ETAPY ANALIZY DANYCH PRODUKCYJNYCH W CHMURZE OBLICZENIOWEJ Z WYKORZYSTANIEM JĘZYKA R

Jakub PIZOŃ, Jerzy LIPSKI, Tomasz CIEPLAK

Streszczenie: Niniejszy artykuł stanowi opis poszczególnych etapów analizy danych produkcyjnych w celu stworzenia modelu prognostycznego przydatnego w ocenie i przewidywaniu awarii z powodu niesprawności pewnych elementów maszyny. Artykuł opisuje ogólny schemat postępowania składający się z czterech etapów. Modele predykcyjne budowane są za pomocą bibliotek język R oraz środowiska maszynowego uczenia w chmurze obliczeniowej.

Słowa kluczowe: data science, utrzymanie ruchu, język R

1. Wstęp

Kluczowym zastosowaniem pakietów statystycznych jest obróbka, przygotowanie, analiza i wizualizacja danych. Oprogramowanie wspomagające te problemy jest najczęściej budowane celem obróbki wyników przeprowadzonych badań i tym samym potwierdzeniem bądź zaprzeczeniem stawianych tez. Kluczowym dla użytkowania pakietów statystycznych jest wydajny sprzęt, który posiada odpowiednie parametry (rozmiar pamięci RAM, wydajny procesor) i dzięki temu zapewnia odpowiednią moc obliczeniową do przeprowadzenia obliczeń.

Z drugiej strony, niezwykle istotne jest to by efekty pracy pakietu statystycznego miały – po za teoretycznym – praktyczny wymiar. Tym samym, by forma i kształt wytworzonego rozwiązania – modelu analitycznego – pozwalała na jego dystrybucję i wykorzystanie w ramach dowolnych architektur rozwiązań systemów informatycznych. Dopiero rozwiązanie, które realizuje te wymagania może zostać uznane za w pełni funkcjonalne i przydatne.

Pojawia się pytanie jak tworzyć tego typu rozwiązania? Odpowiedź na to pytanie nie jest trywialna przede wszystkim ze względu na wielowymiarowość i wymaganą wysoką specjalizację poszczególnych problemów, a tym samym im adekwatnych rozwiązań. Z tego powodu, wymagane jest by proces tworzenia rozwiązania przebiegał w ściśle określony sposób np. w formie analizy wielokryterialnej, pozwalającej na wskazanie odpowiedniego do założonych kryteriów (specyfika problemu, wymagania klienta) rozwiązania.

W celu przeprowadzenia analizy konieczne jest zgromadzenie zestawu wymagań, określenie precyzyjnych kwantyfikowalnych kryteriów jak i pozyskanie wiedzy dziedzinowej w zakresie rozpatrywanego problemu (zarządzanie produkcją, inżynieria produkcji). Ponadto konieczne jest spełnienie wymagań brzegowych tożsamych z dysponowaniem wiedzą na temat dostępnych narzędzi analizy danych czy też inżynierii oprogramowania. Niemniej jednak ponad zbiorem wymagań niezbędny jest też sposób w jaki zbiór deklaratywnych stwierdzeń zostanie przełożony na istotny model dostarczający skutecznych prognoz czy też wyników. Kluczowe jest to, by ten model był też dostępny w wygodny dla aplikacji sposób. Tym samym, środowisko wykorzystane do sporządzenia

modelu powinno mieć możliwość zarówno wygenerowania modelu jak i przygotowania go do formy wykorzystania.

W związku z rozwojem wielu gałęzi przemysłu jak i z rosnącymi możliwościami wykorzystania metod sztucznej inteligencji zauważalny jest wzrost rozwiązań, które przy wykorzystaniu modelu usługowego mocy obliczeniowej udostępniają narzędzia modelowania i generowania modeli analitycznych.

Równoważnie to tego trendu coraz bardziej popularna staje się nauka eksploracji i przetwarzania danych określana mianem data science. Data Science (nauka o danych) stanowi interdyscyplinarną dziedzinę związaną z procesem naukowego i systematycznego wyodrębniania wiedzy (zależności, relacji) z różnej postaci danych występującej w formie ustrukturyzowanej jak i nie ustrukturyzowanej [1].

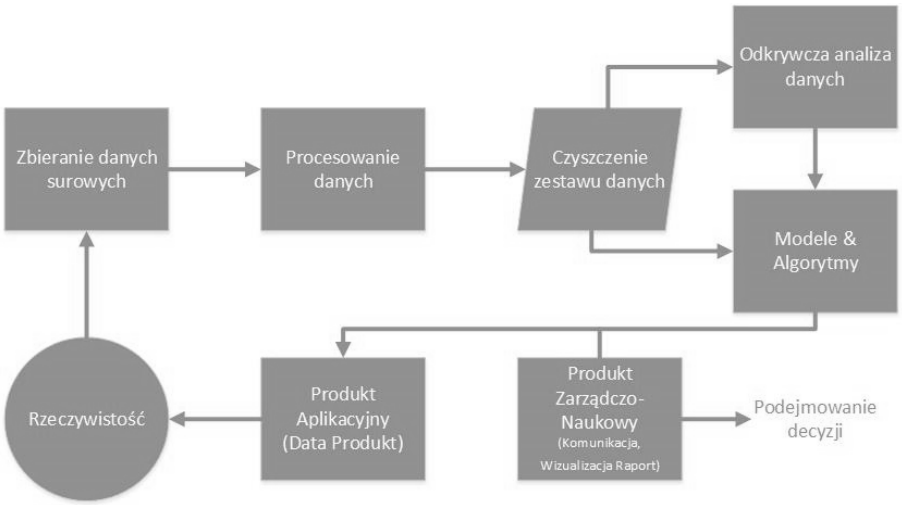
Data science wykorzystuje techniki i teorie zaczerpnięte z wielu dziedzin w ramach szerokich dziedzin matematyki, statystyki, badań operacyjnych, informacji naukowej i informatyki, w tym teorii sygnałów, uczenia maszynowego, eksploracji danych, baz danych, inżynierii danych, rozpoznawania wzorców i uczenia się, wizualizacji, predykcji, programowania, sztucznej inteligencji i obliczeń o wysokiej wydajności.

Kluczowy dla data science jest proces (rys 1.) w którym generowane są mierzalne efekty pracy nad danymi. Proces ten koncentruje się na tym by zbierać dane ze świata rzeczywistego. Następnie procesować je do formy zapewniającej możliwość przetwarzania (najczęściej uporządkowana forma danych np.: tablica). Następnie konieczne jest czyszczenie danych z elementów zaburzających ich postać (elementy odstające, braki danych).

W zależności od charakteru prowadzonych badań po etapie czyszczenia przeprowadzany jest etap analizy odkrywczej bądź etap formułowania modelu i algorytmów. Wystąpienie jednego bądź drugiego etapu zależy od tego czy celem przeprowadzanej analizy jest wyszukanie nowych zależności (hipotez) na podstawie uzyskanych danych czy też sprawdzenie gotowych hipotez pod względem uzyskanych danych. W przypadku pierwszym przypadku wykorzystywana jest eksploracyjna analiza danych (ang. Exploaratory Data Analysis), która prowadzi do zaproponowania modelu analitycznego lub wskazania odkrywczych zależności na podstawie technik umożliwiających przeglądanie i wizualizacje danych. Na podstawie przeglądu formułowane są hipotezy, które w następnym etapie są weryfikowane i prezentowane w formie modeli analitycznych czy też algorytmów. W przypadku, gdy forma danych uniemożliwia ich eksplorację, dane mogą zostać ponownie skierowane do etapu ich translacji.

Bez względu na charakter prowadzonych badań proces data science prowadzi do etapu weryfikacji hipotez i przedstawienia ich w formie modeli jak i algorytmów. W zależności od proponowanych pytań, odpowiednio produkty analizy są testowane do momentu gdy parametry wskazujące o ich istotności potwierdzają ich przydatność. Uzyskane produkty procesu mogą mieć wymiar albo aplikacyjny albo naukowy [2].

Produkty aplikacyjne (gotowe programy, biblioteki statystyczne) mogą posłużyć do przetwarzania danych w realnym świecie w ramach systemów informatycznych bądź innych większych struktur. Natomiast produkty zarządczo-naukowe mają charakter informatywny. Charakter ten wyraża się poprzez to, że fakty mające źródło w danych są komunikowane, wizualizowane w formie raportów czy innych analiz naukowych. Tak przygotowane źródło informacji stanowi podstawę do podejmowania decyzji zarządczych czy też optymalizacyjnych. Z drugiej strony, może też stanowić dowód prowadzonych badań czy też potwierdzać przydatność proponowanych odkryć naukowych.



Rys. 1. Proces nauki o danych [2]

Należy zwrócić uwagę na fakt, że proces data science rozwiązania danego problemu stanowi unikalny zestaw założeń, metod i interpretacji. Tym samym za każdym razem sposób uzyskania wizualnego bądź aplikacyjnego rozwiązania stanowi nowy sposób prowadzenia analizy danych. Przy czym jeden sposób postępowania może się w efekcie okazać lepszy i być bardziej efektywny niż inny. W wyniku czego może to pozwolić na szybszą i efektywniejszą analizę, która zaowocuje lepszymi wynikami czy też zwolnieniem zasobów, które mogą zostać dedykowane innym zadaniom.

1.1. Analiza danych produkcyjnych w chmurze

Konstrukcja rozwiązania aplikacyjnego bądź zarządczo-naukowego wymaga zaangażowania zasobów ludzkich jak i sprzętowych. W celu prowadzenia profesjonalnych i produkcyjnych badań data science niezbędne jest pozyskanie danych oraz infrastruktura techniczna jak i specjalistyczne oprogramowanie.

Dlatego też stworzenie takiego środowiska stanowi znaczący koszt i wymaga czasu. Oczywiście istnieje możliwość wytworzenia prototypowych rozwiązań w oparciu o mniej zaawansowaną infrastrukturę ale dostarczone w ten sposób rozwiązanie może okazać się wadliwe, nie w pełni funkcjonalne bądź nie adekwatne do badanych zagadnień.

Z tego powodu alternatywnym rozwiązaniem jest środowisko data science dostępne w modelu usługowym z poziomu chmury obliczeniowej. Kluczowe dla takiego środowiska jest dynamiczne skalowanie zasobów jak i dostępność środowiska przetwarzania danych pozwalającego na rozbudowanie go o dodatkowe biblioteki – odpowiednie dla klasy i ścieżki prowadzonego badania.

Na potrzeby niniejszego artykułu wykorzystane zostało rozwiązanie proponowane przez firmę Microsoft Azure Machine Learning [3]. Rozwiązanie to stanowi platformę dla naukowców danych pozwalające na łatwe tworzenie i wdrażanie rozwiązań uczenia maszynowego typu end-to-end od surowych danych do konsumowalnego web-service'u (e-usługi online). Platforma zawiera gotowy zestaw narzędzi, które wspierają każdy etap procesu data science i zapewniają integralność przetwarzanych danych.

2. Zagadnienie utrzymania stanu dla procesu produkcyjnego

Proces produkcyjny stanowi kompozycję wielu różnych składników, które współdziałając przyczyniają się do realizacji założonych celów, najczęściej tożsamych z wytwarzaniem dóbr. Bezpośrednio z procesem produkcyjnym wiążą się przestoje, które (poza przyczynami wynikającymi z technologii) generowane są przez awarie [3]. Awarie te powodują kosztowne i niezwykle kłopotliwe zakłócenia procesów produkcyjnych wpływając na ich parametry ekonomiczne. Przewidywanie awarii pozwala na podejmowanie działań zapobiegawczych, po to by unikać awarii i minimalizować ich konsekwencje. Działania takie zwane utrzymaniem według stanu (ang. predictive maintenance) polegają na monitorowaniu stanu maszyn i wyznaczaniu na bieżąco prawdopodobieństwa wystąpienia problemów.

Dzięki wdrożeniu strategii utrzymania, możliwe jest zmniejszenie kosztów awarii oraz obsługi serwisowej i czasu przestojów. Przy czym, jednocześnie wydłużając czas sprawnego działania urządzeń, a tym samym zwiększając bezpieczeństwo.

Utrzymanie według stanu obejmuje różne zagadnienia takie jak: przewidywanie awarii, diagnostyka awarii, wykrywanie awarii, klasyfikacja rodzaju awarii i rekomendacji działań łagodzących skutki lub konserwacji po awarii.

W ramach Azure Machine Learning, firma Microsoft udostępnia opisany sposób, który pomaga naukowcom danych by łatwo zbudować i wdrożyć rozwiązanie predykcyjnej konserwacji. Proponowany sposób postępowania, w ramach procesu data science, skupia się na technikach stosowanych w celu przewidywania, kiedy maszyna w trakcie eksploatacji ulegnie awarii. Pozwoli to z odpowiednim wyprzedzeniem zaplanować jej konserwację. Rozwiązanie zawiera zbiór wstępnie skonfigurowanych modułów uczenia maszynowego, a także skrypty R, które umożliwiają to konstrukcję systemu od przetwarzania danych do rozwiązania aplikacyjnego – modelu maszynowego uczenia się.

Sposób postępowania może zostać dostosowany do różnych scenariuszy prognostycznych konserwacji. Wymagane jest jednak by dostępne były dane pozyskane zarówno w czasie pracy maszyny jak i w warunkach awaryjnych dla danych jej zespołów, przy czym konieczne jest by prawdopodobieństwo awarii zespołu było związane z eksploatacyjnym starzeniem się [3].

Od strony technicznej sposób postępowania bazuje na języku skryptowym R. R to interpretowany język programowania oraz środowisko do obliczeń statystycznych i wizualizacji wyników [4]. W tej chwili GNU R rozprowadzany jest w postaci kodu źródłowego oraz w postaci binarnej wraz z wieloma dystrybucjami Linuksa. Dostępna jest także wersja dla Microsoft Windows i Mac OS. R jest wykorzystywany w wielu znanych firmach, (w tym m.in. Facebook, Google, Merck, Altera, Pfizer, LinkedIn, i inni).

2.1. Gromadzenia danych procesu i maszyn

Po to by prowadzić badania i proponować odpowiednie modele predykcyjne potrzebne są dane danej maszyny i badanego procesu. Pozyskanie tych danych stanowi podstawowy warunek kolejnych działań. W niniejszym artykule etapy analizy danych produkcyjnych w chmurze obliczeniowej z wykorzystaniem języka R przedstawione w odniesieniu do zestawu danych telemetrycznych silników [8]. Przy czym należy wskazać, że kluczowe jest dla artykułu by wskazać etapy analizy. To świadczy o tym, że opisywane zagadnienie posiada uniwersalny charakter i może być rozpatrywane w odniesieniu do wielu specyficznych zagadnień technicznych.

W przypadku środowiska produkcyjnego pozyskanie danych w większości zależy od rodzaju parku maszynowego. W przypadku maszyn CNC bądź sterowników PLC możliwe jest pozyskanie danych z wewnętrznej pamięci kontrolera. Niemniej jednak w przypadku informacji na temat ilości produkcji w toku zebranie odpowiednich danych stanowi wyzwanie.

Wraz z pojawieniem się Internetu rzeczy (ang. Internet of Things, IoT) predykcja parametrów związanych z utrzymaniem zyskuje coraz większą aplikacyjność w przemyśle. Przede wszystkim w wymiarze technologii gromadzenia i przetwarzania danych z wykorzystaniem IoT, które uznawane za na tyle dojrzałe by generować, przesyłać, przechowywać i analizować wszelkiego rodzaju dane w partiach lub w czasie rzeczywistym [5].

Problemy biznesowe w zakresie predykcyjnego utrzymania wiążą się zarówno z wysokim ryzykiem operacyjnym z powodu niespodziewanych awarii jak i ograniczonym wglądem w przyczyny problemów w złożonych środowiskach produkcyjnych. Większość z tych problemów można zawrzeć w formie pytań takich jak:

- Jakie jest prawdopodobieństwo, że dana maszyna ulegnie awarii w najbliższej przyszłości?
- Jaki jest pozostały okres użytkowania sprzętu?
- Jakie są przyczyny awarii oraz jakie czynności konserwacyjne należy wykonać, aby rozwiązać problemy?

Zbieranie danych w celu wykorzystania w metodach predykcji, umożliwia:

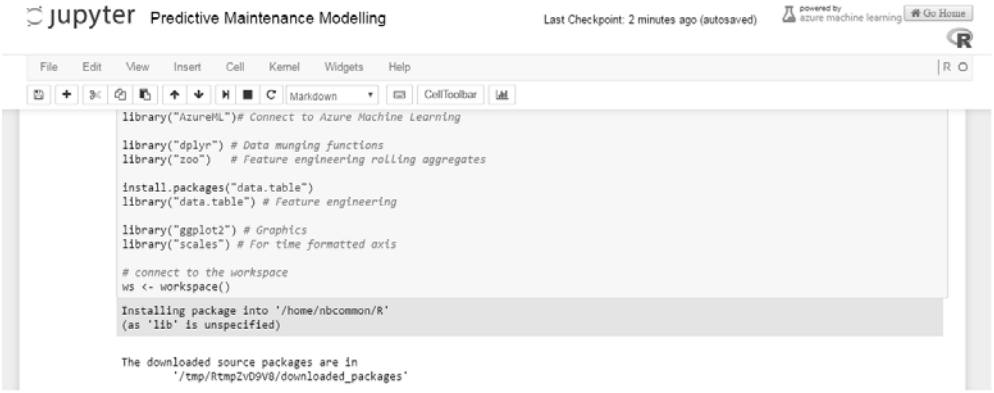
- przez przewidywanie awarii przed ich wystąpieniem możliwe jest zmniejszenie ryzyka operacyjnego,
- zmniejszyć zbędną ilość czynności konserwacyjnych i kontrolnych oraz kosztów konserwacji,
- obniżyć koszty inwentaryzacji poprzez zmniejszenie poziomu zapasów związanych z działaniami naprawy i konserwacji,
- zidentyfikować zależności pomiędzy różnymi problemami utrzymania.

Implementując rozwiązanie analityczne możliwe jest wskazanie kluczowych wskaźników wydajności dla przedsiębiorstwa. Czyli takich, które wskażą jaki jest stanu zdrowia parku maszynowego lub też jaka jest szacunkowa wartość pozostałej żywotności aktywów. W zależności od tych wskaźników możliwe staje się proponowanie działań konserwacyjnych jak i wskazanie szacunkowych dat zamówienia części zamiennych [6].

2.2. Model analizy zgromadzonych danych

Analiza danych produkcyjnych przy wykorzystaniu środowiska maszynowego uczenia się w chmurze pozwala rozwiązać zagadnienie wpisujące się w powyżej opisane problemy biznesowe, a dokładniej odpowiedź na pytanie: "Jakie jest prawdopodobieństwo, że urządzenie ulegnie awarii z powodu uszkodzenia jednego z jego komponentów?". Odpowiedź na tak sformułowane pytanie może zostać uzyskana przy pomocy klasyfikacji wielokryterialnej oraz algorytmu uczenia w celu stworzenia modelu prognostycznego Model uczy się na podstawie danych historycznych zebranych z maszyn.

Techniczna strona obliczeń realizowana jest za pomocą repozytoriów języka R uruchamianych z poziomu notatnika Jupyter działającego w środowisku Azure Machine Learning Studio.



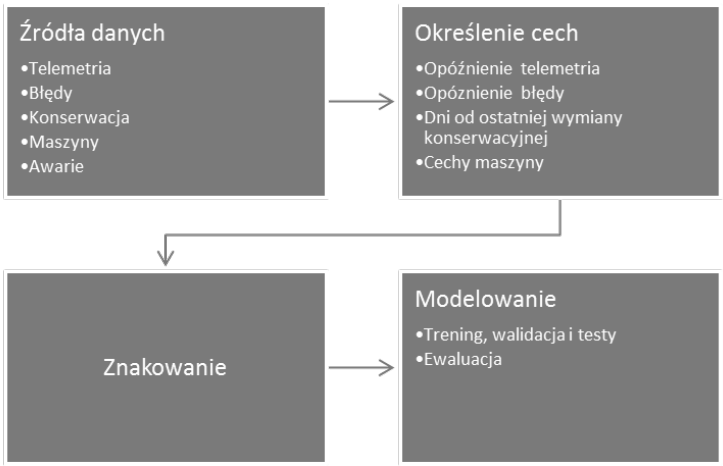
Rys. 2. Notatnik Jupyter w środowisku Azure Machine Learning Studio https://studio.azureml.net/

2.3. Tworzenie modelu analizy danych

Proces tworzenie modelu ma charakter iteratywny, gdzie to czy zostanie wykonana kolejna iteracja zależy od jej wyników. Adekwatnie do uzyskanych wyników wybierane jest krok w którym ponownie rozpoczyna się analiza. Jednak bez względu na to jakie krok zostanie wybrany proces tworzenia modelu pozostaje sekwencyjny – dopiero wyniki decydują o kolejnej iteracji.

Pierwszym etapem tworzenia modelu jest pozyskanie źródeł danych. Typowe źródła danych dla predykcji stanów utrzymania to:

- historia awarii urządzenia lub elementu w maszynie,
- historia naprawy maszyny, (np. kody błędów, poprzednie prace konserwacyjne lub wymiana elementów),
- warunki eksploatacji obrabiarki zarejestrowane za pomocą czujników,
- cechy maszyny, na przykład moc silnika, marka i model, lokalizacja,
- cechy operatora np. płeć, doświadczenie z przeszłości.



Rys. 3. Etapy tworzenia modelu [3]

Dane dla opisywanego przykładu pochodzą z 4 różnych źródeł i stanowią telemetryczne dane rzeczywiste maszyn (silników) zawierające komunikaty o błędach, zapisy o konserwacji zawierające opis awarii oraz informacje na temat maszyny takie jak typ i wiek.

Pierwszym źródłem danych są dane telemetryczne w formie szeregów czasowych pomiarów napięcia, obrotów, ciśnienia i drgań zebranych z 100 maszyn w czasie rzeczywistym uśredniane co godzinę i pobrane w trakcie produkcji w roku 2015.

Tab. 1 Wycinek danych telemetrycznych silników

	data_czas	Maszyna_ID	Napięcie	Obrót	Ciśnienie	Wibracja
876091	2015-12-31 21:00:00	100	153.4046	552.0126	99.13295	37.66681
876092	2015-12-31 22:00:00	100	169.3973	546.1446	107.60703	38.69560
876093	2015-12-31 23:00:00	100	152.1820	414.0116	117.42634	43.96671
876094	2016-01-01 00:00:00	100	168.7400	439.4841	103.15823	47.28900

Drugim ważnym źródłem danych są dzienniki zidentyfikowanych błędów komponentów silnika. Są to błędy nie powodujące przestojów - urządzenie nadal działa i błąd nie ma charakteru awarii. Data i czas błędu są zaokrąglane do najbliższej godziny, ponieważ dane telemetryczne są zbierane w układzie godzinnym. Trzecie źródło, stanowią zaplanowane i niezaplanowane zapisy konserwacji, które mają charakter zarówno regularnych inspekcji komponentów jak i tych związanych z awariami. Rekord w bazie jest generowany, gdy komponent jest wymieniany ze względu na zaplanowaną kontrolę bądź z powodu uszkodzenia. Dane z konserwacji pochodzą zarówno z 2014 i 2015 roku. Czwarte źródło zawiera informacje o maszynach, zawierają typ modelu i wiek aktualizowany co roku w serwisie. Natomiast ostatni zestaw danych to awarie. Zestaw ten to zapis wymiany komponentów z powodu awarii. Każdy rekord zawiera datę i czas, nr identyfikacyjny ID urządzenia i typ komponentu.

Następnym etapem określenia źródeł danych jest wskazanie cech pożądanych dla utrzymania według stanu. Polega to na zestawieniu różnych źródeł danych tak by jak najlepiej opisać cechy obrazujące stan sprawności danej maszyny w określonym momencie. W proponowanym przykładzie są to:

Cecha I: *Opóźnienie telemetria.* - Dane telemetryczne prawie zawsze zawierają znacznik czasu, co sprawia, że nadają się do obliczania opóźnień. Typową metodą budowy takiej cechy jest wybranie rozmiaru okna dla funkcji opóźnienia i utworzenie na jej podstawie agregatów kroczących, takich jak średnia, odchylenie standardowe, minimum, maksimum itd.

Cecha II: *Opóźnienie błędy.* - Podobnie jak w przypadku telemetrii zapisy błędów też są oznaczone znacznikiem czasu. Jednak w przeciwieństwie do telemetrii, błędy oznaczane są przez kategorie do których należą. Tym samym pozostaje tylko agregowanie błędów w ramach kategorii.

Cecha III: *Dni od ostatniej wymiany konserwacyjnej* - Możliwymi cechami tego zestawu danych mogą być, na przykład, liczba wymian każdego składnika w ciągu ostatnich 3 miesięcy wykazująca na częstotliwość wymiany. Można też obliczyć, ile czasu minęło od wymiany ostatniego komponentu. Będzie on lepiej korelować z danymi awarii części składowych, gdyż im dłużej dany komponent jest używany, należy się spodziewać większego jego zużycia.

Cecha IV: *Cechy maszyny* - Cechy maszyny zastosowane są bezpośrednio, ponieważ zawierają opisowe informacje na temat rodzaju maszyn oraz ich wieku oraz, który rok jest serwisowana.

Po wskazaniu wszystkich cech zostają one połączone w jeden zestaw danych umieszczonych w kolumnach o nazwach: dataczasmaszynaID, napięcie_średnia, obrót_średnia, ciśnienie_średnia, wibracje_średnia, napięcie_odchylenie, obrót_odchylenie, ciśnienie_odchylenie, wibracje_odchylenie, błąd3licz, błąd4licz, błąd5licz, odostkomp1, odostkomp2, odostkomp3, odostkomp4, model, wiek, awaria.

Po określeniu cech, następnym etapem jest znakowanie. Przy wykorzystaniu wielokryterialnej klasyfikacji przewidywania awarii z powodu określonego problemu, znakowanie odbywa się w oknie czasowym bezpośrednio poprzedzającym awarię któregoś z komponentów. To okno dla wybranych cech znakuje się „może dojść do awarii" podczas gdy pozostałe oznaczone zostają jako oznaczone „normalne".

Zwraca się uwagę na fakt, że okno czasowe powinno być dobrane w zależności od przypadku produkcyjnego. W niektórych sytuacjach może być wystarczające, aby przewidzieć awarie z godzinnym wyprzedzeniem, podczas gdy w innych przypadkach potrzebne mogą być dnie a nawet tygodnie, aby umożliwić zamówienie i otrzymanie części koniecznej do wymiany.

Przedostatnim etapem jest modelowanie. W rozpatrywanym przypadku predykcja wykonana została przy użyciu wielokryterialnej regresji logistycznej z wykorzystaniem języka R. Dla predykcyjnych problemów utrzymania , strategia podziału zestawu danych zależna od czasu stanowi najlepsze podejście do oceny wydajności. Ocena ta odbywa się poprzez sprawdzanie i testowanie modelu na przykładach, które zanotowane zostały w późniejszym czasie niż przykłady treningowe. W przypadku, gdy okno podziału jest oznakowane jako bezpośrednio poprzedzające awarię, to okno nie jest znakowane i nie będzie użyte do nauki. Natomiast walidacja może być wykonywana poprzez wybranie różnych punktów podziału i oceny wydajności modeli utworzonych o tak sprecyzowany podział.

```
library(gbm)
#stwórz formułę
failure ~ voltmean + rotatemean + pressuremean + vibrationmean +
  voltsd + rotatesd + pressuresd + vibrationsd + voltmean_24hrs +
  rotatemean_24hrs + pressuremean_24hrs + vibrationmean_24hrs +
  voltsd_24hrs + rotatesd_24hrs + pressuresd_24hrs + vibrationsd_24hrs +
  error1count + error2count + error3count + error4count + error5count +
  sincelastcomp1 + sincelastcomp2 + sincelastcomp3 + sincelastcomp4 +
  model + age
trainformula <- as.formula(paste('failure',
                 paste(names(labeledfeatures)[c(3:29)],collapse=' + '),
                 sep=' ~ '))
trainformula

# trenuj model na trzech podziałach
set.seed(1234)
gbm_model1 <- gbm(formula = trainformula, data = trainingdata1,
         distribution = "multinomial", n.trees = 50,
         interaction.depth = 5, shrinkage = 0.1)
gbm_model2 <- gbm(formula = trainformula, data = trainingdata2,
         distribution = "multinomial", n.trees = 50,
         interaction.depth = 5, shrinkage = 0.1)
gbm_model3 <- gbm(formula = trainformula, data = trainingdata3,
         distribution = "multinomial", n.trees = 50,
         interaction.depth = 5, shrinkage = 0.1)

an example
summary(gbm_model1)
#Pokaż relatywny wpływ zmiennych na pierwszy model
Summary(gbm_model1)
```

List.1. Trenowanie modelu w wykorzystaniem biblioteki budującej model predykcyjny gbm (Generalized Boosted Regression Models) [3,7]

Proces kończy etap ewaluacji. W predykcji utrzymania według stanu kluczowe jest to na ile sporządzony model jest wstanie wykryć awarię. Poniżej w tabeli 2 przedstawiono porównanie prawdopodobieństw wykrycia awarii komponentów dla każdego rodzaju awarii dla trzech modeli. Wskaźniki przywołania awarii dla wszystkich komponentów, jak również brak awarii wynosi powyżej 90% oznacza, model był w stanie uchwycić powyżej 90% awarii poprawnie.

Tab. 2. Porównanie prawdopodobieństw wykrycia awarii komponentów

	awaria	gbm_model1_Recall	gbm_model2_Recall	gbm_model3_Recall
1	komp1	0.916666666666667	0.926470588235294	0.928104575163399
2	komp2	0.938202247191011	0.95983379501385	0.973684210526316
3	komp3	0.920673076923077	0.940625	0.950892857142857
4	komp4	0.948630136986301	0.928888888888889	0.909937888198758
5	brak	0.999842106137916	0.999833329861039	0.999820565907522

3. Podsumowanie

Artykuł stanowi opis poszczególnych etapów analizy danych produkcyjnych w celu stworzenia modelu prognostycznego przydatnego w ocenie i przewidywaniu awarii z powodu uszkodzenia elementów maszyny. Artykuł opisuje ogólny schemat postępowania składający się z czterech etapów. Modele predykcyjne budowane są zarówno za pomocą pakietów R i środowiska maszynowego uczenia w środowisku chmury obliczeniowej.

Zaprezentowana ścieżka jest zgodna z procesem data science. Co więcej, scharakteryzowane etapy budowy modelu analitycznego wspierającego przewidywanie wystąpienia awarii ze względu komponent maszyny, stanowi uniwersalny zestaw kroków, który może zostać odwzorowany dla innych przypadków.

Przy czym należy wskazać, że bez względu na rodzaj rozpatrywanych zagadnień i dostępnych źródeł danych kluczowe jest identyfikacja etapów i konsekwentna analiza. Tym samym pisywane zagadnienie posiada uniwersalny charakter i może być rozpatrywane w odniesieniu do wielu specyficznych zagadnień produkcyjnych.

Co więcej w ramach schematu, w zależności od wymagań techniczno-biznesowych możliwe jest proponowanie innych cech, a także wykorzystywanie innych bibliotek analitycznych języka R – właściwe dla analizowanego kontekstu środowiska produkcyjnego, co stanowi dalszy obszar działań autorów.

Literatura

1. Dhar V., Data science and prediction. Commun. ACM 56, 12 December 2013, 64-73
2. O'Neil, Cathy, Schutt, Rachel (2014) Doing Data Science, O'Reilly 2014
3. Boylu Fidan Uz, Predictive Maintenance Modelling Guide, dostępny w Internecie: https://gallery.cortanaintelligence.com/Collection/Predictive-Maintenance-Modelling-Guide-1, dn. 05.01.2016 r.
4. What is R? Introduction to R, dostępne w Internecie https://www.r-project.org/about.html, dn. 05.01.2016 r.
5. Pizoń J., Lipski J., Manufacturing Process Support Using Artificial Intelligence, Applied Mechanics and Materials, Vol. 791, pp. 89-95, Sept. 2015,

6. Pizoń J., Koncepcja wdrożenia technologii „Internetu rzeczy" w systemie logistycznym przedsiębiorstwa, Systemy Logistyczne Wojsk, nr 43, 2015,
7. Ridgeway G.‚Package'gbm' description, dostępny w Internecie: https://cran.r-project.org/web/packages/gbm/gbm.pdf, dn. 05.01.2016 r.
8. Saxena A., Goebel K.,"Turbofan Engine Degradation Simulation Data Set", NASA Ames Prognostics Data Repository (http://ti.arc.nasa.gov/tech/dash/pcoe/prognostic-data-repository/), NASA Ames Research Center, Moffett Field, CA

Mgr inż. Jakub PIZOŃ
Doktorant Studiów Doktoranckich na Wydziale Mechanicznym
Politechnika Lubelska
20-618 Lublin, ul. Nadbystrzycka 36
tel./fax: +48 81 538 44 84/+48 81 538 46 33
e-mail: jakub.pizon@pollub.edu.pl

Prof. dr hab. inż. Jerzy LIPSKI
Dr Tomasz CIEPLAK
Katedra Organizacji Przedsiębiorstwa
Politechnika Lubelska
20-618 Lublin, ul. Nadbystrzycka 38
tel./fax: +48 81 538 44 80/+48 81 538 46 81
e-mail: j.lipski@pollub.pl
t.cieplak@pollub.pl