „not happy". Sentyment może być pozytywny („positive"), neutralny („neutral") i negatywny („negative"). Usługa obecnie obsługuje wiele języków, a nowe są często dodawane.

Usługa Amazon Comprehend obejmuje też zestaw API dla służby zdrowia, *Amazon Comprehend Medical*. Narzędzia z tej kategorii zostały wstępnie wyuczone na podstawie rozbudowanych zbiorów danych medycznych i potrafią rozpoznawać choroby, leki, badania, zabiegi, procedury, aspekty anatomiczne i chronione informacje o zdrowiu.

Zobacz teraz, jak użyć gotowego API Sentiment Analysis z usługi Amazon Comprehend, aby za pomocą kilku wierszy kodu poklasyfikować przykładowe oceny produktów.

Najpierw użyj API `create_document_classifier()` z omawianej usługi, aby utworzyć klasyfikator:

```
training_job = comprehend.create_document_classifier(
    DocumentClassifierName=comprehend_training_job_name,
    DataAccessRoleArn=iam_role_comprehend_arn,
    InputDataConfig={
        'S3Uri': comprehend_train_s3_uri
    },
    OutputDataConfig={
        'S3Uri': s3_output_job
    },
    LanguageCode='en'
)
```

Następnie użyj tego klasyfikatora, aby ocenić sentyment przykładowej *pozytywnej* opinii za pomocą API `detect_sentiment()` z usługi Amazon Comprehend:

```
txt = """I loved it! I will recommend this to everyone."""

response = comprehend.detect_sentiment(
    Text=txt
)
```

Oto dane wyjściowe:

```
{
    "SentimentScore": {
        "Mixed": 0.030585512690246105,
        "Positive": 0.94992071056365967,
        "Neutral": 0.0141543131828308,
        "Negative": 0.00893945890665054
    },
    "Sentiment": "POSITIVE",
    "LanguageCode": "en"
}
```

Teraz użyjemy tego klasyfikatora do oceny sentymentu w przykładowej *negatywnej* opinii, ponownie używając API `detect_sentiment()` z usługi Amazon Comprehend:

```
txt = """Really bad. I hope they don't make this anymore."""

response = comprehend.detect_sentiment(
    Text=txt
)
```

Oto dane wyjściowe dla *negatywnej* opinii:

```
{
    "SentimentScore": {
        "Mixed": 0.030585512690246105,
        "Positive": 0.00893945890665054,
        "Neutral": 0.0141543131828308,
        "Negative": 0.94992071056365967
    },
    "Sentiment": "NEGATIVE",
    "LanguageCode": "en"
}
```

Za pomocą usługi Amazon Comprehend Custom Labels można wyuczyć usługę Amazon Comprehend prognozowania niestandardowych etykiet specyficznych dla zbioru danych.

W rozdziale 3. uczymy w usłudze Amazon Comprehend niestandardowy model, który klasyfikuje wiadomości do działu obsługi klienta za pomocą gwiazdek (od 1 do 5), co jest bardziej precyzyjną wersją analizy sentymentu. Używamy do tego zbioru danych Amazon Customer Reviews.

Pobieranie szczegółowych informacji z CV za pomocą usług Amazon Textract i Comprehend

Organizacje od dawna zmagają się z wydajnym przetwarzaniem częściowo ustrukturyzowanych dokumentów, aby były łatwe do indeksowania i przeszukiwania. Przetwarzanie dokumentów wymaga zwykle daleko posuniętej personalizacji i konfiguracji. Amazon Textract, w pełni zarządzana usługa do dokładnego pobierania tekstu z dokumentu, stosuje optyczne rozpoznawanie znaków i uczenie maszynowe do automatycznego wydobywania informacji z zeskanowanych dokumentów.

Obok optycznego rozpoznawania znaków usługa ta stosuje NLP do parsowania i zapisywania określonych słów, wyrażeń, dat i liczb znalezionych w dokumencie. W połączeniu z usługą Amazon Comprehend Amazon Textract potrafi zbudować i utrzymywać inteligentny indeks zawartości dokumentu. Usługę tę można też zastosować do tworzenia zautomatyzowanych procesów przetwarzania dokumentów i zapewniania zgodności archiwów dokumentów z regulacjami.

Po zakończeniu skanowania i parsowania CV zapisanego w formacie PDF Amazon Textract generuje następującą tekstową wersję danych:

```
NAME
...
LOCATION
...
WORK EXPERIENCE
...
EDUCATION
...
SKILLS
C (Less than 1 year), Database (Less than 1 year),
Database Management (Less than 1 year),
Database Management System (Less than 1 year),
Java (Less than 1 year)
...
TECHNICAL SKILLS
Programming language: C, C++, Java
Oracle PeopleSoft
Internet of Things
Machine Learning
Database Management System
Computer Networks
Operating System worked on: Linux, Windows, Mac
...
NON-TECHNICAL SKILLS
Honest and Hard-Working
```

```
Tolerant and Flexible to Different Situations
Polite and Calm
Team-Player
```

Zobacz teraz, jak wyuczyć usługę Amazon Comprehend rozumienia nowej koncepcji o nazwie „SKILLS" specyficznej dla CV:

```
comprehend_client = boto3.client('comprehend')

custom_recognizer_name = 'resume-entity-recognizer-'+ str(int(time.time()))

comprehend_custom_recognizer_response = \
    comprehend_client.create_entity_recognizer(
        RecognizerName = custom_recognizer_name,
        DataAccessRoleArn = iam_role_textract_comprehend_arn,
        InputDataConfig = {
            'EntityTypes': [
                {'Type': 'SKILLS'},
            ],
            'Documents': {
                'S3Uri': comprehend_input_doucuments
            },
            'EntityList': {
                'S3Uri': comprehend_input_entity_list
            }
        },
        LanguageCode='en'
    )
```

Za pomocą nowego mechanizmu rozpoznawania encji `SKILLS` zbudowanego w usłudze Amazon Comprehend można wykrywać encje w tekstowych CV pobranych wcześniej z formatu PDF za pomocą usługi Amazon Textract:

```
# Uruchamianie zadania rozpoznawania encji:
custom_recognizer_job_name = 'recognizer-job-'+ str(int(time.time()))

recognizer_response = comprehend_client.start_entities_detection_job(
    InputDataConfig = {
        'S3Uri': s3_test_document,
        'InputFormat': 'ONE_DOC_PER_LINE'
    },
    OutputDataConfig = {
        'S3Uri': s3_test_document_output
    },
    DataAccessRoleArn = iam_role_textract_comprehend_arn,
    JobName = custom_recognizer_job_name,
    EntityRecognizerArn = comprehend_model_response['EntityRecognizerProperties']\
                          ['EntityRecognizerArn'],
    LanguageCode = 'en'
)
```

Oto dane wyjściowe z niestandardowego mechanizmu rozpoznawania encji z usługi Amazon Comprehend. Podane są tu tekst, lokalizacja w dokumencie, rodzaj encji (`SKILLS`) i poziom pewności co do poprawności wybranej kategorii:

Początek	Koniec	Pewność	Tekst	Typ
9	39	0,9574943836014351	analytical and problem solving	SKILLS
8	11	0,7915781756343004	AWS	SKILLS
33	41	0,9749685544856893	Solution	SKILLS
20	23	0,9997213663311131	SQL	SKILLS
2	13	0,9996676358048374	Programming	SKILLS
25	27	0,9963501364429431	C,	SKILLS
28	32	0,9637213743240001	C++,	SKILLS
33	37	0,9984518452247634	Java	SKILLS
39	42	0,9986466628533158	PHP	SKILLS
44	54	0,9993487072806023	JavaScript	SKILLS

Wyszukiwanie kognitywne i rozumienie języka naturalnego

Każdemu zdarza się mieć problem ze znalezieniem potrzebnej informacji ukrytej gdzieś w witrynie, firmowym systemie zarządzania treścią, korporacyjnej wiki lub współużytkowanym zasobie plikowym. Każdy wie też, jak irytujące jest wielokrotne odpowiadanie na te same często zadawane pytania.

Próba szybkiego zwracania adekwatnych wyników nie jest nowym problemem. Do jego rozwiązania powstało wiele otwartych narzędzi takich jak Apache Lucene, Apache SOLR i Elasticsearch. Te rozwiązania są oparte na starszych, opracowanych wiele lat temu technikach NLP. Użytkownicy takich narzędzi zwykle szukają informacji na podstawie słów kluczowych, które wpisane błędnie lub w niewłaściwej kolejności mogą prowadzić do otrzymania mało wartościowych wyników wyszukiwania.

Wyszukiwanie kognitywne to nowe rozwiązanie starego problemu znajdowania informacji. Jest oparte na rozumieniu języka naturalnego (ang. *natural language understanding* — NLU) i umożliwia użytkownikom zadawanie pytań w języku naturalnym, w formie stosowanej przez ludzi.

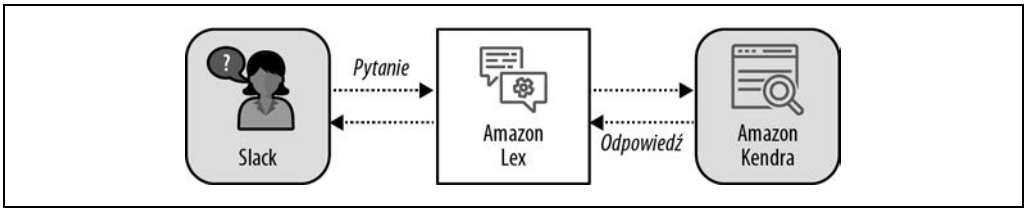
Usługa Amazon Kendra używa uczenia maszynowego, NLU i wyszukiwania kognitywnego do rozwiązywania w nowoczesny sposób problemu wyszukiwania w firmach. Zamiast stosować tradycyjne wyszukiwanie oparte na słowach kluczowych, których wybranie wymaga dodatkowego wysiłku, można zadawać tej usłudze kompletne pytania w języku naturalnym, na przykład „On which floor is the IT department located in this office?”, i otrzymać konkretną odpowiedź: „19th floor”.

Usługę Amazon Kendra można zintegrować z wieloma różnymi źródłami danych takimi jak: Amazon S3, SharePoint, Salesforce, ServiceNow, Amazon RDS Databases, OneDrive i wiele innych. Kendra obsługuje wszystkie rodzaje schematów danych, w tym ustrukturyzowane, nieustrukturyzowane i częściowo ustrukturyzowane. Używa też wielu różnych formatów takich jak: PDF, HTML, RTF, Microsoft Word i PowerPoint.

Choć Kendra ma wbudowane wstępnie wyuczone modele zoptymalizowane na potrzeby różnych dziedzin, można wyuczyć tę usługę na podstawie własnych zbiorów danych, aby poprawić trafność

wyników. Ponadto Kendra aktywnie uczy się i aktualizuje na podstawie wzorców użytkowania, w tym z wykorzystaniem bezpośrednio podawanych informacji zwrotnych w postaci kliknięcia kciuka w górę lub kciuka w dół przy wynikach wyszukiwania.

Łącząc usługi Amazon Kendra i Lex, można zbudować chatbota do obsługi klienta, który będzie działać na różnych urządzeniach i odpowiadać na często zadawane pytania. W tym przykładzie używamy dodatkowo popularnego narzędzia do współpracy zespołowej, Slack (zobacz rysunek 2.3).



Rysunek 2.3. Używanie narzędzi Slack, Amazon Lex i Amazon Kendra do automatycznego odpowiadania na pytania

Poniżej przedstawiamy przykładową rozmowę zarejestrowaną, gdy Antje używała Slackbota do odpowiadania na pytania w trakcie warsztatów prowadzonych na żywo. Ten chatbot odpowiada na często zadawane przez uczestników warsztatów pytania. Dzięki temu prowadzący warsztaty może skupić się na bardziej skomplikowanych kwestiach, które wymagają interwencji człowieka:

Antje: „Hi there””

Slackbot: „Hello! How can I help?”

Antje: „Do you record this workshop?”

Slackbot: „Yes, this workshop is being recorded”.

Antje: „Where do I find the recording?”

Slackbot: „The recording will be shared at https://youtube.datascienceonaws.com within 24 hours”.

Antje: „Do you know how to get started with SageMaker?”

Slackbot: „I think the answer to your question is: On the Amazon SageMaker Studio page, under Get Started, choose Quick Start, then select Create Studio Domain”.

Inteligentne centra obsługi klienta

Wysokiej jakości obsługa klienta jest ważna w każdej branży i firmie (wcześniej wspomnieliśmy, że „obsesja na punkcie klienta” jest kluczową zasadą obowiązująca w Amazonie). W wielu sytuacjach poziom obsługi klienta bezpośrednio wpływa na to, jak dana osoba postrzega firmę. Amazon Connect to chmurowa usługa wspomagania centrum obsługi klienta, stosująca uczenie maszynowe do udostępniania inteligentnych funkcji. Dzięki funkcji Connect Wisdom pracownik działu obsługi klienta może wprowadzić pytanie, na przykład „What is the exchange policy for books?”, a system zwróci

najbardziej adekwatne informacje i najlepszą odpowiedź. Funkcja Connect Wisdom uruchamia też uczenie maszynowe, by przetwarzać zapisy prowadzonych na żywo rozmów, automatycznie identyfikować problemy klienta i rekomendować pracownikom odpowiedzi.

Funkcja Contact Lens for Amazon Connect dodaje mechanizmy uczenia maszynowego do usługi Amazon Connect, chmurowej usługi wspomagania centrum obsługi klienta opartej na technologiach używanych w centrum obsługi klienta Amazona. W funkcji Contact Lens stosuje się transkrypcję mowy na tekst, NLP i wyszukiwanie kognitywne do analizowania interakcji klientów z pracownikami.

Dzięki automatycznemu indeksowaniu transkrypcji rozmów Contact Lens pozwala wyszukiwać konkretne słowa i wyrażenia oraz oceniać sentyment, a także usuwać wrażliwe informacje z zapisów, aby uniknąć ich wycieku. Contact Lens pomaga menedżerom wykrywać w czasie rzeczywistym powtarzające się tematy interakcji, automatycznie uczyć agenty, aby zwiększać ich umiejętności z zakresu obsługi klienta, i na bieżąco kategoryzować interakcje na podstawie słów kluczowych i wyrażeń używanych przez klientów.

Dzięki funkcji Contact Lens for Amazon Connect menedżerowie centrum obsługi klienta mogą w jednym miejscu przyjrzeć się interakcjom klientów z agentami, trendom w opiniach na temat produktów i ewentualnym naruszeniom zgodności z regulacjami. Amazon Connect powtarza udane interakcje, zwraca uwagę na anomalie w opiniach na temat produktów i przekazuje informacje o niskiej jakości interakcjach klienta z agentem do menedżera.

Przemysłowe usługi SI i konserwacja predykcyjna

W ramach zestawu usług przemysłowych platformy AWS dostępne są rozmaite usługi i sprzęt z obszaru SI, w tym Amazon Lookout for Metrics, Lookout for Vision, Lookout for Equipment, Amazon Monitron i AWS Panorama.

Za pomocą usługi Amazon Lookout for Metrics można tworzyć precyzyjne modele wykrywania anomalii. Po wczytaniu danych usługa automatycznie bada je i tworzy model wykrywania anomalii. Jeśli model je znajdzie, usługa pogrupuje powiązane anomalie ze sobą i przypisze im poziom krytyczności. Usługa Lookout for Metrics ma wbudowane konektory dla często używanych źródeł danych takich jak Amazon S3, Amazon Redshift, Amazon CloudWatch, Amazon RDS i różne aplikacje SaaS. Model wykrywania anomalii korzysta z informacji zwrotnych od człowieka, aby stale zwiększać swój poziom.

Defekty w produktach można wykrywać za pomocą usługi Amazon Lookout for Vision. Wykorzystuje ona rozpoznawanie obrazów do identyfikowania wizualnych usterek w obiektach. Pomaga zautomatyzować wykrywanie uszkodzeń części, identyfikowanie brakujących komponentów lub odkrywanie problemów w procesach na liniach produkcyjnych. Usługa Lookout for Vision posiada wstępnie wyuczony model wykrywania anomalii. Wystarczy dostroić go do używanych obrazów.

Można też monitorować stan i wydajność sprzętu, używając usługi Amazon Lookout for Equipment. Należy przesłać historyczne dane z czujników urządzeń, a usługa ta zbuduje niestandardowy model oparty na uczeniu maszynowym, który będzie wykrywał nietypowe działanie sprzętu.

Dodatkowo usługa przesyła alerty, dzięki czemu można podjąć odpowiednie działania. Amazon Lookout for Equipment działa dla dowolnych szeregów czasowych z analogowymi danymi (na przykład z czujników temperatury, prędkości przepływu itd.).

Narzędzie Amazon Monitron umożliwia wdrożenie kompleksowej konserwacji predykcyjnej. Obejmuje ono czujniki sprzętowe, bramę do bezpiecznego łączenia się z platformą AWS i zarządzaną usługę do analizowania danych pod kątem wzorców oznaczających nietypowe działanie maszyn. Amazon Monitron rejestruje dane z czujników, identyfikuje typowe wzorce danych z czujników i tworzy model oparty na uczeniu maszynowym dostosowany do danego sprzętu. Można też przekazywać informacje zwrotne za pomocą aplikacji mobilnej Amazon Monitron, aby usprawnić działanie modelu.

Narzędzie AWS Panorama umożliwia zastosowanie rozpoznawania obrazów do materiału z kamer monitoringu. To narzędzie obejmuje urządzenia, które da się podłączyć do sieci i istniejących kamer. Następnie można zainstalować w tym urządzeniu aplikacje do rozpoznawania obrazów, aby przetwarzać strumienie wideo z podłączonych kamer. Producenci kamer mogą korzystać z pakietu SDK AWS Panorama do projektowania nowych kamer, które uruchamiają modele rozpoznawania obrazów „na obrzeżu".

Automatyzacja domu za pomocą narzędzi AWS IoT i Amazon SageMaker

Żyjemy w świecie, w którym mniej więcej pięć miliardów osób posiada jakiegoś rodzaju urządzenie mobilne. Ponad połowa ruchu w internecie jest generowana przy użyciu takich urządzeń. Ponadto przemysłowa rewolucja IoT doprowadziła do zainstalowania miliardów podłączonych do sieci czujników i urządzeń w mieszkaniach, biurach, fabrykach, samochodach, statkach, samolotach, na platformach wiertniczych, polach uprawnych itd.

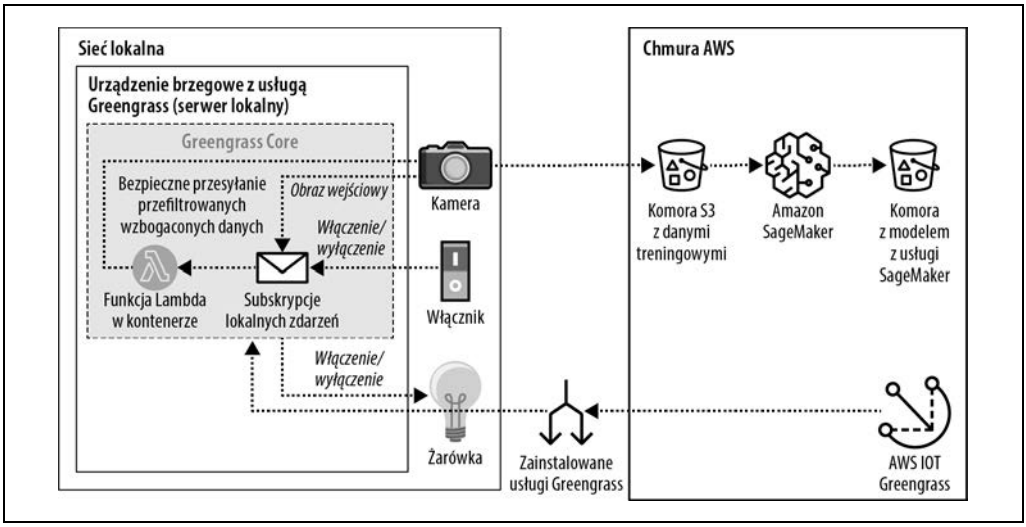
Trend przechodzenia na urządzenia mobilne i IoT powoduje też przenoszenie obliczeń na obrzeża sieci niezależnie od tego, czy celem jest analiza i wstępne przetwarzanie danych przed ich przesłaniem do scentralizowanych jezior danych (na przykład na potrzeby zachowania zgodności z regulacjami z obszaru prywatności danych), czy poprawa doświadczeń użytkowników w wyniku szybszego udostępniania odpowiedzi w aplikacji dzięki wyeliminowaniu opóźnienia związanego z przesyłaniem danych do chmury i z powrotem. Także niektóre etapy uczenia maszynowego coraz częściej są wykonywane na obrzeżach sieci. Choć trening modeli z obszaru uczenia maszynowego niejednokrotnie wymaga dużej ilości zasobów obliczeniowych, wnioskowanie na podstawie tych modeli jest zwykle dużo mniej wymagające obliczeniowo.

Wnioskowanie na obrzeżach sieci pomaga zmniejszyć opóźnienia i koszty, ponieważ nie trzeba przesyłać danych do chmury i z powrotem. Pozwala to również szybciej rejestrować i analizować predykcje, uruchamiać lokalnie określone działania lub przesyłać przeanalizowane dane z powrotem do chmury, aby ponowić uczenie maszynowe i usprawnić model.

Usługa AWS IoT Greengrass instaluje model z S3 w urządzeniu na obrzeżach sieci, aby generować predykcje na podstawie lokalnych danych z tego urządzenia. Usługa synchronizuje też wyniki wnioskowania wygenerowane przez model z komorą S3. Dane o predykcjach można następnie

wykorzystać do ponownej nauki i usprawniania modelu w usłudze SageMaker. Usługa AWS IoT Greengrass obsługuje instalowanie bezprzewodowe, działa lokalnie na każdym urządzeniu i stanowi rozszerzenie platformy AWS w urządzeniach.

Na rysunku 2.4 przedstawiony jest scenariusz automatyzacji domu. Usługa AWS IoT Greengrass działa tu na lokalnym serwerze automatyzacji domu nazywanym „urządzeniem brzegowym". Usługa instaluje model z SageMakera w urządzeniu brzegowym i przetwarza dane z kamer, włączników światła i żarówek, używając brzegowej wersji funkcji Lambda działającej na wspomnianym urządzeniu.



Rysunek 2.4. Automatyzacja domu z wykorzystaniem usługi AWS IoT Greengrass

AWS udostępnia różne usługi pozwalające stosować uczenie maszynowe na obrzeżach sieci. Na przykład AWS IoT Greengrass umożliwia instalowanie modeli, SageMaker Neo służy do optymalizowania modeli, a SageMaker Edge Manager — do zarządzania modelami na obrzeżach sieci. Szczegółowe omówienie usług SageMaker Neo i Edge Manager znajdziesz w rozdziale 9.

Pobieranie informacji medycznych z dokumentów służby zdrowia

Platforma AWS udostępnia wiele wyspecjalizowanych usług dla branży opieki zdrowotnej. Te usługi zostały specjalnie opracowane z uwzględnieniem cech i wymogów związanych z danymi z tej branży oraz z zachowaniem zgodności z regulacjami. Zestaw usług SI oferowanych przez Amazon dla branży opieki zdrowotnej, dostosowanych do ustawy HIPAA obejmuje narzędzia Amazon Comprehend Medical, Amazon Transcribe Medical i Amazon HealthLake.

Comprehend Medical to usługa NLP wstępnie wyuczona do rozumienia języka medycznego. Comprehend Medical automatyzuje pobieranie danych medycznych z tekstów takich jak notatki lekarzy, raporty z badań klinicznych lub rejestry medyczne pacjentów.

Transcribe Medical to usługa automatycznego rozpoznawania mowy, która także jest wstępnie wyuczona na podstawie języka medycznego. Za pomocą tej usługi można przekształcać wypowiedzi medyczne na tekst. Za pomocą prostych wywołań API można zautomatyzować proces tworzenia dokumentacji medycznej, a nawet dodać napisy w rozmowach telemedycznych.

HealthLake to bezpieczne jezioro danych zgodne ze standardem branżowym Fast Healthcare Interoperability Resources. Obok składowania, indeksowania i przekształcania danych medycznych usługa ta stosuje uczenie maszynowe do identyfikowania, rozumienia i pobierania informacji medycznych z surowych danych (na przykład z raportów medycznych i notatek pacjentów). Usługi Amazon QuickSight, Athena i SageMaker umożliwiają przeprowadzanie zaawansowanych analiz danych medycznych oraz uczenie maszynowe na ich podstawie.

Samooptymalizująca i inteligentna infrastruktura chmury

Usługi SI i UM Amazona, które przedstawiliśmy do tej pory, nie są jedynymi narzędziami do zaawansowanego uczenia maszynowego. Coraz więcej istniejących usług z platformy AWS jest wzbogacanych o funkcje uczenia maszynowego. Ponadto wprowadza się nowe, oparte na uczeniu maszynowym usługi do wykonywania rozmaitych zadań. Przyjrzyj się niektórym z tych ukrytych perełek.

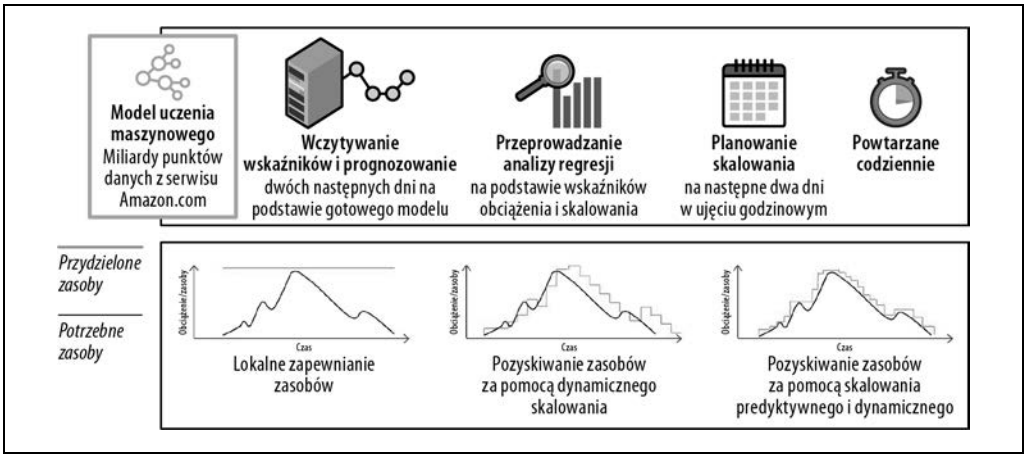
Predyktywne automatyczne skalowanie w Amazon EC2

Amazon EC2 (ang. *Elastic Compute Cloud*) zapewnia instancje wirtualnych serwerów w chmurze AWS. Jedną z trudności przy uruchamianiu aplikacji w instancjach Amazon EC2 jest dostosowywanie liczby instancji do aktualnego obciążenia, czyli dopasowywanie podaży do popytu. Na szczęście istnieje usługa Amazon EC2 Auto Scaling, która w tym pomaga. Można tak skonfigurować tę usługę, aby na podstawie zmian w zapotrzebowaniu automatycznie dodawała lub zwalniała zasoby obliczeniowe. Tego typu dynamiczne skalowanie nadal jest jednak rozwiązaniem reaktywnym, ponieważ opiera się na monitorowaniu ruchu i wskaźnikach obciążenia instancji Amazon EC2.

Można przejść na wyższy poziom i zastosować proaktywne podejście w połączeniu z usługą *AWS Auto Scaling*. Zapewnia ona pojedynczy interfejs, w którym można skonfigurować automatyczne skalowanie wielu usług platformy AWS, w tym Amazon EC2. AWS Auto Scaling łączy skalowanie dynamiczne z predyktywnym. Na potrzeby skalowania predyktywnego AWS używa algorytmów uczenia maszynowego do prognozowania przyszłego ruchu na podstawie trendów dziennych i tygodniowych oraz przygotowuje odpowiednią liczbę instancji Amazon EC2 zgodnie z oczekiwanymi zmianami (zobacz rysunek 2.5).

Wykrywanie anomalii w strumieniach danych

Technologie strumieniowania zapewniają narzędzia do pobierania, przetwarzania i analizowania strumieni danych w czasie rzeczywistym. AWS udostępnia wiele technologii strumieniowania, w tym Amazon MSK i Amazon Kinesis. Dokładne omówienie analizy strumieni danych i uczenia maszynowego z użyciem narzędzi Amazon Kinesis and Apache Kafka znajduje się w rozdziale 10., tu chcemy zwrócić uwagę na usługę Kinesis Data Analytics, która jest prostym, a jednocześnie dającym



Rysunek 2.5. Skalowanie predyktywne za pomocą usługi AWS Auto Scaling pozwala przewidzieć zmiany w ruchu i przydzielić odpowiednią liczbę instancji Amazon EC2

duże możliwości mechanizmem do tworzenia za pomocą kilku wierszy kodu aplikacji korzystających ze strumieni danych.

Kinesis Data Analytics obejmuje narzędzie do wykrywania anomalii. Jest nim algorytm Random Cut Forest (RCF), który umożliwia budowanie modeli uczenia maszynowego w czasie rzeczywistym i obliczanie wskaźnika anomalii dla wartości liczbowych z poszczególnych wiadomości. Wskaźnik ten określa, jak różni się dana wartość od zaobserwowanego trendu. Algorytm RCF oblicza też wskaźnik wkładu każdej kolumny, odzwierciedlający, na ile nietypowe są dane w poszczególnych kolumnach. Suma wskaźników wkładu wszystkich kolumn daje łączny wskaźnik anomalii.

Kognitywna i predyktywna analityka biznesowa

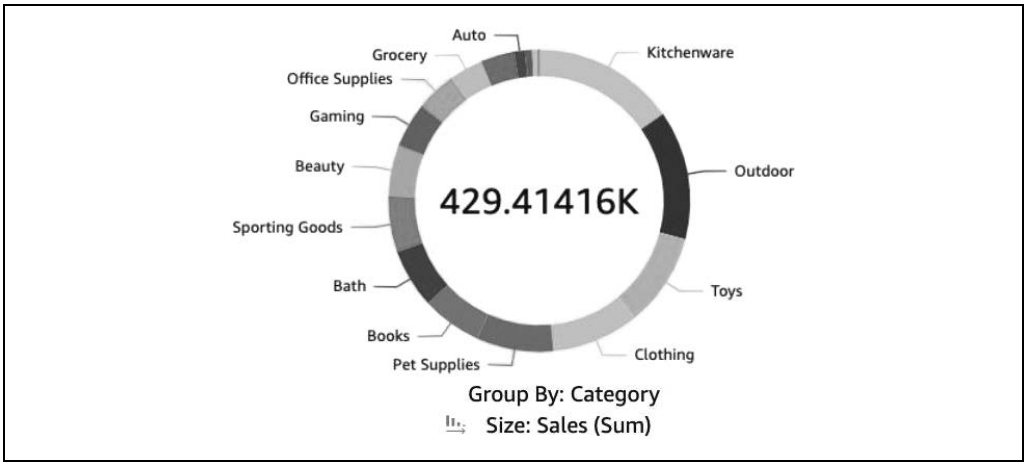
Wiele aplikacji i modeli z obszaru uczenia maszynowego wymaga, aby dane były dostępne w jeziorze danych (jeziora danych omawiamy w rozdziale 4.). Jednak w praktyce duża część danych jest składowana i przetwarzana w ustrukturyzowanych bazach relacyjnych. Aby można było zastosować uczenie maszynowe do danych ustrukturyzowanych, trzeba albo wyeksportować dane, albo opracować niestandardową aplikację, która wczyta dane przed rozpoczęciem uczenia maszynowego. Czy nie byłoby idealnie, gdyby można było przeprowadzać uczenie maszynowe bezpośrednio w usłudze do analityki biznesowej, w hurtowni danych lub w bazie? Zobacz, jak zrobić to na platformie AWS.

Zadawanie pytań w języku naturalnym za pomocą usługi Amazon QuickSight

Amazon QuickSight to usługa do analityki biznesowej, która generuje interaktywne zapytania i tworzy wizualizacje na podstawie źródeł danych takich jak Amazon Redshift, Amazon RDS, Amazon Athena i Amazon S3. QuickSight potrafi też wykrywać anomalie, generować prognozy i odpowiadać na pytania w języku naturalnym, używając funkcji QuickSight ML Insights i QuickSight Q.

Funkcja QuickSight ML Insights uruchamia algorytm RCF, by wykrywać zmiany w milionach wskaźników w miliardach punktów danych. Ta funkcja umożliwia też prognozowanie na podstawie zaobserwowanych wskaźników. Algorytm RCF automatycznie wykrywa sezonowość w danych, wyklucza wartości odstające i uzupełnia brakujące wartości.

Za pomocą funkcji QuickSight Q można zadawać pytania w języku naturalnym, na przykład „What are the best-selling product categories in the US state of California?". QuickSight używa uczenia maszynowego, aby zrozumieć pytanie, po czym stara się znaleźć odpowiedź w danych i utworzyć reprezentujący ją wykres (zobacz rysunek 2.6). Dokładne omówienie usługi QuickSight znajdziesz w rozdziale 5.



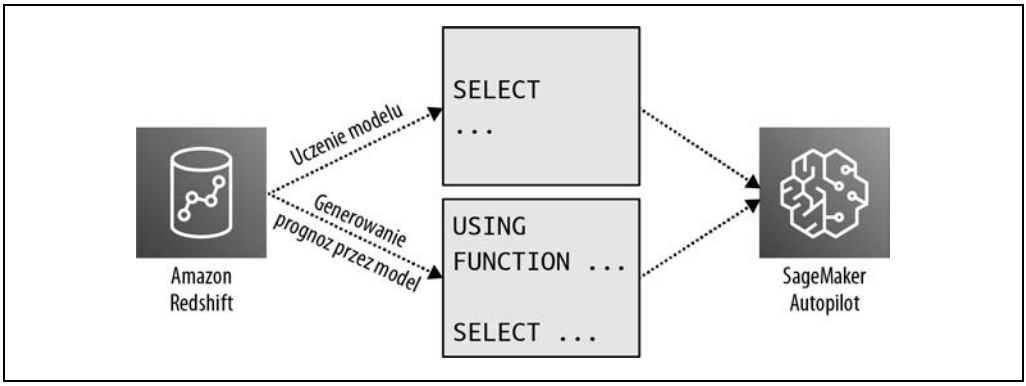
Rysunek 2.6. Funkcja QuickSight Q rozumie pytania w języku naturalnym i automatycznie generuje wykresy z odpowiedziami

Uczenie i wywoływanie modeli z usługi SageMaker w hurtowni Amazon Redshift

Amazon Redshift to w pełni zarządzana hurtownia danych, która umożliwia uruchamianie złożonych zapytań analitycznych na petabajtach ustrukturyzowanych danych. Dzięki usłudze Amazon Redshift ML można używać tej hurtowni do uczenia modeli za pomocą usługi SageMaker Autopilot w reakcji na pojawianie się nowych danych. SageMaker Autopilot automatycznie uczy, dostraja i instaluje model. Następnie można zarejestrować i wywoływać model jako funkcje zdefiniowane przez użytkownika w zapytaniach w hurtowni Amazon Redshift. Na rysunku 2.7 pokazane jest generowanie predykcji za pomocą klauzuli SQL-owej `USING FUNCTION`. Bardziej szczegółowy przykład zastosowania usług Amazon Redshift ML i SageMaker Autopilot znajdziesz w rozdziale 3.

Wywoływanie modeli z usług Amazon Comprehend i SageMaker w SQL-owej bazie Amazon Aurora

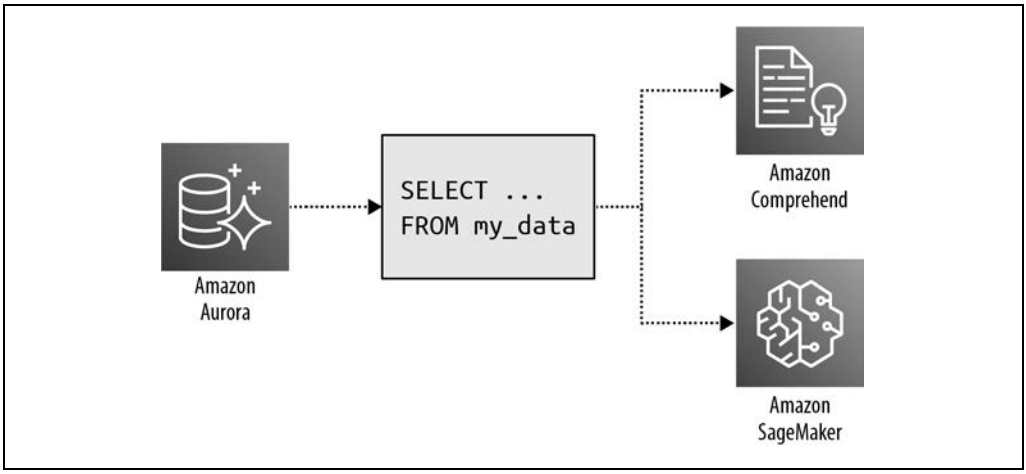
Aurora to relacyjna baza danych zgodna z bazami MySQL i PostgreSQL, zintegrowana z usługami Amazon Comprehend i Amazon SageMaker (zobacz rysunek 2.8).



Rysunek 2.7. Hurtownia Amazon Redshift używa usługi SageMaker Autopilot do uczenia i wywoływania modelu z usługi SageMaker w formie funkcji zdefiniowanych przez użytkownika

Funkcje zdefiniowane przez użytkownika wywołujące dowolne usługi platformy AWS można utworzyć za pomocą funkcji Lambda. Poniższa przykładowa funkcja zdefiniowana przez użytkownika wywołuje funkcję Lambda:

```
USING FUNCTION invoke_lambda(input VARCHAR)
RETURNS VARCHAR TYPE LAMBDA_INVOKE WITH
(lambda_name='<NAZWA_FUNKCJI_LAMBDA>') SELECT invoke('<DANE_WEJŚCIOWE>');
```



Rysunek 2.8. Aurora ML potrafi wywoływać modele z usług Amazon Comprehend i SageMaker

W zapytaniach można używać albo wbudowanych funkcji SQL-owych (dla usługi Amazon Comprehend), albo niestandardowych funkcji SQL-owych (dla usługi Amazon SageMaker), by przeprowadzić uczenie maszynowe na podstawie danych. We wcześniejszych punktach pokazaliśmy, że można zastosować usługę Amazon Comprehend do analizy sentymentu klientów (za pomocą wbudowanych funkcji SQL-owych) na podstawie recenzji produktów lub usługę Amazon SageMaker do integracji niestandardowych modeli uczenia maszynowego.

Załóżmy, że w tabeli relacyjnej zapisane są przykładowe recenzje produktu:

```
CREATE TABLE IF NOT EXISTS product_reviews (
       review_id INT AUTO_INCREMENT PRIMARY KEY,
       review_body VARCHAR(255) NOT NULL
);

INSERT INTO product_reviews (review_body)
VALUES ("Great product!");
INSERT INTO product_reviews (review_body)
VALUES ("It's ok.");
INSERT INTO product_reviews (review_body)
VALUES ("The worst product.");
```

Następnie można użyć wbudowanych funkcji SQL-owych, aby usługa Amazon Comprehend zwróciła informacje o sentymencie i poziomie pewności:

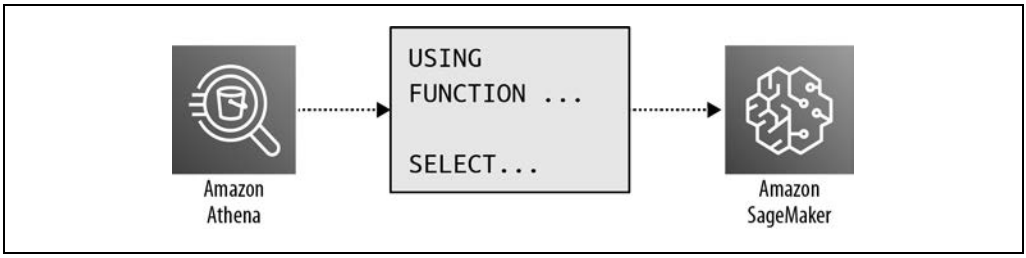
```
SELECT review_body,
       aws_comprehend_detect_sentiment(review_body, 'en') AS sentiment,
       aws_comprehend_detect_sentiment_confidence(review_body, 'en') AS confidence
FROM product_reviews;
```

Ten kod da wyniki podobne do poniższych:

```
review_body         sentiment       confidence
-------------------------------------------------------
Great product!      POSITIVE        0.9969872489
It's ok.            POSITIVE        0.5987234553
The worst product.  NEGATIVE        0.9876742876
```

Wywoływanie modelu z usługi SageMaker w usłudze Amazon Athena

Można też użyć usługi Amazon Athena, która pozwala pobierać dane z Amazon S3 za pomocą zapytań SQL-owych, do wywoływania modeli uczenia maszynowego z usługi SageMaker i wyciągania wniosków bezpośrednio na poziomie zapytań (zobacz rysunek 2.9).



Rysunek 2.9. Usługa Amazon Athena umożliwia wywoływanie modeli z usługi SageMaker

Tu używamy SQL-owej instrukcji `USING FUNCTION` do zdefiniowania funkcji, która wywołuje niestandardowy punkt końcowy z usługi Amazon SageMaker zwracający oceny sentymentu. Wszystkie kolejne instrukcje `SELECT` w zapytaniu mogą używać takiej funkcji, aby przekazywać wartości do modelu.

Oto prosty przykład:

```
USING FUNCTION predict_sentiment(review_body VARCHAR(65535))
    RETURNS VARCHAR(10) TYPE
    SAGEMAKER_INVOKE_ENDPOINT WITH (sagemaker_endpoint = '<NAZWA_PUNKTU_KOŃCOWEGO>')

SELECT predict_sentiment(review_body) AS sentiment
    FROM "dsoaws"."amazon_reviews_tsv"
    WHERE predict_sentiment(review_body)="POSITIVE";
```

Generowanie predykcji na podstawie danych grafowych za pomocą bazy Amazon Neptune

Amazon Neptune to w pełni zarządzana grafowa baza danych, która umożliwia tworzenie i uruchamianie aplikacji opartych na wysoce powiązanych zbiorach danych. Funkcja Neptune ML stosuje grafowe sieci neuronowe do generowania predykcji na podstawie danych grafowych. Algorytmy takie jak XGBoost zostały opracowane dla tradycyjnych tabelowych zbiorów danych, natomiast grafowe sieci neuronowe są specjalnie projektowane w taki sposób, aby radziły sobie ze złożonością typową dla grafów i nawet miliardami powiązań. Funkcja Neptune ML korzysta z otwartej biblioteki Deep Graph i usługi Amazon SageMaker do automatycznego wybierania, uczenia i instalowania najlepszego modelu dla używanych danych grafowych.

Edukacja następnego pokolenia programistów SI i UM

Amazon i AWS oferują wiele programów oraz usług pomocnych w edukacji następnego pokolenia programistów SI i UM. Program Amazon's Machine Learning University (*https://oreil.ly/CnXwM*), według którego szkoleni byli pracownicy Amazona, został w 2020 roku publicznie udostępniony. Jednostka AWS Training and Certification (T&C) oferuje szeroki zakres szkoleń (internetowych i stacjonarnych) przygotowujących do uzyskania certyfikatu AWS Machine Learning. Ponadto AWS współpracuje z serwisami Udacity, Coursera i DeepLearning.AI nad przygotowywaniem różnych masowych otwartych kursów online, które zapewniają uczestnikom praktyczne doświadczenie z zestawem narzędzi Amazona przeznaczonych do SI i UM.

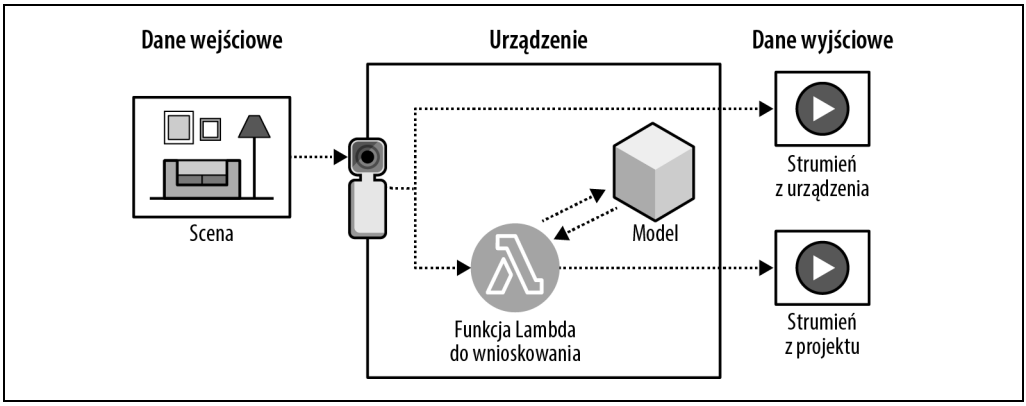
W tym podrozdziale omawiamy urządzenia AWS oparte na uczeniu głębokim, które są ciekawym i edukacyjnym narzędziem pozwalającym w praktyce zapoznać się z rozpoznawaniem obrazów, uczeniem przez wzmacnianie i sieciami GAN (ang. *generative adversarial network*).

Ta rodzina przeznaczonych głównie dla programistów urządzeń obejmuje następujące produkty: AWS DeepLens, DeepRacer i DeepComposer. AWS DeepLens to bezprzewodowa kamera wideo z funkcją uczenia głębokiego. AWS DeepRacer to w pełni autonomiczny samochodzik wyścigowy w skali 1:18, który porusza się na podstawie uczenia przez wzmacnianie. AWS DeepComposer to keyboard, który za pomocą sieci GAN przekształca grane przez użytkownika melodie w oryginalne utwory.

Budowanie modeli rozpoznawania obrazów za pomocą urządzenia AWS DeepLens

AWS DeepLens to kamera wideo z funkcją uczenia głębokiego oferowana z bogatym zestawem samouczków z obszaru rozpoznawania obrazów i gotowymi modelami. Jeśli chcesz się nauczyć, jak tworzyć aplikacje do rozpoznawania obrazów, i zobaczyć pierwsze efekty w przeciągu kilku minut, możesz użyć jednego z wielu przykładowych projektów obejmujących gotowe modele i proste funkcje wnioskowania. Kamera wykonuje wtedy wnioskowanie lokalnie w urządzeniu na podstawie zainstalowanego modelu.

Jeśli masz więcej doświadczenia, możesz zbudować i wyuczyć niestandardowy model oparty na konwolucyjnych sieciach neuronowych, wykorzystując dowolną z obsługiwanych platform uczenia głębokiego takich jak TensorFlow, Apache MXNet lub Caffe, a następnie zainstalować projekt w urządzeniu AWS DeepLens. Na rysunku 2.10 pokazany jest typowy proces pracy z urządzeniem AWS DeepLens.



Rysunek 2.10. AWS DeepLens rejestruje wejściowe strumienie wideo, przetwarza je za pomocą zainstalowanego modelu i generuje dwa wyjściowe strumienie wideo

AWS DeepLens jest jednocześnie urządzeniem brzegowym i kamerą. Dlatego uruchamia oprogramowanie AWS IoT Greengrass Core i może wykonywać własne funkcje Lambda. Nowe modele są wczytywane do urządzenia AWS DeepLens za pomocą usługi AWS IoT Greengrass. Kamera rejestruje wejściowy strumień wideo i generuje dwa strumienie wyjściowe: przekazywany bez zmian strumień z urządzenia i dodatkowo strumień z projektu, który jest wynikiem przetworzenia klatek nagrania przez zainstalowany model.

Każdy instalowany projekt musi obejmować funkcję Lambda (*funkcję wnioskowania*), która będzie przetwarzać wejściowe klatki nagrania. Najpierw należy połączyć tę funkcję ze środowiskiem uruchomieniowym Lambda i z wyuczonym modelem. Następnie można zainstalować projekt za pomocą usługi AWS IoT Greengrass w urządzeniu AWS DeepLens.

Poznawanie uczenia przez wzmacnianie za pomocą urządzenia AWS DeepRacer

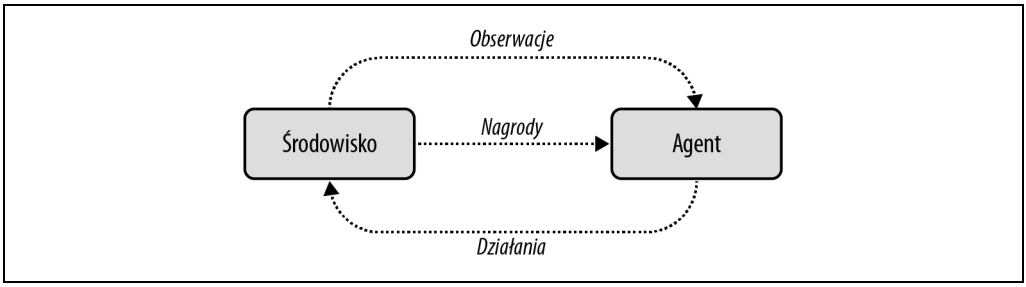
AWS DeepRacer to w pełni autonomiczny samochodzik w skali 1:18 poruszający się z wykorzystaniem uczenia przez wzmacnianie. Samochodzik jest wyposażony w dwie kamery, czujnik LiDAR i zintegrowany moduł obliczeniowy. Moduł ten przeprowadza wnioskowanie i steruje ruchem pojazdu.

Uczenie przez wzmacnianie stosuje się w rozmaitych problemach z dziedziny autonomicznego podejmowania decyzji. Podejście to zyskało popularność, gdy zespół naukowców, inżynierów i specjalistów od uczenia maszynowego z firmy DeepMind (*https://deepmind.com*) pokazał AlphaGo — pierwszy program komputerowy, który pokonał zawodowego gracza w go (miało to miejsce w 2015 roku).

Go to starożytna strategiczna gra planszowa znana ze złożoności. Została wymyślona w Chinach mniej więcej trzy tysiące lat temu i nadal grają w nią zarówno amatorzy, jak i zawodowcy w profesjonalnych ligach na całym świecie.

Choć AlphaGo uczył się gry na podstawie tysięcy partii rozgrywanych przeciwko ludziom, kolejna wersja programu, AlphaGo Zero, uczyła się wyłącznie na podstawie gry ze sobą. Była to następna rewolucja w dziedzinie uczenia przez wzmacnianie, ponieważ program ten osiągnął jeszcze wyższy poziom i pokazał, że potrafi odkrywać nową wiedzę i stosować niekonwencjonalne strategie, aby zwyciężyć.

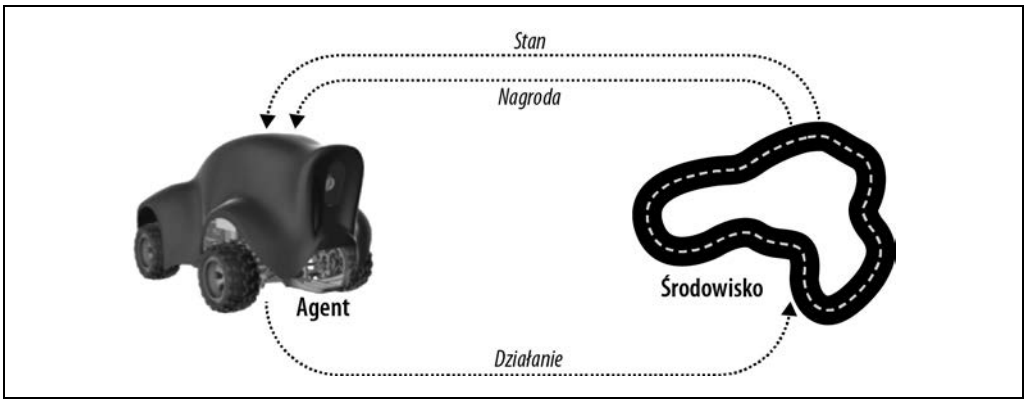
Na ogólnym poziomie uczenie przez wzmacnianie to metoda uczenia maszynowego, która ma umożliwić podejmowanie autonomicznych decyzji przez agenta dążącego do osiągnięcia określonego celu w wyniku interakcji ze środowiskiem (zobacz rysunek 2.11). Nauka odbywa się metodą prób i błędów.



Rysunek 2.11. Uczenie przez wzmacnianie to metoda uczenia maszynowego, która ma umożliwiać podejmowanie autonomicznych decyzji przez agenta dążącego do określonego celu w wyniku interakcji ze środowiskiem

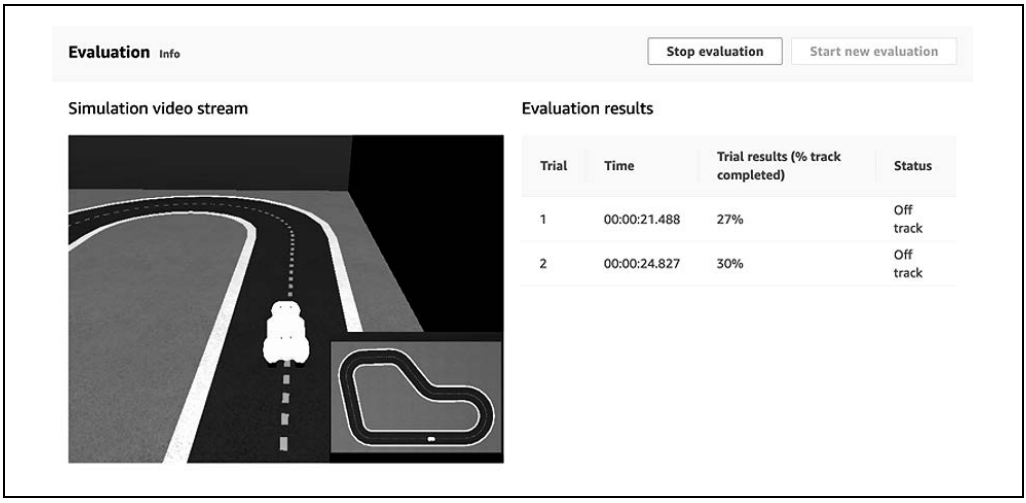
Dokładne omówienie uczenia przez wzmacnianie i porównywania modeli w środowisku produkcyjnym za pomocą „wielorękich bandytów" przedstawiamy w rozdziale 9., a na razie wróćmy do autonomicznego samochodzika. W tym przykładzie agentem jest pojazd AWS DeepRacer, a środowisko obejmuje układ toru, trasy przejazdu i warunki drogowe. Działania to skręt w lewo, skręt w prawo,

hamowanie i przyspieszanie. Są one tak wybierane przez agenta, aby zmaksymalizować wartość nagrody powiązanej z celem, jakim jest szybkie dotarcie do celu bez wypadków. Działania prowadzą do stanów. Na rysunku 2.12 pokazane są działania, nagrody i stany samochodzika AWS DeepRacer.



Rysunek 2.12. AWS DeepRacer podejmuje działania na podstawie stanu i nagrody

Nie potrzeba nawet fizycznego toru lub samochodziku, aby rozpocząć zabawę. Na początku można trenować niestandardowy model uczenia przez wzmacnianie w konsoli AWS DeepRacer i użyć symulatora do oceny jakości modelu na wirtualnym torze, co ilustruje rysunek 2.13.



*Rysunek 2.13. Ocena modelu w symulatorze AWS DeepRacer. Źródło: AWS DeepRacer Developer Guide (*https://oreil.ly/rN3dR*)*

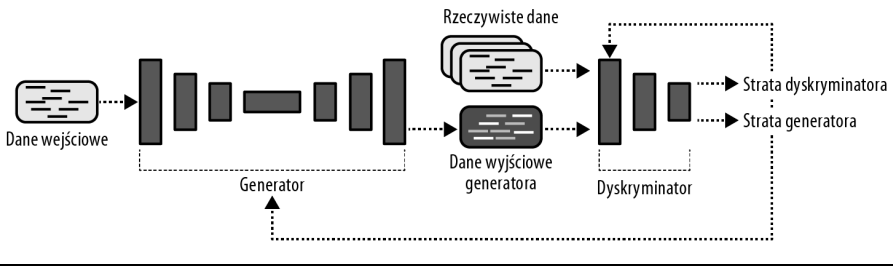
AWS prowadzi też globalną ligę AWS DeepRacer i ranking wyników w oficjalnych wyścigach, które odbywają się zarówno na fizycznych, jak i na wirtualnych torach w ciągu roku.

Poznaj sieci GAN za pomocą urządzenia AWS DeepComposer

To prawda, wszyscy wyglądali na nieco zaskoczonych, gdy firma AWS zaprezentowała urządzenie AWS DeepComposer na corocznej konferencji AWS re:Invent w grudniu 2019 roku. Jednak już po niedługim czasie zaczęliśmy słyszeć charakterystyczne dźwięki dochodzące z holi z różnych hoteli w Las Vegas. AWS DeepComposer to keyboard ze złączem USB, który pomaga w nauce generatywnej SI. Jest zaprojektowany do współpracy z usługą AWS DeepComposer, która przekształca proste melodie w oryginalne utwory. Omawiane urządzenie jest pokazane na rysunku 2.14.

*Rysunek 2.14. AWS DeepComposer to keyboard ze złączem USB, który pomaga w nauce generatywnej SI. Źródło: AWS (*https://oreil.ly/qk6zr*)*

Generatywna SI (przede wszystkim w postaci sieci GAN) jest używana do generowania nowych treści na podstawie przekazanych danych wejściowych. Tymi danymi mogą być obrazy, tekst, a także — naprawdę — muzyka. Modele generatywnej SI automatycznie wykrywają wzorce w danych i uczą się ich oraz wykorzystują tę wiedzę do generowania na tej podstawie nowych danych. W sieciach GAN w procesie generowania nowych treści używane są dwa współzawodniczące algorytmy, generator i dyskryminator (zobacz rysunek 2.15).



Rysunek 2.15. W sieciach GAN używane są algorytmy generatora i dyskryminatora

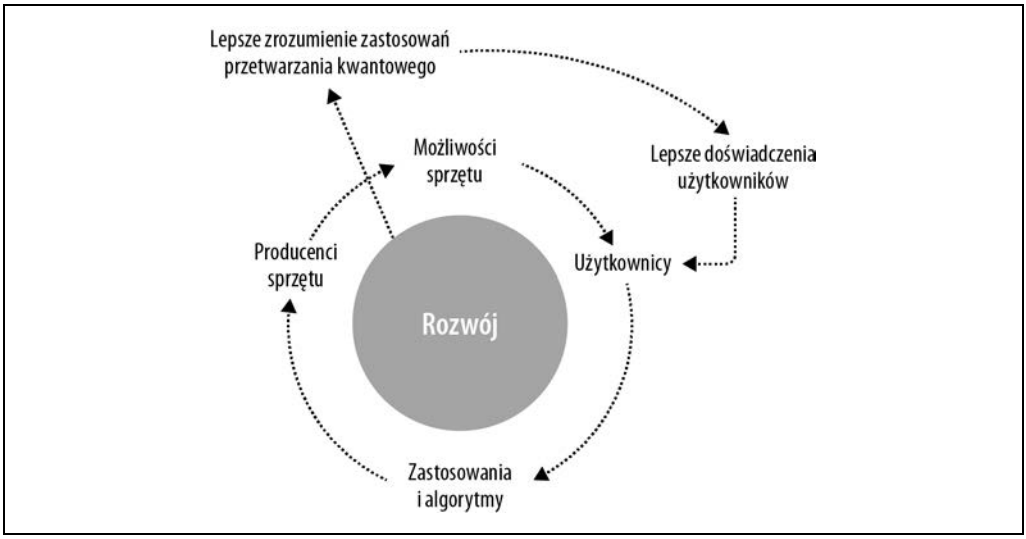
Generator to konwolucyjna sieć neuronowa, która uczy się tworzyć nowe treści na podstawie wzorców z danych wejściowych. Dyskryminator to inna konwolucyjna sieć neuronowa, uczona odróżniać rzeczywiste treści od wygenerowanych. Generator i dyskryminator są uczone naprzemiennie. Celem generatora jest tworzenie coraz bardziej realistycznych treści, natomiast dyskryminator uczy się coraz lepiej odróżniać treści syntetyczne od rzeczywistych.

Wróćmy do przykładu z muzyką. Gdy grasz melodię na keyboardzie, AWS DeepComposer może dodać nawet trzy dodatkowe ścieżki akompaniamentu, aby uzyskać nową kompozycję. Sieć generatora została zaadaptowana ze znanej architektury U-NET używanej do rozpoznawania obrazów, a do nauki wykorzystano publicznie dostępny zbiór danych z kompozycjami Bacha.

Zaprogramuj naturalny system operacyjny za pomocą przetwarzania kwantowego

Budowanie przydatnych aplikacji kwantowych wymaga nowych umiejętności i radykalnie odmiennego podejścia do rozwiązywania problemów. Zdobycie potrzebnej wiedzy wymaga czasu, a także dostępu do technologii kwantowych i narzędzi programistycznych z tego obszaru.

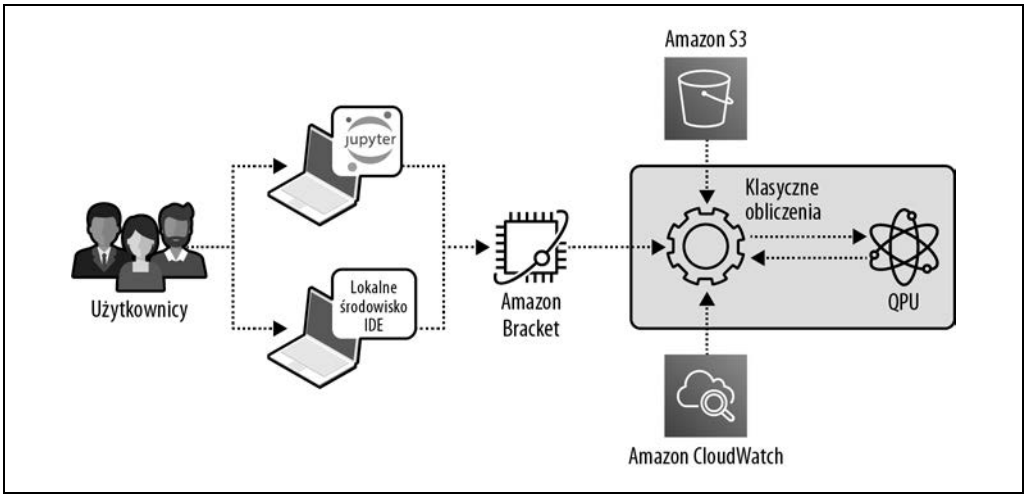
Amazon Braket pomaga zapoznać się z potencjałem sprzętu kwantowego, zrozumieć algorytmy kwantowe i dostosować zestaw narzędzi do kwantowej przyszłości. Na rysunku 2.16 pokazane jest sprzężenie zwrotne związane z rozrastaniem się ekosystemu przetwarzania kwantowego dzięki lepszemu sprzętowi, a także większej liczbie programistów i zastosowań.



Rysunek 2.16. Usługa Amazon Braket jest kołem zamachowym rozwoju przetwarzania kwantowego

Jest wiele podobieństw między dzisiejszymi procesorami graficznymi (GPU) a procesorami kwantowymi jutra (QPU). Procesory GPU zrewolucjonizowały SI i UM dzięki wysoce równoległym obliczeniom cyfrowym. Procesory GPU wymagają specjalnych umiejętności, bibliotek (na przykład NVIDIA CUDA) i sprzętu, aby możliwe było wykorzystanie tak wysokiego poziomu równoległości. Ponadto procesory GPU znajdują się „poza" procesorami CPU, które tradycyjnie zarządzają większymi procesami obliczeniowymi. Synchronizacja danych między procesorami CPU i GPU wymaga specjalnego sprzętu i oprogramowania, aby uwzględnić ich niezależność takich układów.

Podobnie procesory QPU wykonują wysoce równoległe obliczenia kwantowe. Są one o wiele rzędów wielkości bardziej równoległe od ich cyfrowych odpowiedników. Także procesory QPU wymagają specjalnych umiejętności, bibliotek i sprzętu. Znajdują się „poza układem" względem procesorów CPU, dlatego niezbędne są specjalny sprzęt i oprogramowanie do synchronizacji operacji (podobnie jak w przypadku procesorów GPU), co ilustruje rysunek 2.17.



Rysunek 2.17. Używanie procesorów QPU razem z klasycznym komputerem cyfrowym

Bity kwantowe a bity cyfrowe

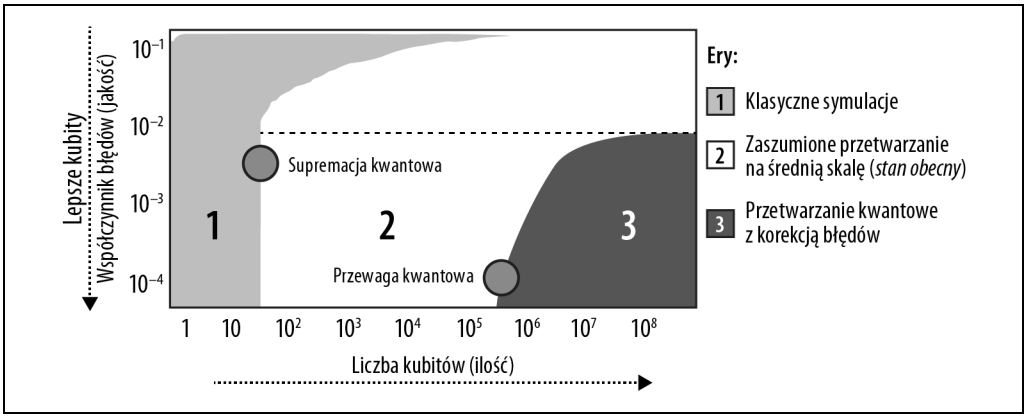
Bity kwantowe (kubity) to odpowiedniki klasycznych bitów cyfrowych używane w obliczeniach kwantowych. Jednak ich stan (0 lub 1) jest probabilistyczny, dlatego przed ustaleniem wartości potrzebna jest operacja odczytu. Ten probabilistyczny stan jest nazywany „superpozycją" i jest ważnym aspektem obliczeń kwantowych.

Obecnie dostępne komputery kwantowe mają mniej więcej 70 – 100 kubitów. Jednak duża ich część jest potrzebna do korekcji błędów z powodu „pełnego szumu" środowiska sprzętu kwantowego. Na przykład w kryptografii potrzebnych jest prawie 6000 czystych kubitów, aby złamać 2048-bitowy szyfr RSA. Taka liczba kubitów wymaga mniej więcej miliona redundantnych kubitów do korekcji błędów, aby uwzględnić pełne szumu środowisko typowe dla obecnie dostępnego sprzętu kwantowego.

Supremacja kwantowa i ery przetwarzania kwantowego

Do niedawna znajdowaliśmy się na etapie klasycznych symulacji, na którym możliwe było symulowanie poprawy wydajności komputerów kwantowych. Jednak w 2019 roku osiągnęliśmy etap supremacji kwantowej, ponieważ z powodu ograniczeń dzisiejszych komputerów cyfrowych niemożliwe stało się symulowanie i mierzenie dodatkowego wzrostu wydajności uzyskiwanego dzięki komputerom kwantowym.

Obecnie mamy erę *zaszumionego przetwarzania kwantowego na średnią skalę* (ang. *Noisy Intermediate-Scale Quantum*). W tej erze specjaliści starają się korygować szum generowany przez środowisko przetwarzania kwantowego, co wymaga ściśle określonej temperatury i odpowiedniego ciśnienia. Podobnie jak koryguje się błędy w rejestrach i układach RAM, konieczne będzie korygowanie błędów w kubitach i układach QRAM, aby możliwe stało się przejście do następnej ery *komputerów kwantowych z korekcją błędów* (zobacz rysunek 2.18).



Rysunek 2.18. Ery przetwarzania kwantowego

Łamanie szyfrów

Szacuje się, że komputerom kwantowym brakuje tylko mniej więcej 10 lat, aby możliwe stało się łamanie współczesnych szyfrów RSA. Obecnie kryptografia jest skuteczna, ponieważ nie mamy wystarczającej mocy obliczeniowej, aby przeprowadzić faktoryzację niezbędną do łamania haseł.

Szacuje się jednak, że 6000 „doskonałych" kubitów (niewymagających korekcji błędów) pozwoli złamać hasła RSA w zaledwie kilka minut. Jest to powód do obaw, dlatego powstała kryptografia „uwzględniająca przetwarzanie kwantowe" lub „postkwantowa", na przykład otwarta implementacja s2n protokołu TLS opracowana przez Amazon (*https://oreil.ly/o3U7G*), w której stosowana jest kryptografia postkwantowa zamiast klasycznej. Więcej o kryptografii postkwantowej piszemy w rozdziale 12.

Symulacje molekularne i wykrywanie leków

Komputery kwantowe mają wyjątkowe możliwości, jeśli chodzi o przetwarzanie równoległe, i potrafią natywnie operować stanami kwantowo-mechanicznymi. Dlatego mogą pomóc w rozwiązaniu bardzo ważnych problemów takich jak mapowanie struktury elektronowej cząsteczek. Symulacje kwantowe prawdopodobnie doprowadzą do odkrycia nowych materiałów, katalizatorów, leków i nadprzekaźników wysokotemperaturowych.

Logistyka i optymalizacje finansowe

Problem optymalizacji występuje w wielu dziedzinach, w tym w logistyce łańcucha dostaw i usługach finansowych. Próba znalezienia optymalnego rozwiązania w wykładniczo rosnącym zbiorze możliwości może szybko doprowadzić do przeciążenia klasycznego komputera cyfrowego. Komputery kwantowe mogą przekroczyć granice możliwości tradycyjnego sprzętu i przyspieszyć działanie wielu technik optymalizacji, w tym algorytmów programowania liniowego i metody Monte Carlo.

Uczenie maszynowe i sztuczna inteligencja na komputerach kwantowych

Niestety, obecnie zastosowania komputerów kwantowych w UM i SI są dość ograniczone. Pojawiają się pierwsze usprawnienia algorytmów liniowych, na przykład SVM (ang. *support vector machines*) i PCA (ang. *principal component analysis*). Ponadto badania nad przetwarzaniem kwantowym okazały się inspiracją do rozwoju klasycznych algorytmów rekomendacji (*https://oreil.ly/H99mZ*). W przyszłości komputery kwantowe z korekcją błędów prawdopodobnie doprowadzą do powstania całej grupy skalowalnych i wysoce wydajnych kwantowych modeli UM i SI.

Programowanie komputera kwantowego za pomocą usługi Amazon Braket

Usługa Amazon Braket obsługuje notatniki Jupytera i udostępnia SDK Pythona, aby umożliwić programistom interakcje z komputerem kwantowym. Za pomocą tego SDK można asynchroniczne przesyłać zadania do zdalnego procesora kwantowego. W podobny sposób przesyłaliśmy zadania i „wypożyczaliśmy" współdzielony komputer w początkowych latach informatyki. Mechanizm ten przypomina także przenoszenie obliczeń z procesora CPU na procesor GPU. Ważną różnicą jest to, że procesory CPU i GPU korzystają z tych samych klasycznych cyfrowych podstaw, natomiast procesor QPU tego nie robi.

Poniższy kod pokazuje, jak zbudować kwantowy obwód obejmujący wiele kubitów. Ten przykład ilustruje, jak przeprowadzić „kwantową teleportację", w ramach której informacje (a *nie* materia) są transportowane z jednego kubita do innego bez używania klasycznych obwodów cyfrowych lub kabli sieciowych:

```
from braket.aws import AwsDevice
from braket.circuits import Circuit, Gate, Moments
from braket.circuits.instruction import Instruction

device = AwsDevice("arn:aws:braket:::device/qpu/ionq/ionQdevice")

# Alicja i Robert początkowa mają ten sam stan Bella.
circ = Circuit();
circ.h([0]);
circ.cnot(0,1);

# Definiowanie schematu kodowania dla Alicji.
# Definiowanie czterech możliwych komunikatów i odpowiadających im bramek.
message = {
           "00": Circuit().i(0),
           "01": Circuit().x(0),
           "10": Circuit().z(0),
           "11": Circuit().x(0).z(0)
          }

# Alicja wybiera komunikat do przesłania. Niech będzie to '01'.
m = "01"

# Alicja koduje komunikat, stosując zdefiniowane wcześniej bramki.
circ.add_circuit(message[m]);

# Alicja przesyła kubit do Roberta. W efekcie Robert ma oba kubity.
# Robert odkodowuje komunikat Alicji, rozplątując dwa kubity.
```

```
circ.cnot(0,1);
circ.h([0]);

print(circ)

### DANE WYJŚCIOWE ###

T  : |0|1|2|3|4|
q0 : -H-C-X-C-H-
        |   |
q1 : ---X---X---
T  : |0|1|2|3|4|
```

Centrum przetwarzania kwantowego AWS

AWS we współpracy z instytutem Caltech zbudowało centrum przetwarzania kwantowego AWS (*AWS Center for Quantum Computing*), które zostało otwarte w 2021 roku. Jednostka ta pracuje nad przydatnymi zastosowaniami przetwarzania kwantowego, kubitami z korekcją błędów, modelami programowania kwantowego i nowym sprzętem kwantowym.

Wzrost wydajności i obniżenie kosztów

Co by się stało, gdyby można było podwoić szybkość działania kodu i zmniejszyć wielkość puli serwerów o połowę? Oznaczałoby to znaczną oszczędność pieniędzy. Jaki byłby efekt automatycznego wykrywania problemów operacyjnych w aplikacjach i wyświetlania rekomendacji poprawek zwiększających dostępność aplikacji? Zmniejszenie czasu przestojów to następny obszar pozwalający na znacznie oszczędności.

W tym podrozdziale przedstawiamy w pełni zarządzane usługi Amazon CodeGuru Reviewer, Amazon CodeGuru Profiler i Amazon DevOps Guru. CodeGuru Reviewer i Profiler pomagają poprawić wydajność kodu i zmniejszyć ilość potrzebnych zasobów, natomiast Amazon DevOps Guru pomaga wykrywać problemy operacyjne i zwiększać dostępność aplikacji.

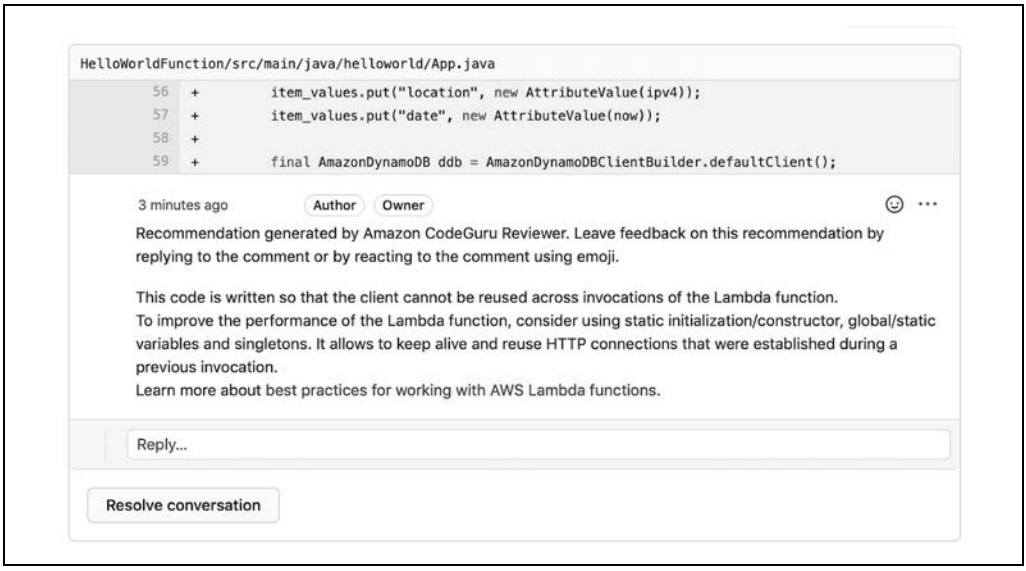
Automatyczna inspekcja kodu za pomocą usługi CodeGuru Reviewer

Inspekcja kodu to znana zalecana praktyka z obszaru rozwoju oprogramowania. Polega ona na tym, że kod jest sprawdzany przez doświadczonych członków zespołu, którzy oceniają wydajność, jakość i bezpieczeństwo. Oprócz wiedzy z danej dziedziny osoby te posiadają też wiedzę o idiomach programistycznych stosowanych w zespole, a także są wyczuleni na „zapachy kodu".

Jednak czasem nawet najbardziej doświadczony członek zespołu może nie zauważyć subtelnych wąskich gardeł, które obniżają wydajność, lub błędnej obsługi wyjątków. Inspektorzy kodu często skupiają się na problemach specyficznych dla dziedziny, na przykład na niskiej jakości implementacji modelu dziedziny lub błędnie skonfigurowanych mechanizmach integracji w usłudze. Ponadto takie osoby często mają dostęp tylko do statycznego kodu, a nie do generowanych na żywo wskaźników ze środowiska uruchomieniowego. Usługa CodeGuru obejmuje usługę CodeGuru Reviewer, która automatycznie wykonuje inspekcję kodu, i CodeGuru Profiler, służącą do monitorowania wydajności kodu.

CodeGuru Reviewer automatyzuje proces inspekcji kodu i wyświetla sugestie oparte na modelach uczenia maszynowego wyuczonych na podstawie milionów wierszy kodu z setek tysięcy fragmentów wewnętrznego kodu bazowego Amazona, a także ponad 10 000 otwartych projektów z serwisu GitHub.

Wystarczy w bezpieczny sposób i z zachowaniem prywatności wskazać usłudze CodeGuru repozytorium z kodem źródłowym, a zacznie ona wyświetlać sugestie. CodeGuru analizuje wszystkie prośby o akceptację zmian w repozytorium z kodem źródłowym i automatycznie oznacza krytyczne usterki takie jak wycieki danych uwierzytelniających, wycieki zasobów, sytuacje wyścigu w kodzie współbieżnym i niewydajne wykorzystanie zasobów platformy AWS. Sugeruje też zmiany w odpowiednich wierszach kodu, które pozwolą zaradzić defektom (zobacz rysunek 2.19).



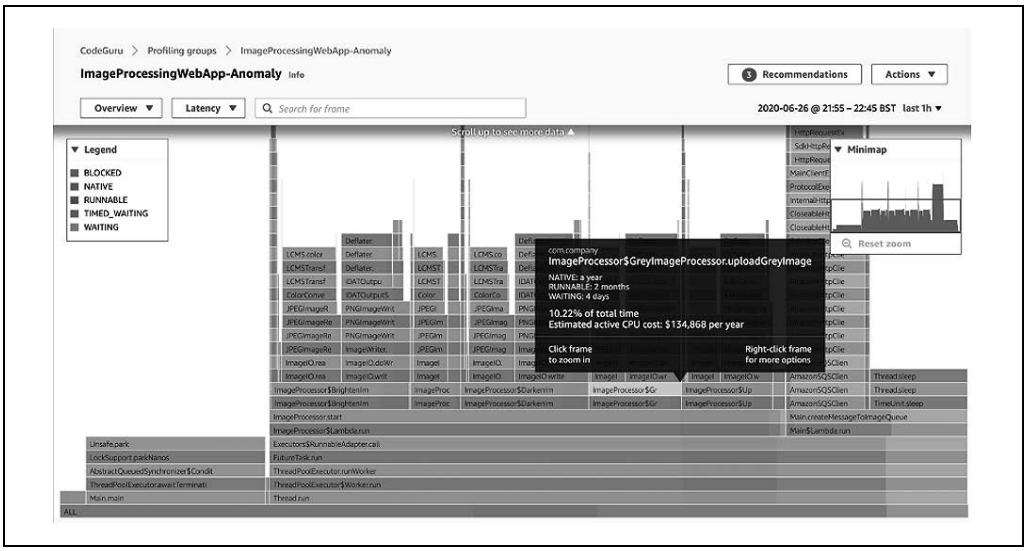
Rysunek 2.19. CodeGuru Reviewer analizuje kod źródłowy i wyświetla sugestie, aby zwiększyć wydajność i ograniczyć koszty

W tym przykładzie oryginalny kod z funkcji Lambda przy każdym wywołaniu tworzył nowego klienta bazy DynamoDB, zamiast utworzyć go raz i zapisać w pamięci podręcznej. Bez tej zmiany system marnuje cykle obliczeniowe i rejestry pamięci, nieustannie tworząc od nowa ten sam obiekt klienta bazy DynamoDB przy każdym wywołaniu. Dzięki wprowadzonej zmianie funkcje Lambda mogą obsłużyć więcej żądań na sekundę, co oznacza zużycie mniejszej ilości zasobów i niższe koszty.

CodeGuru Reviewer sprawdza kod pod kątem zalecanych praktyk z języków Python i Java z uwzględnieniem pul połączeń i obsługi wyjątków. Usługa obejmuje mechanizm Security Detector do wykrywania problemów z bezpieczeństwem takich jak przekazywanie niesprawdzonych argumentów w wywołaniach podprocesów Pythona z poziomu systemu operacyjnego. CodeGuru Reviewer wykrywa też „zapachy kodu", zmniejsza dług techniczny i ułatwia konserwację kodu bazowego.

Zwiększanie wydajności aplikacji za pomocą usługi CodeGuru Profiler

CodeGuru Profiler potrafi wykrywać wąskie gardła w kodzie w czasie wykonywania aplikacji, analizując jej profil wykonania, oznaczając najbardziej kosztowne obliczeniowo wiersze kodu i udostępniając inteligentne rekomendacje. Profiler generuje wizualizacje takie jak wykres typu flame z rysunku 2.20, aby pokazać, na jakich fragmentach należy się skupić w celu zoptymalizowania wydajności i zaoszczędzenia jak największej ilości pieniędzy.



Rysunek 2.20. Wykres typu flame wygenerowany przez usługę CodeGuru Profiler w celu wskazania w kodzie wąskich gardeł zmniejszających wydajność

Wykres typu flame przedstawia stos wywołań w czytelnej dla człowieka postaci z dokładnymi nazwami funkcji. W trakcie analizowania wykresów typu flame należy skupić się na widocznych wypłaszczeniach. Często oznaczają one, że zasób jest zablokowany w oczekiwaniu na sieć lub dyskowe operacje wejścia-wyjścia. W idealnym scenariuszu na wykresie typu flame widocznych jest wiele wąskich szczytów i niewielka liczba wypłaszczeń.

Zwiększanie dostępności aplikacji za pomocą usługi DevOps Guru

Amazon DevOps Guru to oparta na UM usługa operacyjna, która automatycznie wykrywa problemy operacyjne w aplikacji i rekomenduje poprawki. DevOps Guru sprawdza wskaźniki aplikacji, dzienniki i zdarzenia, aby zidentyfikować odbiegające od standardowych wzorców sytuacje takie jak większe opóźnienie odpowiedzi, wyższy współczynnik błędów i nadmierne zużycie zasobów. Po wykryciu takiego wzorca DevOps Guru przesyła alert wraz z podsumowaniem anomalii, potencjalnymi ich przyczynami i możliwym rozwiązaniem.

Podsumowanie

W tym rozdziale pokazaliśmy wiele różnych sytuacji, w których można zastosować różne gotowe usługi SI i UM z platformy AWS, wymagające niewielkiej lub nawet zerowej ilości kodu. Niezależnie od tego, czy jesteś programistą aplikacji i nie masz bogatej wiedzy na temat uczenia maszynowego, czy jesteś doświadczonym danologiem chcącym skupić się na trudnych problemach z obszaru uczenia maszynowego, warto zapoznać się z zarządzanymi usługami SI i UM oferowanymi przez Amazon.

Można łatwo wzbogacić aplikacje o gotowe do użycia usługi SI i to zarówno wtedy, gdy Twoja firma wymaga przeniesienia uczenia maszynowego do urządzeń brzegowych, jak i wtedy, gdy dopiero rozpoczynasz przygodę z SI i UM oraz szukasz ciekawych i pouczających sposobów na zastosowanie rozpoznawania obrazów, uczenia przez wzmacnianie lub sieci GAN.

Przedstawiliśmy kilka przykładów ilustrujących, jak zastosować wysokopoziomowe usługi SI, w tym Amazon Personalize do generowania rekomendacji i Forecast do prognozowania zapotrzebowania na zasoby.

Pokazaliśmy, że uczenie maszynowe jest stosowane w wielu istniejących usługach z platformy AWS, w tym do predyktywnego automatycznego skalowania i tworzenia puli wstępnie zainicjowanych instancji w usłudze Amazon EC2. Wyjaśniliśmy, jak wykrywać wyciek wrażliwych danych i zapobiegać temu problemowi z użyciem usługi Macie, a także jak chronić się przed oszustwami za pomocą usługi Amazon Fraud Detector. Omówiliśmy ulepszanie doświadczeń klienta za pomocą narzędzi Amazon Contact Lens for Amazon Connect, Comprehend, Kendra i Lex. Ponadto wyjaśniliśmy, jak automatyzować inspekcje kodu źródłowego oraz identyfikować możliwe poprawki wydajności i optymalizacje kosztów za pomocą usług CodeGuru Reviewer, CodeGuru Profiler i DevOps Guru.

W rozdziale 3. omawiamy zautomatyzowane uczenie maszynowe. Pokazujemy, jak zbudować modele predykcyjne za pomocą kilku kliknięć w narzędziach Amazon SageMaker Autopilot and Amazon Comprehend.

AWS i inżynieria danych: tak zwiększysz wydajność i obniżysz koszty!

Platforma Amazon Web Services jest uważana za największą i najbardziej dojrzałą chmurę obliczeniową. Zapewnia bogaty zestaw specjalistycznych narzędzi ułatwiających realizację projektów z zakresu inżynierii danych i uczenia maszynowego. W ten sposób inżynierowie danych, architekci i menedżerowie mogą szybko zacząć używać danych do podejmowania kluczowych decyzji biznesowych. Uzyskanie optymalnej efektywności pracy takich projektów wymaga jednak dobrego rozeznania w możliwościach poszczególnych narzędzi, usług i bibliotek.

Dzięki temu praktycznemu przewodnikowi szybko nauczysz się tworzyć i uruchamiać procesy w chmurze, a następnie integrować wyniki z aplikacjami. Zapoznasz się ze scenariuszami stosowania technik sztucznej inteligencji: przetwarzania języka naturalnego, rozpoznawania obrazów, wykrywania oszustw, wyszukiwania kognitywnego czy wykrywania anomalii w czasie rzeczywistym. Ponadto dowiesz się, jak łączyć cykle rozwoju modeli z pobieraniem i analizą danych w powtarzalnych potokach MLOps. W książce znajdziesz też zbiór technik zabezpieczania projektów i procesów z obszaru inżynierii danych, takich jak stosowanie usługi IAM, uwierzytelnianie, autoryzacja, izolacja sieci, szyfrowanie danych w spoczynku czy postkwantowe szyfrowanie sieci dla danych w tranzycie.

Implementowanie solidnego kompletnego procesu uczenia maszynowego to żmudne zadanie, dodatkowo komplikowane przez szeroki zakres dostępnych narzędzi i technologii. Autorzy wykonali świetną robotę, a jej efekty pomogą zarówno nowicjuszom, jak i doświadczonym praktykom realizować to zadanie z wykorzystaniem możliwości, jakie dają usługi AWS

— **Brent Rabowsky**
danolog w firmie Amazon Web Services

W książce między innymi:

- narzędzia AWS związane ze sztuczną inteligencją i z uczeniem maszynowym
- kompletny cykl rozwoju modelu przetwarzania języka naturalnego
- powtarzalne potoki MLOps
- uczenie maszynowe w czasie rzeczywistym
- wykrywanie anomalii i analiza strumieni danych
- zabezpieczanie projektów i procesów z obszaru inżynierii danych

Chris Fregly jest głównym ambasadorem deweloperów w obszarach sztucznej inteligencji i uczenia maszynowego w AWS. Regularnie występuje na konferencjach poświęconych SI i UM na całym świecie.

Antje Barth jest starszą ambasadorką deweloperów w obszarach sztucznej inteligencji i uczenia maszynowego w AWS. Jest też współzałożycielką düsseldorfskiego oddziału organizacji Women in Big Data.

Helion
helion.pl
HELION SA
ul. Kościuszki 1c
44-100 Gliwice
tel.: 32 230 98 63
helion@helion.pl
KOD KORZYŚCI
Sięgnij po więcej!
ISBN 978-83-283-9128-4
9 788328 391284
Cena: 129,00 zł