

Współczesna Gospodarka



Contemporary Economy
Electronic Scientific Journal
www.wspolczesnagospodarka.pl

Vol. 8 Issue 1 (2017) 1-26
ISSN 2082-677X

UCZENIE MASZYNOWE DO SAMOORGANIZACJI SYSTEMÓW ROZPROSZONYCH W ZASTOSOWANIACH GOSPODARCZYCH

Jerzy Balicki, Waldemar Korlub

Streszczenie

W pracy omówiono uczenie maszynowe do samoorganizacji systemów rozproszonych w zastosowaniach gospodarczych ze szczególnym uwzględnieniem sieci neuronowych do predykcji finansowych oraz szacowania ratingu przedsiębiorstw. Oprócz sieci neuronowych, istotną rolę w przygotowaniu i testowaniu informatycznych systemów finansowych może pełnić programowanie genetyczne. Z tego powodu omówiono uczenie maszynowe w aplikacjach konstruowanych automatycznie. W celu uzyskania przewagi konkurencyjnej uczenie maszynowe może być wykorzystane do zarządzania zasobami samoorganizujących się mgieł i chmur obliczeniowych do realizacji obliczeń biznesowych w firmie. Opisano także wybrane zastosowania gospodarcze samoorganizujących się systemów rozproszonych, w tym odniesiono się do badań nad wiarygodnością kredytobiorców, a także wskazano na metody stosowane do szacowania ryzyka sektora bankowego. Na zakończenie przedstawiono kluczowe wnioski oraz kierunki dalszych badań.

Słowa kluczowe: predykcja giełdowa, rating przedsiębiorstw, inteligentne systemy finansowe, programowanie genetyczne, sztuczne sieci neuronowe.

Wstęp

Uczenie maszynowe do niedawna odgrywało rolę drugoplanową w zastosowaniach biznesowych. Wprawdzie po pokonaniu przez komputer IBM *Deep Blue* mistrza świata w szachy w 1996 roku eksperci spodziewali się, że niekonwencjonalne zastosowania inteligentnych aplikacji pojawią się w obszarze gospodarki, to jednak proces ten zachodził relatywnie wolno i był ukierunkowany zasadniczo na wykorzystanie sztucznych sieci neuronowych (ANN, od ang. *Artificial Neural Networks*) do antycypacji inwestycji giełdowych.

Przełom nastąpił w 2010 roku, kiedy komputer IBM *Watson* pokonał dwóch najlepszych graczy w amerykańskim quizie telewizyjnym *Jeopardy* – w Polsce quiz ten znany jest jako *Va Banque*. W superkomputerze zainstalowano 32 400 ośmiordzeniowych procesorów, które mogą współbieżnie wykonywać około miliona wątków obliczeniowych. W tak rozproszonym środowisku uzyskano znaczące przyspieszenie działania prawie 100 algorytmów sztucznej inteligencji.

cji, które usiłują w krótkim czasie znaleźć odpowiedź dla wybranej kategorii w grze: *ludzie, miejsca, literatura, czy zagadki słowne*. Wybór następuje na podstawie największej wartości poziomu ufności dla wyznaczonych odpowiedzi. Powyższe podejście dla innych danych wdrożono w 2015 roku w diagnostyce onkologicznej w kilku amerykańskich klinikach, w tym w wiodącej klinice *Mayo Clinic*. W celu ułatwienia wykorzystania zasobów systemu udostępniono usługę diagnostyki medycznej w chmurze obliczeniowej.¹

Do tego „wyścigu zbrojeń” dołączyło *Google* z systemem *AlphaGo*, który pokonał mistrza świata w najbardziej złożonej dwuosobowej grze planszowej *Go* w marcu 2016 roku. W systemie może działać 64 współbieżnych algorytmów uczenia ze wzmocnieniem, które wyznaczają strategię gry na 1 920 procesorach wielordzeniowych oraz na 280 procesorach graficznych.²

W odpowiedzi w październiku 2016 roku IBM rozpoczęło realizację projektu dotyczącego wykorzystania w biznesie oprogramowania *Watson* współpracującego z *Internetem Rzeczy* (ang. *Internet of Things*, IoT). W ramach projektu o budżecie 3 mld USD ok. 6 000 firm zamierza wdrożyć inteligentny system *IBM Watson IoT* w celu usprawnienia produktów i uzyskania przewagi konkurencyjnej. W szczególności kilkadziesiąt firm współpracuje z europejskim centrum IBM w Monachium w ramach budżetu 200 mln USD, w tym producent pozepszółów do środków transportowych firma *Schaeffler*. Na podstawie analizy napływających danych z eksploatowanych pojazdów zamierza się usprawniać rozwiązania konstrukcyjne. Ponadto holenderski producent miejskich dronów *Aerialtronics* zamierza wykorzystać inteligentne systemy do monitorowania miast pod kątem optymalizacji ruchu, minimalizacji zanieczyszczeń oraz zwiększenia poziomu bezpieczeństwa.³

Należy się zatem spodziewać licznych krajowych zastosowań gospodarczych inteligentnych systemów klasy *IBM Watson IoT*. W szczególności bardzo interesującym otwartym zagadnieniem badawczym jest samoorganizacja systemów rozproszonych, w tym wykorzystanie uczenia maszynowego do zwiększania autonomii sieci aplikacji pracujących we mgle obliczeniowej. Dlatego też celem artykułu jest zaproponowanie efektywnych zasad uczenia maszynowego do samoorganizacji systemów rozproszonych w zastosowaniach gospodarczych.

Praca zorganizowana jest następująco. W pierwszej sekcji omówiono nadzorowane uczenie ANN do predykcji finansowych. Druga sekcja poświęcona jest programowaniu genetycznemu w systemach finansowych. Uczenie maszynowe w aplikacjach programowanych genetycznie omówiono w sekcji trzeciej. Z kolei w sekcji czwartej scharakteryzowano samoorganizujące się chmury obliczeniowe stosowane do realizacji obliczeń biznesowych. Wybrane zastosowania gospodarcze samoorganizujących się systemów rozproszonych opisano w sekcji piątej. W szczególności odniesiono się do metod ewolucyjno-neuronowych w kontekście badań nad wiarygodnością kredytobiorców, a także wskazano na neuronowe metody stosowane do szacowania ryzyka sektora bankowego w celu wczesnego ostrzegania przed kryzysami. Na zakończeniu przedstawiono kluczowe wnioski oraz kierunki dalszych badań.

1. Nadzorowane uczenie sieci neuronowych do predykcji finansowych

Sztuczne sieci neuronowe stosuje się do predykcji inwestycji giełdowych, szacowania ratingu firm oraz do „wczesnego ostrzegania” przed kryzysami finansowymi. Bankowość to system naczyń połączonych, w którym upadłość jednego banku może wywołać efekt domina. Regulacje ostrożnościowe relatywnie wolno zmieniają się w czasie. Natomiast ANN oceniają

¹ *IBM Watson*. <http://www.ibm.com/watson/what-is-watson.html>, dostęp: 20 października 2016.

² *Explore the AlphaGo Games*. www.deepmind.com, Dostęp: 20 października 2016.

³ *IBM Watson*. *Ibidem*. dostęp: 20 października 2016.



sytuację ekonomiczną oraz kondycję finansową banków dynamicznie w dowolnym momencie. W ten sposób dysponujemy efektywnym narzędziem ostrzegania przed kryzysem bankowym.⁴

Warto podkreślić, że modelowy kryzys bankowy zazwyczaj przebiega w trzech dwuletnich fazach. W pierwszej fazie obserwujemy perturbacje związane z bieżącym funkcjonowaniem banków, co jest dobrym okresem do rozpoczęcia działań łagodzących skutki nadchodzącego kryzysu. Podczas drugiej fazy następuje znaczące dofinansowanie upadających banków, co wpływa na co najmniej 10% spadek PKB danego kraju. W trzeciej fazie gospodarka i sektor bankowy zazwyczaj wracają do pierwotnej kondycji – to jest najlepszy okres na przemyślaną referomę bankowości i unowocześnienie gospodarki.⁵

Warto podkreślić, że niekiedy skutki kryzysu bankowego mogą być poważniejsze.⁶ Kryzys bankowy w Grecji wywołał poważne skutki ekonomiczno-społeczne obserwowane od 2009 roku. W rezultacie należy się nawet spodziewać wyjścia Grecji ze strefy Euro. Nieco lepiej sytuacja wygląda w Portugalii i Włoszech, w których sektory bankowe wymagają istotnego wsparcia. W Polsce PKB początkowo spadł o 13,5%, z 364 mld euro w 2008 roku do 315 mld euro w 2009 roku. Załamanie nie było długotrwałe, a w 2015 roku PKB osiągnął już 428 mld euro wg danych Eurostatu. Warto zwrócić uwagę na fakt, że w Szwajcarii właściwie nie zaobserwowano kryzysu bankowego, a PKB wzrósł z 439 mld euro w 2010 roku do 605 mld euro w 2015 roku. Helweci postawili na inteligentną informatyzację i wprowadzili odważne reformy w gospodarce w najlepszym do tego okresie – po wystąpieniu kryzysu bankowego na świecie.⁷

W Szwajcarii realizowany jest od 2005 roku flagowy projekt Unii Europejskiej o nazwie *Blue Brain*, którego celem głównym jest symulacja działania mózgu człowieka w wybranych zastosowaniach, w tym gospodarczych. Budżet projektu to ok. dwa mld euro w ciągu 15 lat. Cel pomocniczy odnosi się do zastosowań metod neuronowych w biznesie, w tym w finansach i bankowości. Na Uniwersytecie w Lozannie zbudowano superkomputer *IBM Blue Brain*, w którym symulowano działanie 36 000 neuronów połączonych za pomocą 37 mln synaps. Natomiast na Uniwersytecie w Waterlo w Kanadzie uproszczono model neuronu w stosunku do modelu *Blue Brain*, co umożliwiło symulację działania aż 2,5 mln neuronów na superkomputerze *Spaun*.⁸ Przykładowy efekt działania ANN przedstawiono na rysunku 1, na którym zobrazowano ratingi ok. 85 tys. firm. Klasyfikacji dokonano na podstawie wskaźnika $WC_{TA} = \text{kapitał operacyjny/aktywa}$, przy czym *WC* - *working capital*, *TA* - *total assets*.

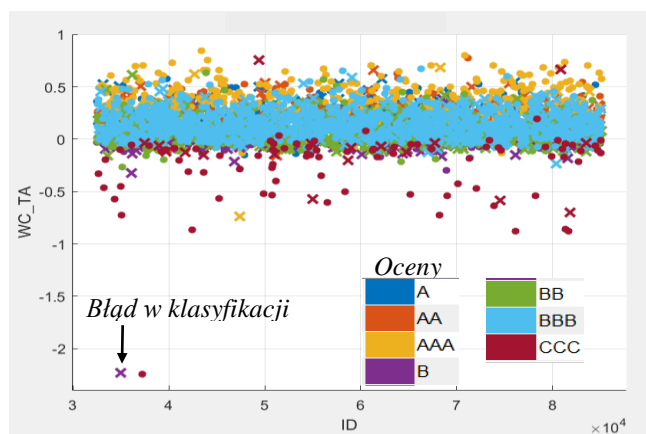
⁴ Frankel J. A., Rose A. K.: *Currency crashes in emerging markets: an empirical treatment*. Journal of International Economics, vol. 41, no. 3-4, 1996, p. 352

⁵ Hanschel E., Monnin P.: *Measuring and forecasting stress in the banking sector: evidence from Switzerland*. Investigating the Relationship between the Financial and Real Economy, BIS Papers, no. 22, 2005, p. 432.

⁶ Kaminsky G. L., Reinhart C. M.: The twin crises: the causes of banking and balance-of-payments problems. American Economic Review, vol. 89, no. 3, 1999, p. 480.

⁷ Eurostat. *Your key to European statistics*. <http://ec.europa.eu/eurostat/web/national-accounts/data/main-tables>, dostęp: 7.10.2016 r.

⁸ Eliasmith C. et al.: *A large-scale model of the functioning brain*, Science, Vol. 338, 2012, pp. 1202–1205, p. 1203.



Rysunek 1. Rating firm szacowany za pomocą sztucznej sieci neuronowej

Źródło: Opracowanie własne.

1.1. ANN do predykcji ratingu przedsiębiorstw

W modelu klasyfikacji każda firma cechuje się unikatowym identyfikatorem *ID* (rys. 1). Kropka oznacza prawidłową klasyfikację, a krzyżyk – błąd w predykcji. Na wejście klasyfikatora wprowadzane są dane *ID* oraz *WC_TA*, a na wyjściu otrzymujemy rating firmy. Przydatność modelu jest istotna z punktu widzenia banku, jeśli firma stara się w nim o kredyt. Jeśli zakładane jest konsorcjum firm do realizacji dużego projektu biznesowego, to rating potencjalnych konsorcjantów może decydować o wygraniu przetargu.

Warto omówić aspekt dotyczący powiązania między predykcją a klasyfikacją. W tym wypadku udało się dokonać transformacji zagadnienia predykcji ratingu firmy w problem klasyfikacji za pomocą zabiegu polegającego na wykorzystaniu do ratingu siedmiu możliwych ocen w skali od *CCC* do *A* (rys. 1). Sieć neuronowa lub inny klasyfikator może zatem nadawać ratingi w ramach siedmiu klas. W ten sposób aplikacja szacuje rating firmy w relatywnie krótkim czasie. Natomiast agencje ratingowe podają aktualny rating zazwyczaj znacznie rzadziej niż są publikowane dane firm o kapitale operacyjnym i aktywach. Jeśli jednak predykcja opiera się na danych uszeregowanych czasowo, to klasyfikatorów nie można zastosować do predykcji.

Czas treningu sieci dla omawianego zastosowania to ok. trzy godziny na komputerze klasy PC oraz ok. siedmiu minut w wypadku wykorzystania przetwarzania równoległego na procesorach graficznych. Zastosowanie superkomputerów, gridów, chmur lub mgieł obliczeniowych może znacząco zredukować niezbędny czas do ułamków sekund, co jest szczególnie istotne, jeśli liczba firm będzie większa lub model predykcji zostanie rozbudowany o dodatkowe wskaźniki. Zazwyczaj w modelach predykcji ratingów firm wykorzystuje się wartość zysku zatrzymanego *RE* (ang. *Retained Earnings*), który pozostaje w korporacji po wypłacie dywidendy. Wskaźnik *RE_TA*, który odzwierciedla stosunek zysku zatrzymanego *RE* do wartości łącznych aktywów *TA*, jest zatem drugą istotną cechą modelu klasyfikacji przedsiębiorstw.

W trzecim wskaźniku *EBIT_TA* stosowanym do predykcji ratingu firm wykorzystuje się zysk operacyjny *EBIT* (ang. *Earnings Before deducting Interest and Taxes*), którym jest zysk przed odliczeniem odsetek i podatków. *EBIT* umożliwia porównywanie firm bez względu na obciążenia podatkowe, stopień wykorzystania dźwigni finansowej, czy wielkość zysków i strat nadzwyczajnych. Zysk operacyjny z księgowego punktu widzenia to *sprzedaż – koszty operacyjne*. Natomiast z finansowego punktu widzenia *EBIT* wyznacza się sumując *zysk brutto* oraz *odsetki*. Trzecia metoda oszacowania zysku operacyjnego polega na redukcji *marży brutto* o *koszty stałe* przedsiębiorstwa. Stosunek zysku operacyjnego do wartości aktywów to właśnie wskaźnik *EBIT_TA* (rys. 2).

Wartość rynkowa kapitału własnego (*MVE* od ang. *Market Value of Equity*) wpływa również na rating przedsiębiorstwa. Kapitał własny to wartościowy odpowiednik składników majątku, do których firmie przysługuje prawo własności. Kapitał własny wnoszony jest do firmy przez właścicieli (założycieli), którzy w ten sposób nabywają prawo do zwrotu kapitału w razie likwidacji firmy oraz do uczestniczenia w podziale zysku. Kapitał własny zwiększa się poprzez przeznaczenie części zysku na cele rozwojowe. Ten rodzaj kapitału może mieć postać pieniężną lub rzeczową.

Podstawowym składnikiem wartości rynkowej kapitału własnego przedsiębiorstwa jest jego kapitalizacja giełdowa będąca iloczynem aktualnego kursu akcji spółki oraz liczby akcji firmy zarówno w obrocie giełdowym, jak i poza obrotem giełdowym. Kapitalizacja giełdowa spółki stanowi w pewnym sensie wycenę jej kapitału (aktywów netto) dokonaną przez rynek kapitałowy. W związku z tym, że w tradycyjnym sprawozdaniu finansowym przedsiębiorstwa nie uwzględnia się wielu współczesnych aspektów rynku finansowego, takich jak wartość nazwy marki przedsiębiorstwa, kapitalizacja giełdowa może znacząco się różnić od wyliczeń księgowych kapitału.

Wartość rynkowa kapitału własnego wyliczona w oparciu o kapitalizację giełdową jest zwiększana o wartość giełdowych instrumentów pochodnych, którą wyznacza się zazwyczaj za pomocą rozwiązania równań różniczkowych w modelu *Blacka–Scholesa*. Ostateczna wartość rynkowa kapitału własnego przedsiębiorstwa zawiera także wartość zamiennych papierów wartościowych (zamiennych obligacji, akcji uprzywilejowanych o regularnych odsetkach, uprzywilejowanych akcji konwertowanych do akcji zwykłych, czy warrantów subskrypcyjnych, które są wyceniane bez redukcji o koszty konwersji). Stosunek rynkowej wartości kapitału do wartości księgowej długu (ang. *Book Value of Total Debt*, *BVTD*) to czwarty istotny wskaźnik *MVE_BVTD*, który zazwyczaj uwzględnia się jako wejście do sztucznej sieci neuronowej.

Piąty wskaźnik *S_TA* to stosunek sprzedaży do aktywów. Poprawność predykcji na poziomie 75% jest możliwa w oparciu o klasyczne klasyfikatory, takie jak: sieci neuronowe, drzewa decyzyjne, czy też metodę wektorów wspierających *SVM* (ang. *Support Vector Machine*). Uwzględnienie większej liczby firm, większej liczby wskaźników, a przede wszystkim uzyskanie wyższej precyzji prognozy jest możliwe z wykorzystaniem znacząco bardziej rozbudowanych klasyfikatorów trenowanych w wydajnym środowisku obliczeniowym.

W szczególności uczenie odpowiednio dużej sztucznej sieci neuronowej wymaga zastosowania nowoczesnych metod uczenia wielopoziomowego (ang. *deep learning*). Uczenie hierarchiczne to pewna klasa metod uczenia maszynowego, w której wykorzystuje się kaskadę wielu warstw nieliniowych neuronów do ekstrakcji cech, a następnie do przetwarzania danych wejściowych. Do klasyfikacji stosuje się uczenie nadzorowane, a do analizy wzorców – uczenie nienadzorowane. Hierarchiczna reprezentacja cech podlega nienadzorowanemu uczeniu na wielu poziomach cech lub danych. Wyższy poziom cech opiera się na cechach niższego poziomu. W kaskadzie warstw neuronów występują warstwy ukryte neuronów oraz złożone procedury obliczeniowe. Stosowane są również wielopoziomowe sieci wielowarstwowe ze sprzężeniami zwrotnymi między warstwami (ang. *Deep Belief Networks*) oraz wielopoziomowe maszyny Boltzmana (ang. *Deep Boltzmann Machines*).⁹

⁹ Balicki J., Dryja P., Korłub W., Przybyłek P., Tyszka M., Zadroga M., Zakidalski M.: Metody neuronowe do prognozowania finansowego, *Współczesna Gospodarka*, Vol. 7, No. 2, 2016, pp. 21-36, p. 23



Rysunek 2. Uwzględnienie wskaźnika EBIT_TA w modelu predykcyjnej sieci neuronowej poprawia precyzję prognozy

Źródło: Opracowanie własne.

1.2. Neuronowa predykcja giełdowa a klasyfikacja ratingów

W predykcji inwestycji na giełdzie papierów wartościowych jednokierunkowa ANN złożona jest zazwyczaj z kilku warstw neuronów: wejściowej, 3 warstw ukrytych, a także warstwy wyjściowej.¹⁰ W wypadku uczenia z nauczycielem, na wejścia podawane są dane historyczne z giełdy, a następnie sieć oblicza rekomendację dla gracza. Rezultat ten jest porównywany z wzorcowym wynikiem, co umożliwi skorygowanie wag za pomocą wybranego algorytmu treningu.¹¹ W predykcji mogą być stosowane różne zestawy cech na wejściu sieci. Ponadto istotna jest długość czasu, na podstawie której zwielokrotniane są niektóre cechy wejściowe.¹²

Dla przyjętego modelu sieci dane dzielone są na trzy zbiory: treningowy, walidacyjny i testowy. Na podstawie zbioru treningowego sieć jest uczona za pomocą algorytmu typu *backpropagation* czy metody gradientowej w celu minimalizacji wybranej funkcji błędu. Inna metoda treningu polega na inicjacji wag synaptycznych za pomocą symulowanego wyżarzania, algorytmu genetycznego czy też tabu search, a następnie naprzemiennie stosuje się metody gradientów sprzężonych i wybranej metaheurystyki. Warto zauważyć, że zbiór walidacyjny nie jest używany do treningu sieci, gdyż powinien umożliwiać zatrzymanie procesu uczenia w iteracji, w której funkcja błędu zaczyna rosnąć dla zbioru walidacyjnego. Zapobiega to wystąpieniu zjawiska przeuczenia sieci, co zazwyczaj występuje podczas przedłużającego się treningu sieci.

Wówczas wprawdzie błąd średniokwadratowy dla zbioru treningowego jest zadowalająco niewielki, ale ten sam błąd dla zbioru walidacyjnego jest znacząco większy i dalszy proces uczenia powoduje jego wzrost. Prawidłowa jest sytuacja zatrzymania treningu, gdy błąd średniokwadratowy dla zbioru treningowego maleje, a dla zbioru walidacyjnego zaczyna rosnąć.

¹⁰ Gatley E. *Sieci Neuronowe. Prognozowanie finansowe i projektowanie systemów transakcyjnych*. WIG-Press, Warszawa 1999, s. 221.

¹¹ Davis E. P., Karim D.: *Comparing early warning systems for banking crises*. Journal of Financial Stability, vol. 4, no. 2, 2008, p. 89.

¹² Srivastava R. P.: *Automating judgmental decisions using neural networks: a model for processing business loan applications*, Proceedings of the 1992 ACM Annual Conf. on Communications, p. 355.

Natomiast zbiór testowy nie jest wykorzystywany ani do treningu ani do zatrzymania uczenia. Zbiór ten odgrywa rolę pomocniczą do obserwacji zmiany poziomu błędów popełnianych przez sieć podczas treningu oraz podczas działania w warunkach rzeczywistych.¹³

Na rysunku 3 przedstawiono liczebności błędnych (kolor czerwony) i poprawnych (kolor zielony) prognoz w tablicy pomyłek podczas treningu sieci neuronowej. Wraz z kolejnymi epokami uczenia minimalizowany jest zazwyczaj błąd średniokwadratowy, co w tablicy pomyłek odpowiada zwiększaniu się wartości elementów na przekątnej głównej.

True class \ Predicted class	A	AA	AAA	B	BB	BBB	CCC
A	409	51				115	
AA	64	295	26				
AAA		55	523				2
B				143	160	2	15
BB				53	690	183	1
BBB	64				152	799	
CCC		1	1	45	1		82

Rysunek 3. Liczebności prognoz w tablicy pomyłek podczas treningu sieci neuronowej

Źródło: Opracowanie własne.

Natomiast wartości elementów poza przekątną główną maleją. Przeuczenie sieci jest wówczas, gdy w tablicy pomyłek dla danych walidacyjnych zachodzi proces odwrotny niż w tablicy pomyłek dla zbioru treningowego. W tym wypadku należy zatrzymać proces treningu.

Warto wspomnieć, że wielkości dużych zbiorów uczących zależą od częstotliwości pobierania danych z giełdy, co może odbywać się nawet co milisekundę.¹⁴ Wymaga to jednak dokonywania transakcji na algorytmicznej platformie wymiany papierów wartościowych. Z tego powodu istotne jest zastosowanie uczenia hierarchicznego z wykorzystaniem *Big Data*.^{15 16} Na giełdzie krajowej predykcję utrudnia niewielka ilość dostarczanej informacji – dostępne są jedynie: cena otwarcia, najwyższa cena dnia, najniższa cena dnia, cena zamknięcia oraz wolumen obrotu. Z tego względu prognoza może dotyczyć ceny otwarcia lub zamknięcia, ale wydaje się to tylko kwestią rozwoju giełdy, kiedy będzie można prognozować, np. co milisekundę. Na pewno dobrym krokiem jest instalacja nowego systemu informatycznego w 2014 roku.

Ważną grupę metod uczenia stanowią metody uczenia sieci neuronowych bez nadzorowania, które mogą być wykorzystywane do wykrywania anomalii na giełdzie. Anomaliami tymi może być zarówno niespotykana sytuacja na rynku, jak też sprawdzaniu podlegać mogą pojedyncze transakcje na rynku w celu wykrycia nadużyć.¹⁷

Na rysunku 4 zobrazowano przestrzeń błędów popełnianych przez klasyfikator dla zadanej klasy. Bez względu na sposób treningu ANN metodyka oceny jakości klasyfikatora jest jednakoowa. *Stopa błędów FPR* (ang. *false positive rate*) to częstość popełniania błędów przez klasy-

¹³ Nazari M., Alidadi M.: *Measuring credit risk of bank customers using artificial neural network*. Journal of Management Research, Vol. 5, No. 2, 2013, p. 23.

¹⁴ Demirguc-Kunt A., Detragiache E.: *Monitoring banking sector fragility: a multivariate logit approach*. World Bank Economic Review, vol. 14, no. 2, 2000, p. 287.

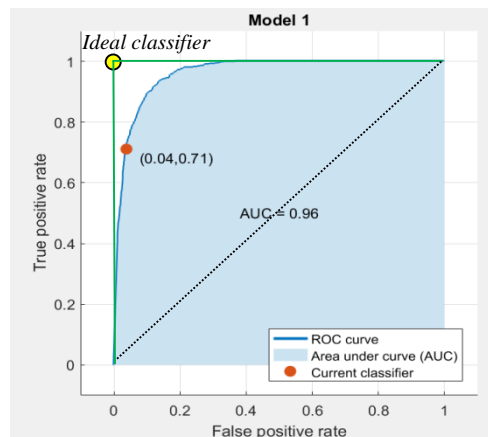
¹⁵ *The “Big Data” Solution For Wall Street*. Stock Forecast Based On a Predictive Algorithm. <http://iknowfirst.com/the-big-data-solution-for-wall-street>, dostęp: 10.10.2016 r.

¹⁶ Sensmeier L.: *Building Stock Trading Strategies: 20% Faster with Hadoop*. <http://hortonworks.com/blog/building-stock-trading-strategies-20-faster-with-hadoop/>, dostęp: 10.10.2016 r.

¹⁷ Ferdousi Z., Maeda A.: *Anomaly detection using unsupervised profiling method in time series data*, Proc. Of 10th Conference on Advances in Databases and Information Systems, Tesaloniki 2006.



fikator dla wybranej klasy, np. „sprzedaj akcje firmy X”.¹⁸ Jeśli stopa błędów wynosi 0,04, to 4% wskazań sieci neuronowej stanowią odwrotne odpowiedzi do decyzji „sprzedaj akcje firmy X”. W idealnym przypadku stopa błędów powinna wynosić 0. Natomiast *poziom wiarygodności TPR* (ang. *true positive rate*) oznacza częstość wskazań przez sieć klasy „sprzedaj akcje firmy X” dla prawidłowych wskaźników na wejściu sieci. Jeśli *poziom wiarygodności klasyfikacji* wynosi 0,71, to 71% decyzji sieci jako „sprzedaj akcje firmy X” klasyfikuje się jako prawidłowe. Idealny poziom wiarygodności to 1.



Rysunek 4. Ocena jakości klasyfikatora w przestrzeni błędów dla klasy „sprzedaj akcje firmy X”

Źródło: Opracowanie własne.

W wyniku predykcji dla różnych zbiorów danych jakość pracy klasyfikatora dla wybranej klasy może ulegać zmianie, co odpowiada przemieszczaniu się po krzywej pracy klasyfikatora ROC (ang. the *Receiver Operating Characteristic*). Krzywa ROC obrazuje zmiany poziomu wiarygodności w zależności od zmian stopy błędów dla wybranej klasy. Dla każdej klasy krzywa ROC jest na ogół odmienna, mimo że klasyfikator jest ten sam. Idealny klasyfikator pracuje bezbłędnie, a jego krzywa pracy to kąt prosty o wierzchołku [0; 1] (kolor zielony na rys. 4). Natomiast losowy klasyfikator pracuje zgodnie z krzywą oznaczoną czarną linią przerywaną.¹⁹

Im większy obszar AUC (ang. the *Area Under Curve*) pod krzywą klasyfikacji ROC, tym wyższej jakości jest klasyfikator dla zadanej klasy. Z doświadczenia autorów wynika, że najlepszymi klasyfikatorami są metody oparte na sztucznych sieciach neuronowych oraz na programowaniu genetycznym. Dobrych klasyfikacji można uzyskać za pomocą metod: wektorów wspierających SVM, najbliższych sąsiadów *kNN* oraz *k*-średnich. Porównanie klasyfikatorów utrudnia obszerna problematyka klasyfikacji, a nawet „odporność” niektórych klas dla relatywnie niezbyt złożonych problemów.

Przed wykorzystaniem danych do uczenia sieci, należy poddać je procesowi wstępnego przetwarzania. Pierwszym krokiem jest sprowadzenie predykcji giełdowej do analizy szeregów czasowych. Następnie, przykłady uczące powinny być rozbudowane o wartości wskaźników giełdowych: wskaźniki zależne lub niezależne od opóźnienia. Wskaźniki zależne od opóźnienia to takie, których parametrem jest liczba kolejnych wartości, a przykładem takiego wskaźnika jest średnia krocząca. W jej przypadku wartość opóźnienia określa, z jakiej liczby historycznych wartości obliczana jest wartość średniej. Takie wskaźniki pozwalają wyznaczyć kierunek tren-

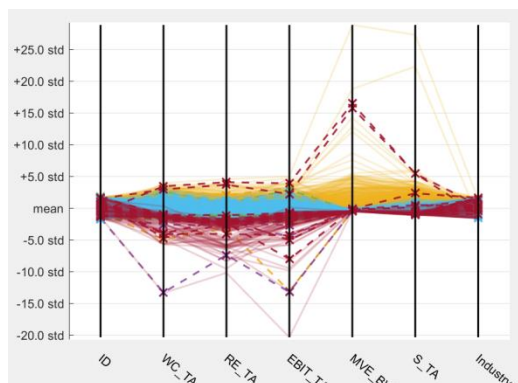
¹⁸ Oet M., Eiben R., Bianco T., Gramlich D., Ong S., Wang J.: *SAFE: an early warning system for systemic banking risk*. Proc. of the 24th Australasian Finance and Banking Conference, SSRN, 2011, p. 232.

¹⁹ Pandey V., Wee-Keong Ng, Ee-Peng Lim: *Financial advisor agent in a multi-agent financial trading system*. Proc. 11th Int. Workshop on Database and Expert Systems Applications, 2000, p. 482.

du, a także pomagają przewidywać jego zmianę oraz agregować informacje z dłuższego okresu. Wyznacza się także inne wskaźniki: regresja liniowa, kierunek regresji liniowej, tempo zmian, indeks względnej stabilności, a także wykładnicza średnia krocząca. Ponadto oblicza się wskaźniki niezależne od opóźnienia: miarę zbieżności i rozbieżności średnich ruchomych, indeks równowagi ceny do liczby akcji, czy oscylator *Chaikina*.²⁰

Wskaźniki papierów wartościowych w wypadku predykcji inwestycyjnych lub wskaźniki przedsiębiorstw w wypadku prognoz ich ratingu mogą wpływać na poprawę jakości klasyfikacji.²¹ Na rysunku 5 zobrazowano błędy klasyfikacji w odniesieniu do szacowania ratingów spółek. Na pionowych osiach reprezentujących wskaźniki klasyfikowanej spółki. Zauważono, że ze stosunkiem zysku operacyjnego do wartości aktywów *EBIT_TA* związana jest największa liczba nieprawidłowych oszacowań ratingów. Aby poprawić jakość klasyfikacji pod tym względem można wprowadzić dodatkowe dane do zbioru treningowego, dokonać ponownej ekstrakcji cech, zmienić algorytm klasyfikujący lub algorytm uczący. Dla tego samego modelu predykcji możliwe są liczne kombinacje uporządkowanych czwórek (*zbiór treningowy, wektor cech, algorytm klasyfikujący, algorytm uczący*), co może być efektywnie zweryfikowane i porównane za pomocą bardzo wyajnych systemów rozproszonych w ciągu milisekund.

Sieci neuronowe wykorzystywane są w wielu innych obszarach finansowych, np. do optymalizacji portfela akcji, gdyż sprawdziły się w problemach NP-trudnych. Warto wymienić także: analizę zdolności kredytowej klientów banków, analizę ryzyka związanego z udzieleniem kredytu hipotecznego, opracowanie strategii ofertowych, klasyfikację ryzyka giełdowych instrumentów finansowych, wykrywanie regularności w zmianach cen instrumentów finansowych oraz prognozowanie upadłości firm. Sieci neuronowe mogą również identyfikować lokalne zaburzenia rynku czy też zależności występujące przez krótki czas na rynkach finansowych.



Rysunek 5. Graficzne zobrazowanie błędów klasyfikatora w przestrzeni danych wejściowych

Źródło: Opracowanie własne.

Warto nadmienić, że modelem hybrydowym wykorzystującym sztuczną inteligencję do przewidywania kursu na giełdzie jest model składający się z morfologicznej sieci neuronowej MMNN (ang. *Morphological Neural Network*) oraz z algorytmu optymalizacji roju cząstek PSO (ang. *Particle Swarm Optimizer*).²² Rout et. al. zaproponowali do prognozowania indeksów giełdowych w 2011 roku nieco inny model hybrydowy, w którym wykorzystano sieć *Adaptive Linear Neurons* (ang. *Adaptive Linear Neurons*) trenowaną za pomocą ewolucji różnicowej DE (ang. *diffe-*

²⁰ Pietrzak E., Markiewicz M. (red.): *Finanse, bankowość i rynki finansowe*, Wyd. Uniwersytetu Gdańskiego, Gdańsk 2006.

²¹ Pietrzak J.: *Czynniki przewagi konkurencyjnej na rynku bankowych usług detalicznych*, Wyd. Uniwersytetu Gdańskiego, Gdańsk 2002, s. 32.

²² Chaveesuk R., Srivaree-Ratana C., Smith A.E.: *Alternative neural network approaches to corporate bond rating*. *Journal of Engineering Valuation and Cost Analysis*, vol. 2, 1999, ss. 117.



rential evolution).²³ Wykorzystano 3 228 danych z lat 1994-2006, na podstawie których utworzono 3 168 wzorców, z czego 2 510 włączono do zbioru uczącego, a 658 - do testowego. Sieć prognozuje wartości dwóch indeksów giełdowych DJIA oraz S&P 500 z wyprzedzeniem jednego, trzech i siedmiu dni. W badaniu wykorzystano średni bezwzględny odsetek błędów *MAPE* (ang. *mean absolute percentage of error*), który wyznacza się w oparciu o y – rzeczywistą cenę akcji, \hat{y} – przewidywaną cenę akcji oraz M – liczbę wzorców w zbiorze treningowym. Eksperymenty numeryczne wykazały większą szybkość oraz mniejszą liczbę błędów popełnionych przez algorytm DE w stosunku do PSO.²⁴

1.3. Wybrane tendencje w inwestowaniu finansowym

Inteligentne klasyfikatory trenowane na superkomputerach są stosowane podczas bardzo intensywnego handlu walorami na dużą skalę (ang. *High Frequency Trading, HFT or High Volume Trading, HVT*) obejmującego tysiące transakcji na sekundę. Na wybranych giełdach z wydajnymi systemami informatycznymi można być właścicielem akcji przez mili-, a nawet mikrosekundy, co może prowadzić także do znaczących zysków lub strat.²⁵ Inwestorzy indywidualni i instytucje finansowe mogą współpracować nie tylko z maklerami czy doradcami inwestycyjnymi, ale także z brokerami oraz animatorami rynków giełdowych.

Ze względu na dużą konkurencję klasyfikator jest uczony permanentnie w cyklu: *uczenie->testowanie->predykcja->uczenie-> ...*. Dla inwestorów wykorzystujących *HFT* znaczenie ma każda milisekunda, gdyż opóźnienie w reakcji może oznaczać, że transakcję zrealizuje klasyfikator innego inwestora. Warto zauważyć, że zysk z jednej transakcji może być mniejszy od ułamkowych części jednostki monetarnej, a w przypadku transakcji *Forex* – jednego pipsa. W ciągu milisekundy wybitny człowiek-makler wyposażony w zaawansowany program do analizy technicznej jest bez szans wobec wytrenowanego klasyfikatora zbudowanego w oparciu o *IBM Watson IoT, AlphaGo, Blue Bran* czy *Spaun*, podobnie jak wcześniej mistrzowie w warcaby, tryktraka, szachy, go, niektóre quizy telewizyjne czy wybrane gry komputerowe.

Warto podkreślić skalę zjawiska. W USA podczas kryzysu bankowego w 2009 roku wykorzystywano *HFT* do handlu ok. 70% wolumenu papierów wartościowych. Natomiast po wyjściu z kryzysu bankowego w 2012 roku intensywność handlu walorami obniżyła się do 50%. Przyczyną był 36 minutowy kryzys *Flash Crash* na nowojorskich giełdach 6 maja 2010 roku.

Pewna grupa aplikacji nieuczciwych inwestorów intensywnie składała mikrooferty kupna/sprzedaży wybranych akcji. Jednocześnie rozesłali oni informacje o oczekującej „dużej ofercie” dotyczącej tychże akcji, która powinna być zrealizowana za pewien czas. Ceny walorów wg „dużej oferty” były jednak o wiele bardziej korzystne niż aktualne ceny mikroofert wystawione przez oszustów. W tej sytuacji wiele programów innych inwestorów zrealizowało aktualne mikrooferty ze względu na dużą transakcję oczekującą. Następnie, licząc na zysk, przedstawili je inwestorowi, który powinien odkupić akcje. Tyle tylko, że oczekiwana transakcja została anulowana tuż przed jej zrealizowaniem.

W ciągu 36 minut nastąpiło setki tysięcy niekorzystnych transakcji dla większości inwestorów, co wywołało panikę na giełdach, a w rezultacie jedną z najpoważniejszych jednodniowych turbulencji w historii giełd. W 2015 roku wprowadzono regulacje zabraniające uwzględniania cen kontraktów terminowych, które nie zostały zrealizowane, ze szczególnym uwzględnieniem instrumentów pochodnych klasy *futures* lub *forward*.

²³ Rout M., Majhi B., Majhi R., Panda G.: *Novel stock market prediction using a hybrid model of adaptive linear combiner and differential evolution*. Communications in Computer and Information Science, Vol. 142, 2011, pp 187.

²⁴ Brown C.: *Technical analysis for the trading professional*, Second Edition: Strategies and techniques for today's turbulent global financial markets. The McGrawHill Companies, New York 2011, p. 223.

²⁵ *Knight Shows How to Lose \$440 Million in 30 Minutes*. <http://www.bloomberg.com/bw/articles/2012-08-02/knight-shows-how-to-lose-440-million-in-30-minutes>, dostęp: 10.10.2016



W celu opracowania precyzyjnej prognozy kursów akcji zdobywane są informacje o firmach w elektronicznym formacie tekstowym, które są dostępne z wielu źródeł, w tym od komercyjnych dostawców, takich jak *Bloomberg*, publicznych serwisów informacyjnych, czy serwisów społecznościowych, takich jak *Twitter*. Aplikacje mogą identyfikować nazwy firm, słowa kluczowe, a także semantykę wiadomości, dzięki czemu możliwe jest badanie tzw. sentymentu wiadomości (informacja pozytywna, negatywna, neutralna).²⁶

Na Warszawskiej Giełdzie Papierów Wartościowych platformę informatyczną *Warset* z 2000 roku zastąpiono w 2013 roku systemem obsługi zleceń giełdowych UTP (ang. *Universal Trading Platform*) amerykańskiej firmy *NYSE Euronext*. Ze względu na handel algorytmiczny w zakresie instrumentów pochodnych platforma UTP powinna spełniać oczekiwania inwestorów krajowych. Wydajność systemu to 20 tysięcy zleceń na sekundę, a dodanie serwera *Trading Unit* pozwala na zwiększenie wydajności systemu o 10 tys. zleceń/s. Opóźnienie odpowiedzi między momentem złożenia zlecenia a otrzymaniem informacji o jego obsłudze to 150 mikrosekund. Możliwe jest zrealizowanie 15 milionów zleceń w trakcie sesji giełdowej. Platforma cechuje się wysoką dostępnością, a w przypadku poważnej awarii możliwa jest kontynuacja notowań. W algorytmie realizacji zleceń o kolejności realizacji zlecenia decyduje najpierw cena, a w przypadku zleceń z tym samym limitem ceny najpierw realizowane jest zlecenie, które wpłynęło wcześniej.

Platforma umożliwia działanie animatorów rynków giełdowych w ramach *Systemu Animatora Rynku* (SAR). W szczególności system UTP CDE przeznaczony jest dla instrumentów rynku kasowego, derywatów, a także produktów ETP (ang. *Exchange Trading Products*), takich jak instrumenty strukturyzowane, czy warranty. Aby dany instrument był notowany w tym segmencie, konieczne jest posiadania animatora tego instrumentu na rynku, który je kwotuje. Tylko jeden podmiot może pełnić funkcję animatora dla danego instrumentu. Brak kwotowań animatora skutkuje zatrzymaniem handlu tym instrumentem.

2. Programowanie genetyczne w systemach finansowych

Programowanie genetyczne, oprócz sztucznych sieci neuronowych, jest istotnym paradygmatem sztucznej inteligencji. Programowanie genetyczne umożliwia automatyczne wytwarzanie programów, które w szczególności mogą wyznaczać decyzje inwestycyjne.²⁷ Popularnym podejściem jest wykorzystanie programowania genetycznego w połączeniu z analizą techniczną.²⁸ Pozwala ono na automatyczne i dynamiczne przewidywanie kursów oraz generowanie wskazówek kupna/sprzedaży wg strategii *market timing* w krótkim odcinku czasu. W odróżnieniu od strategii inwestycyjnej *buy and hold*, zakładającej długotrwałe posiadanie aktywów, strategia *market timing* pozwala na wykorzystanie krótkoterminowych fluktuacji rynku w przypadku, gdy rynek jest stabilny lub gdy odnotowywano spadki.²⁹

Interesującym obszarem, w którym zastosowano programowanie genetyczne jest predykcja rozwoju firm. Prognoza bankructwa firmy może mieć istotny wpływ na inwestorów.³⁰ *Etemadi*,

²⁶ Bechler A.: *Porównanie efektywności sieci neuronowych i modeli ekonometrycznych we wspomaganiu decyzji kredytowych*, StatSoft Polska, Kraków 2003, s. 37

²⁷ Schwaerzel R.: *Financial time series prediction and evaluation by genetic programming with trigonometric functions and high-order statistics*. Ph.D. Dissertation. The University of Texas at San Antonio, 2006, p. 227

²⁸ Svargard N.; Nordin P.; Lloyd S.; Wihlborg C.: *Evolving short-term trading strategies using genetic programming*. Proc. of the Congress on Evolutionary Computation, vol. 2, 2002, p. 2007.

²⁹ Potvin J.-Y., Soriano P., Vallée M.: *Generating trading rules on the stock markets with genetic programming*, Computers & Operations Search, Vol. 31, No. 7, June 2004, p. 1033

³⁰ Winiarski J.: *Ryzyko w działalności gospodarczej przedsiębiorstw*. Wyd. Uniwersytet Gdański. Instytut Transportu i Handlu Morskiego, Gdańsk 2014, s. 12.

Rostamy i *Dehkordi* porównali modele, z których model oparty o programowanie genetyczne pozwolił na uzyskanie lepszych prognoz rozwoju firm notowanych na giełdzie w *Teheranie*.³¹

2.1. Jakość oprogramowania a koniunktura na giełdzie

Współczesne giełdy papierów wartościowych wymagają wydajnych, niezawodnych i bezpiecznych platform informatycznych działających w czasie rzeczywistym. Transakcje o wysokiej intensywności mogą być zawierane kilka tysięcy razy na minutę przez jednego inwestora. W szczególności firma brokerska może realizować nawet kilkadziesiąt tysięcy transakcji na sekundę, a animatorzy rynków giełdowych mają dostęp do kilkuset platform giełdowych jednocześnie. Oprogramowanie animatora rynku giełdowego powinno zatem być w stanie komunikować się i prowadzić transakcje giełdowe w imieniu wielu brokerów i inwestorów indywidualnych na kilkuset platformach giełdowych. W tej sytuacji należy zastosować programowanie genetyczne do samoorganizacji tej klasy rozproszonych systemów informatycznych.

Błąd w oprogramowaniu inwestora, brokera lub animatora rynków giełdowych może doprowadzić do ogromnych strat, o czym przekonała się firma *Knight Capital Group* o wartości rynkowej 296 milionów USD, która straciła 462 miliony USD w ciągu 45 minut. Wcześniej firma czterokrotnie zwiększyła przychody w ciągu roku, a jej oprogramowanie cieszyło się wysoką reputacją wśród fachowców z branży. 1 sierpnia 2012 roku między 9.30 a 10.15 rano komputery firmy *Knight* przeprowadziły nieudane transakcje giełdowe na 150 platformach giełdowych jednocześnie. Szczególnie intensywnie handlowano papierami na platformach: *Berkshire Hathaway*, *Nokia* oraz *Exelon*. Na giełdzie *Exelon* tracono około 15 centów na parze transakcji kupno/sprzedaż, które przeprowadzano z intensywnością 2 400 transakcji na minutę.

System z ogromną szybkością kupował akcje zbyt wysoko, a sprzedawał zbyt nisko. Usterka doprowadziła do zrealizowania dodatkowo 4 mln niekorzystnych transakcji dotyczących 550 mln akcji. Wcześniej tego rodzaju niekorzystne transakcje nie były realizowane przez system rozproszony *Knight*, który regularnie realizował zlecenia dla uznanych brokerów, np. *E*Trade Financial* czy *Fidelity Investments*. W czerwcu 2012 roku średnia wartość transakcji zrealizowanych w ciągu dnia przez system *Knight* wyniosła 19,56 mld dla 3,1 mld akcji. Do sierpnia 2012 roku rozproszony system obsługi zleceń firmy *Knight Capital Group* uznawany był za jeden z najbardziej bezpiecznych i wydajnych w branży.

Co było przyczyną ogromnych strat poniesionych przez uznanego animatora giełd papierów wartościowych? Jedną z przyczyn był błąd administratora systemu, który 27 lipca 2012 roku wprawdzie zainstalował na 7 serwerach SMARS oprogramowanie *Retail Liquidity Program RLP* firmy *NYSE Euronext*, ale w ósmym serwerze SMARS nie wykonał tej instalacji. Instalacja dodatkowego oprogramowania *RLP* była konieczna, aby system *Knight* mógł realizować transakcje na giełdzie nowojorskiej. Serwery SMARS automatycznie zbierały zlecenia (ang. *parent orders*) od brokerów i od inwestorów indywidualnych, a następnie generowały zlecenia (ang. *child orders*) do platform giełdowych z wysoką częstotliwością.

Kod *RLP* powinien zastąpić niewykorzystywany moduł wewnątrz rutera zleceń, którym była funkcjonalność testowa *Power Peg* zaprojektowana kilkanaście lat wcześniej. Moduł *Power Peg* dedykowany był do testowania systemu *Knight* w laboratoryjnym rozproszonym systemie giełdowym i weryfikował sytuację odwrotną do stosowanej zazwyczaj przez inwestorów, w której zawyżano cenę akcji kupowanej oraz zmniejszano cenę akcji sprzedawanej. Moduł ten powinien być usunięty znacznie wcześniej przez testera systemu, a nie został – to drugi poważny błąd człowieka.

Ale to nie koniec feralnego splotu okoliczności. Aby wywołać moduł *RLP* administrator wykorzystał flagę, którą wcześniej stosowano przy aktywacji *Power Peg*. Po ustawieniu flagi

³¹ Etemadi H., Rostamy A., Dehkordi H., *A genetic programming model for bankruptcy prediction: Empirical evidence from Iran*, Expert Systems with Applications, Vol. 36, No. 2, March 2009, pp. 3199.

na „yes” wywołany powinien być moduł *RLP* w nowej wersji aplikacji *Knight*. Natomiast w starszej wersji wywołanie to odnosiło się do modułu testowego. To właśnie ten trzeci błąd człowieka doprowadził do błędnych decyzji zakupu/sprzedaży akcji na giełdzie nowojorskiej przez ósmy serwer SARS ze starszą wersją oprogramowania.

1 sierpnia 2012 roku o 8.00 rano serwery SARS otrzymały zlecenia od brokerów dotyczące transakcji na giełdzie nowojorskiej, która rozpoczynała działanie o godz. 9.30. Siedem serwerów przygotowało poprawne zlecenia transakcji na platformę giełdową. Przezornie wysłały 97 wiadomości do pracowników firmy o identyfikacji błędu „*Power Peg disabled*”. Nikt jednak z pracowników funkcyjnych nie zareagował, a wyjaśnieniem przyczyny wysłania komunikatu zajęli się administratorzy systemu. Warto podkreślić, że oprogramowanie animatorów rynku, to bardzo złożone aplikacje. To czwarty błąd człowieka – mimo, że wstrzymanie działania oprogramowania do wyjaśnienia sytuacji również było niekorzystne ze względu na obroty rzędu 20 mld USD, to spowodowałoby znacznie mniejsze straty niż kontynuowanie realizacji zleceń.

W tym czasie ósmy serwer uaktywnił testowy moduł *Power Peg*, który przygotował zlecenia po niekorzystnej cenie, a administratorzy systemu usiłowali poradzić sobie z naprawą systemu. W jednej z prób wycofano moduł *RLP* z nowszej wersji oprogramowania, aby wrócić do wersji z modułem *Power Peg* poprawnie działającej w poprzednich dniach – to już piąty błąd człowieka. W rezultacie 8 serwerów SARS przeprowadziło niekorzystne transakcje od godz. 9.30, co prowadziło do ogromnych strat w tempie 172 tys. USD na sekundę.

Firma *Knight Capital Group* z *New Jersey* została przejęta przez *Global Electronic Trading Company* z *Chicago* w 2013 roku, a utworzony holding *KCG Holdings Inc.* z siedzibą w *New Jersey* cechował się wartością rynkową na poziomie 1,5 mld USD i zatrudniał ok. 1000 osób w 2014 roku. Wartość informatycznej platformy transakcji giełdowych *KCG Hotspot* wyceniana jest na 365 mln USD. Przychody holdingu w 2015 roku wyniosły 1 198 mln USD, w tym przychody netto to 836 mln USD.

Błędy ludzkie oraz niska jakość oprogramowania, to dwie zasadnicze przyczyny niepowodzenia systemu wymiany papierów wartościowych i upadku firmy *Knight Capital Group*. Administratorzy błędnie postawili diagnozę nieprawidłowego działania systemu, a oprogramowanie nie było wcześniej rzetelnie przetestowane oraz zaktualizowane, mimo, że liczba serwerów wynosiła 8, a nie setki tysięcy. Czy zatem istnieje możliwość poradzenia sobie z tej klasy problemami w tak złożonych systemach? Odpowiedź brzmi „tak”, a jednym ze sposobów jest zastosowanie programowania genetycznego do automatycznego testowania i modyfikacji oprogramowania napisanego przez człowieka. Przecież inteligentny system informatyczny inwestora, brokera lub animatora rynku powinien się autonomicznie naprawiać, organizować, optymalizować i działać prawidłowo mimo uszkodzeń. Można oczywiście zwiększać liczbę administratorów, programistów i testerów, ale są jakieś rozsądne ograniczenia zatrudnienia w firmie liczącej 1 000 pracowników.

Warto zatem zwięźle scharakteryzować programowanie genetyczne, które należy do metaheurystyk ewolucyjnych operujących na populacji programów. Elementami populacji są kompaktowe programy komputerowe, które mają za zadanie rozwiązać wybrany problem (rys. 6).³²

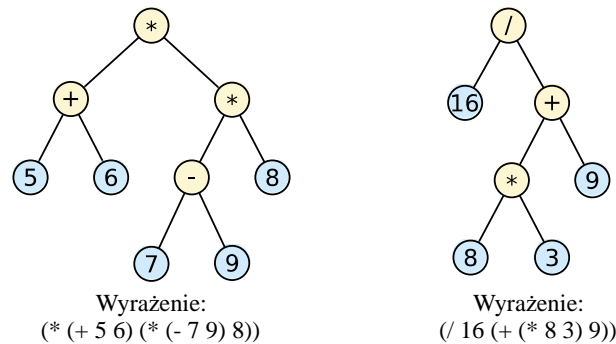
³³ Programy konstruuje się, stosując procedury ze zdefiniowanego zbioru instrukcji \mathcal{F} oraz terminale ze zbioru danych \mathcal{T} .³⁴ Jednym ze sposobów reprezentacji programów komputero-

³² Koza J. R., et al., *Genetic programming IV. Routine human-competitive machine intelligence*, Kluwer Academic Publishers, New York, 2003, p. 230

³³ Bartz-Beielstein T., Branke J., Mehnen J., Mersmann O., *Evolutionary Algorithms*, WIREs Data Mining Knowledge Discovery, vol. 4, John Wiley & Sons, New York, 2014, s. 178.

³⁴ Koza J. R., *op.cit.*, p. 275

wych są drzewa wyrażeń (rys. 6). Liście drzew odpowiadają elementom ze zbioru terminali, a węzły wewnętrzne – elementom ze zbioru procedur.³⁵

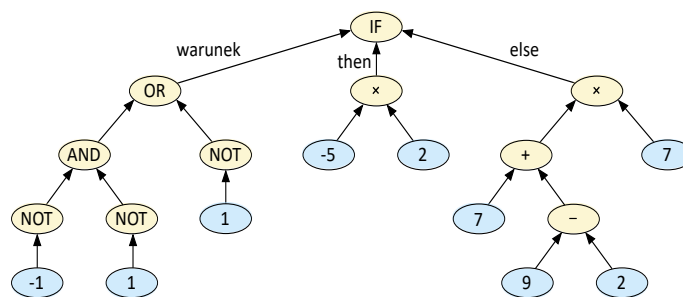


Rysunek 6. Przykłady drzew i odpowiadających im wyrażeń w notacji prefiksowej

Źródło: Opracowanie własne.

Zgodnie z *warunkiem zgodności* procedury muszą akceptować w roli argumentów wszystkie terminale oraz wszystkie wartości, jakie może zwrócić dowolna z procedur. Jest to istotne, ponieważ konstrukcja drzew odbywa się w sposób losowy, podobnie jak późniejsze operacje krzyżowania i mutacji. Spełnienie warunku zgodności gwarantuje, że wytwarzane programy będą działały poprawnie.³⁶ W programowaniu genetycznym zazwyczaj nie korzysta się ze standardowych instrukcji warunkowych i pętli w językach programowania, a przygotowuje się je niezależnie ze względu na warunek zgodności. Z tego powodu zbiór \mathcal{F} można rozszerzyć o trójargumentową procedurę warunkową *IF*. Pierwszy argument interpretowany jest jako warunek logiczny, przy czym warunki złożone można konstruować za pomocą procedur *OR*, *AND* oraz *NOT*. Drugi argument procedury *IF* reprezentuje wartość zwracaną, gdy warunek jest prawdziwy, a trzeci argument – wartość zwracaną dla fałszywego warunku (rys. 7).

Zbiór może zostać również rozszerzony o dwuargumentową procedurę pętli *WHILE*. Pierwszy argument reprezentuje warunek kontynuacji pętli, który interpretuje się tak samo, jak w przypadku procedur logicznych. Drugi argument reprezentuje ciało pętli wykonywane dopóki warunek kontynuacji nie zmieni wartości z logicznej prawdy na fałsz. Aby taka zmiana była możliwa, konieczne jest wprowadzenie procedur do obsługi zmiennych programu w drzewie. W przeciwnym razie wystąpić mogą pętle nieskończone.



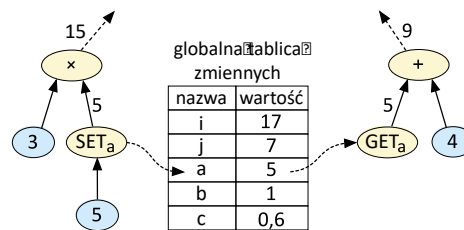
Rysunek 7. Przykład instrukcji warunkowej w drzewie programu

Źródło: Opracowanie własne.

³⁵ Poli R., Langdon W.B., McPhee N.F., *A Field Guide to Genetic Programming*, <http://www.gp-field-guide.org.uk>, dostęp: 4 września 2016.

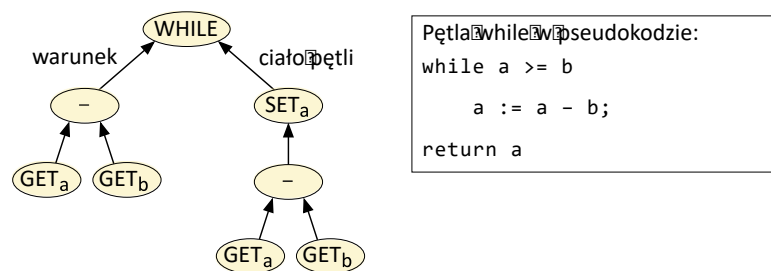
³⁶ Koza J. R., *ibid.*, p. 223.

Dla każdej zmiennej programu $var_1, \dots, var_d, \dots, var_D$ wprowadza się procedurę GET_{var_d} służącą do odczytu jej wartości oraz procedurę SET_{var_d} do aktualizacji wartości. Warunkiem koniecznym, aby pętla była skończona jest obecność procedury GET_{var_d} w wyrażeniu na warunek kontynuacji i procedury SET_{var_d} w ciele pętli. W ten sposób możliwa jest zmiana wartości warunku w wyniku wykonania ciała pętli. Zasadę działania procedur obsługi zmiennych przedstawiono na rysunku 8, a przykład drzewa programu zawierającego pętlę - na rysunku 9.



Rysunek 8. Zasada działania procedur obsługi zmiennych programu w drzewie

Źródło: Opracowanie własne.



Rysunek 9. Drzewo programu wyznaczającego resztę z dzielenia zmiennych a i b

Źródło: Opracowanie własne.

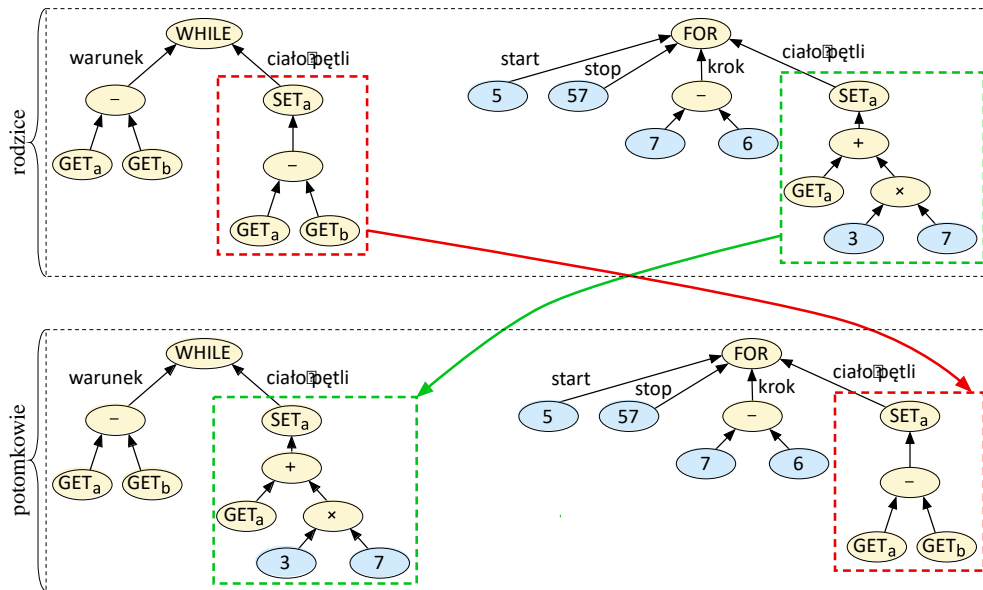
Posługując się rozszerzonym zbiorem procedur w następującej postaci: $\mathcal{F} = \{+, -, *, /, AND, OR, NOT, WHILE, SET_{var_1}, \dots, SET_{var_D}, GET_{var_1}, \dots, GET_{var_D}\}$, można automatycznie wytwarzać złożone programy do samoorganizacji systemów finansowych. Osobniki populacji początkowej budowane są w sposób losowy.³⁷ Następnie poddaje się je operacjom genetycznym, a zmodyfikowane programy składają się na populację potomków.

Operacje krzyżowania i mutacji definiuje się w kontekście struktury drzew programów.³⁸ Krzyżowanie polega na wymianie fragmentów poddrzew pomiędzy dwoma wybranymi osobnikami (rodzicami), co zobrazowano na rysunku 10. Wybór rodziców opiera się o oceny przypisane osobnikom z funkcji sprawności. Zazwyczaj stosuje się selekcję ruletkową.

Mutacja osobnika może przebiegać na dwa sposoby. W przypadku mutacji pojedynczej, którą zobrazowano na rysunku 11, wybiera się losowy węzeł drzewa, a następnie zastępuje go innym węzłem. Jeśli wybrano węzeł wewnętrzny reprezentujący procedurę, w jego miejsce umieszcza się inną procedurę o takiej samej liczbie argumentów. W przypadku wyboru liścia reprezentującego terminal zastępuje się go innym terminalem.

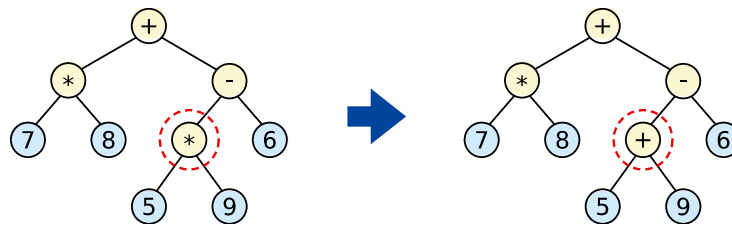
³⁷ Luke S., Panait L., *A survey and comparison of tree generation algorithms*, w: Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001), 2001, s. 81–88.

³⁸ Koza J. R., *ibid.*, p. 323.



Rysunek 10. Przykład krzyżowania dwóch drzew programów

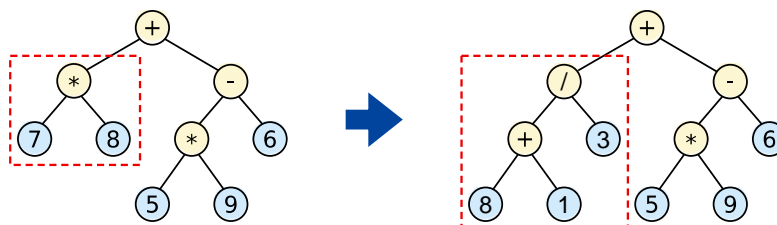
Źródło: Opracowanie własne.



Rysunek 11. Przykład mutacji pojedynczej

Źródło: Opracowanie własne.

Mutacja poddrzewa polega na wybraniu losowego węzła, a następnie usunięciu poddrzewa o korzeniu w tym węźle i zastąpieniu go nowym losowo skonstruowanym poddrzewem³⁹. Ten typ mutacji zobrazowano na rysunku 12.

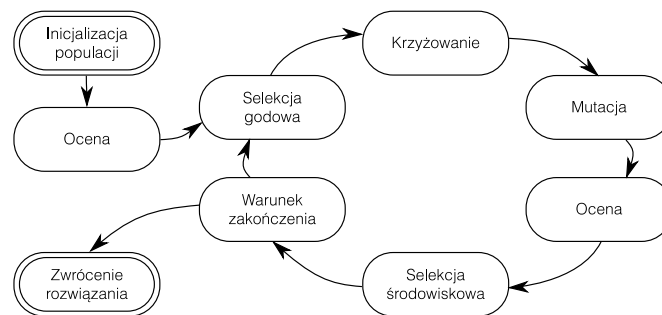


Rysunek 12. Przykład mutacji poddrzewa

Źródło: Opracowanie własne.

³⁹ Angeline P. J., *Subtree crossover: Building block engine or macromutation?*, Genetic Programming 1997: Proceedings of the Second Annual Conference, Stanford University, CA, USA, Morgan Kaufmann, 1997, s. 9.

Po wykonaniu operacji genetycznych nowe osobniki poddaje się ocenie za pomocą funkcji sprawności. Następnie spośród osobników poprzedniej populacji i nowych osobników należy wybrać m rozwiązań zakwalifikowanych do kolejnej epoki na etapie selekcji środowiskowej. W programowaniu genetycznym zazwyczaj stosuje się prostą operację zastąpienia wcześniejszych osobników nowymi rozwiązaniami. Aby uniknąć utraty sprawnych rozwiązań w wyniku tej operacji zastosować można archiwum zewnętrzne, w którym zapisywane są najlepsze spośród dotychczasowych osobników (rys. 13).



Rysunek 13. Schemat programowania genetycznego

Źródło: Opracowanie własne na podstawie⁴⁰.

3. Uczenie maszynowe w aplikacjach programowanych genetycznie

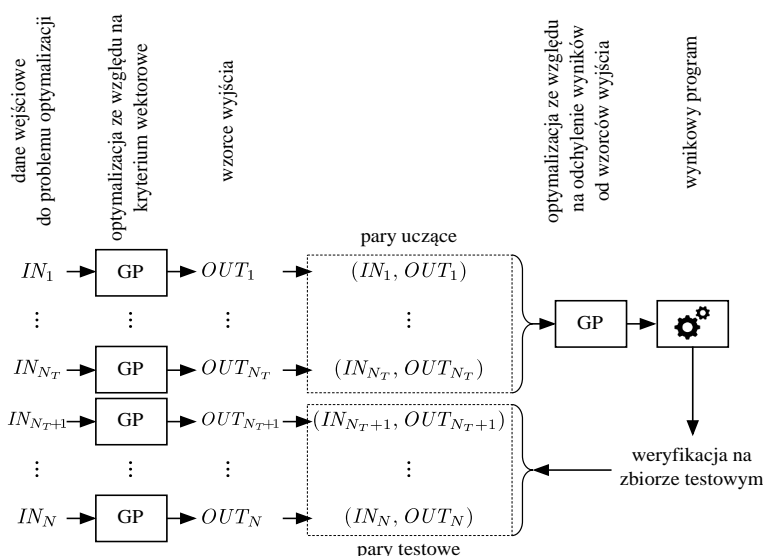
Zasady uczenia maszynowego zbioru programów genetycznych są prawie identyczne jak reguły treningu sieci neuronowych. Programowanie genetyczne pozwala na wyznaczanie rozwiązań sprawnych ze względu na wybrane kryterium skalarne bądź wektorowe w wielu zagadnieniach optymalizacji o istotnych zastosowaniach praktycznych, w tym w zagadnieniach związanych z zarządzaniem zasobami mgieł lub chmur obliczeniowych realizujących operacje finansowe. Konstrukcja kompaktowych programów w wielu zastosowaniach wymaga długotrwałych obliczeń zanim wytworzone aplikacje zwrócą rozwiązania o satysfakcjonujących ocenach dla wybranych funkcji celu. Interesujące w tym kontekście jest wykorzystanie nadzorowanego uczenia maszynowego do klasyfikacji czy predykcji giełdowej.

Do opracowania par uczących można wykorzystać programowanie genetyczne realizujące optymalizację ze względu na wybrane kryterium wektorowe. Zestaw danych wejściowych do zagadnienia optymalizacji reprezentuje wejście do programowania genetycznego, a wyznaczone rozwiązanie jest wzorcem wyjścia dla pary uczącej. Uzyskane pary stosowane są w drugiej turze programowania genetycznego, w której funkcja celu jest funkcją odchylenia wyznaczanych rozwiązań od wzorców wyjścia z par treningowych. Koncepcję zastosowania uczenia maszynowego w programowaniu genetycznym przedstawiono na rysunku 14.

Omówione podejście pozwala na automatyczne zaimplementowanie programu, który wyznacza wysokiej jakości rozwiązania dla wielu zestawów danych wejściowych. Przeprowadzana w ostatnim kroku weryfikacja na zbiorze walidacyjnym umożliwia wykrycie przeuczenia lub niedouczenia programu. Opracowanie zbioru par treningowych wymaga jednak wielokrotnego uruchomienia programowania genetycznego w celu wyznaczenia wzorców wyjścia. Aby program uzyskał zdolność do uogólniania wyznaczanych rozwiązań konieczny jest zbiór treningowy obejmujący wiele różnorodnych instancji. W zależności od rozpatrywanego zagadnienia czas przygotowania par uczących może sięgać nawet kilku tysięcy godzin na klasycznych kom-

⁴⁰ De Jong K. A., *Evolutionary Computation: a Unified Approach*, MIT Press, Cambridge, 2006, p. 226.

puterach. Niezbędne zatem jest wykorzystanie wirtualnych superkomputerów w celu redukcji czasu przygotowania danych i treningu do sekund lub nawet milisekund.



Rysunek 14. Uczenie maszynowe w programowaniu genetycznym do optymalizacji

Źródło: Opracowanie własne.

4. Zarządzanie zasobami samoorganizujących się chmur obliczeniowych

Inteligentne systemy klasy *IBM Watson IoT* czy przyszłościowe platformy giełdowe typu *UTP* wymagają wprowadzenia mechanizmów samoorganizacji. Dlatego ważnym obszarem zastosowań programowania genetycznego jest wytworzenie programów do optymalizacji przydziału zasobów sprzętowych i programistycznych w systemach rozproszonych ze szczególnym uwzględnieniem mgieł i chmur obliczeniowych.⁴¹ Wykorzystanie obliczeń w chmurze pozwala na bardziej efektywne spożytkowanie dostępnych zasobów i ułatwia ich alokację do obsługi nowych usług wprowadzanych przez dynamicznie rozwijające się przedsiębiorstwa. Część firm rezygnuje jednak z tej klasy systemów w obawie przed umieszczeniem wrażliwych danych u zewnętrznego usługodawcy, którego infrastruktura jest współdzielona z innymi podmiotami.

W tym kontekście interesującą alternatywą jest wykorzystanie własnej chmury prywatnej, w której zasoby sprzętowe w postaci serwerów i innych urządzeń kupuje się na własność i instaluje w bezpiecznej lokalizacji pod kontrolą firmy. Następnie konfiguruje się oprogramowanie *OpenStack*⁴². W tym środowisku infrastruktura oferuje elastyczność oraz bezpieczeństwo danych i obliczeń, co doceniły ostatnio: bank *Santander* oraz holding finansowy *PayPal*.

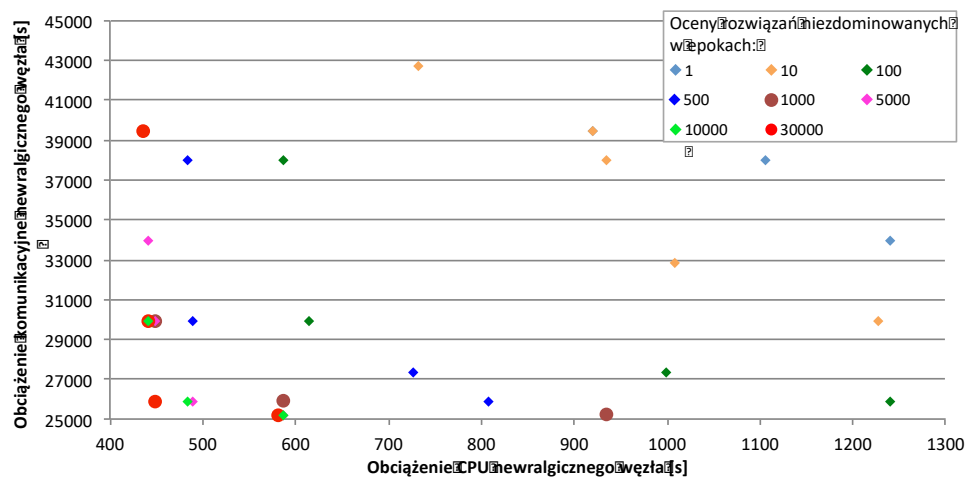
Warto zwrócić uwagę, że poszczególne komponenty aplikacji uruchamianej w chmurze prywatnej w różnym stopniu wykorzystują dostępne zasoby. Przykładowo, komponent bazodanowy zazwyczaj posiada wysokie wymagania w zakresie dostępnej pamięci dyskowej i operacyjnej. Z kolei serwer zasobów rozproszonych cechuje się zwiększonym obciążeniem interfejsów sieciowych, a serwer aplikacji realizujący logikę biznesową – wysokim obciążeniem procesorów komputera. Niezwykle istotny jest właściwy dobór parametrów kupowanych ser-

⁴¹ Estrada T., Wyatt M., Taufer M., *A genetic programming approach to design resource allocation policies for heterogeneous workflows in the cloud*, , IEEE 21st Int. Conf. on Parallel and Distributed Systems, Melbourne, 2015, p. 373.

⁴² *OpenStack*, <https://www.openstack.org/>, dostęp: 4 września 2016.

werów oraz przypisanie komponentów aplikacji do węzłów. Ponadto ważny jest sposób przypisania zadań występujących w systemie do komponentów aplikacji w celu zrównoważenia występujących obciążeń i wyeliminowania wąskich gardeł systemu.⁴³ Przypisanie zadań do komponentów programistycznych, typów komputerów do węzłów chmury prywatnej i komponentów do węzłów określa się mianem *strategii zarządzania chmurą*.

Programowanie genetyczne zastosowano do optymalizacji strategii zarządzania chmurą obejmującą zmodernizowaną wersję systemu rozproszonego *Comcute*, który eksploatowany jest na Politechnice Gdańskiej.⁴⁴ W systemie tym wyróżnia się dwa typy programów w warstwie pośredniczącej, które różnią się obciążeniami.⁴⁵ Przeanalizowano rozbudowę chmury prywatnej do 15 węzłów fizycznych, na których uruchomiono 15 modułów programistycznych pierwszego typu i 30 komponentów drugiego typu. Ponadto uwzględnia się realizację testowego zadania użytkowego, którym może być obsługa zleceń na giełdzie, obejmującego obsługę 10^5 transakcji.⁴⁶ Na podstawie pomiarów obciążeń w rzeczywistym systemie zastosowano programowanie genetyczne do wytworzenia strategicznych programów zarządzających chmurą. Optymalizacji podlegało obciążenie procesorów newralgicznego węzła chmury prywatnej oraz obciążenie interfejsów sieciowych węzła newralgicznego pod względem komunikacji. Ze względu na konflikt między kryteriami uzyskano reprezentację zbioru rozwiązań *Pareto*-optymalnych. Migrację wyznaczonych strategii do brzegu *Pareto* zaprezentowano na rysunku 15.



Rysunek 15. Zbieżność rozwiązań do frontu *Pareto*

Źródło: Opracowanie własne.

Z kolei na rysunku 16 zaprezentowano porównanie ocen strategii wyznaczonych za pomocą programowania genetycznego (algorytm MOGPA) z rozwiązaniami uzyskanymi za pomocą innych metaheurystyk. Do porównania wybrano algorytm ewolucyjny NSGA-II, wielokryte-

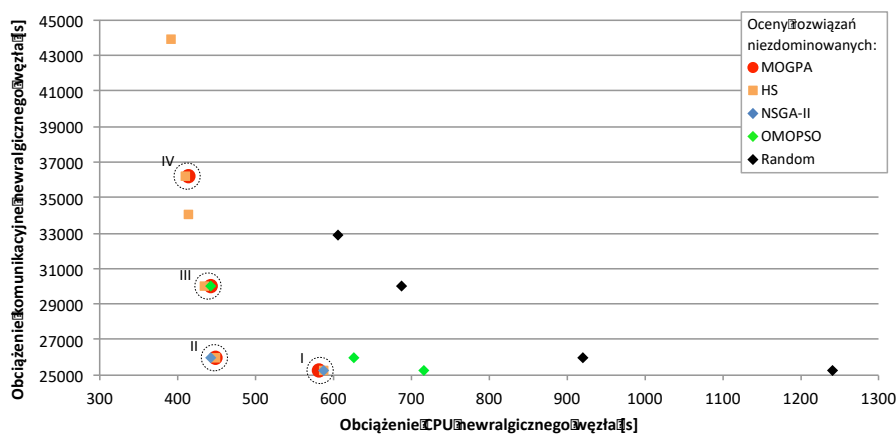
⁴³ Chowdhury A., Tripathi P., *A metrics based analysis of cloud resource management techniques*, International Conference on Advanced Communication Control and Computing Technologies (ICACCCT 2014), Ramanathapuram, 2014, pp. 1632-1636.

⁴⁴ Balicki J., Kuchta J., *Obliczenia rozproszone w systemach komputerowych o architekturze klasy grid*, Wydawnictwo Politechniki Gdańskiej, Gdańsk, 2012.

⁴⁵ Balicki J., Korłub W., Krawczyk H., Paluszak J.: Genetic programming with negative selection for volunteer computing system optimization, The 6th IEEE Int. Conf. on Human System Interaction, Sopot, Poland, June 6-8, 2013, pp. 271.

⁴⁶ Balicki J., Przybyłek P., Zadroga M., Zakidalski M.: Sztuczne sieci neuronowe oraz metoda wektorów wspierających w bankowych systemach informatycznych, *Współczesna Gospodarka*, Vol. 4, 2013, ss. 7.

rialną odmianę algorytmu roju OMOPSO oraz algorytm harmoniczny HS. Na rys. 11 liczbami rzymskimi I-IV oznaczono wspólne obszary zbieżności różnych metaheurystyk. Warto zauważyć, że za pomocą każdego z algorytmów MOGPA, NSGA-II i HS wyznaczono przynajmniej jedno rozwiązanie, które nie jest dominowane przez żadne z rozwiązań innych algorytmów.



Rysunek 16. Porównanie rozwiązań wyznaczonych przez programowanie genetyczne do rozwiązań uzyskanych za pomocą innych metaheurystyk

Źródło: Opracowanie własne.

W obszarze I najlepsze strategie wyznaczono za pomocą programowania genetycznego. Jest to istotne, ponieważ obszar I znajduje się najbliżej teoretycznego punktu idealnego w przestrzeni ocen dla wybranych kryteriów optymalizacji. Oznacza to, że rozwiązania z tego obszaru są rekomendowane do wdrożenia w docelowym systemie. Najwierniejszą reprezentację frontu Pareto uzyskuje się łącząc rozwiązania wyznaczone za pomocą różnych metaheurystyk. Algorytm programowania genetycznego MOGPA wyznacza rozwiązania porównywalnej jakości, jak inne znane z literatury algorytmy oraz uzyskuje przewagę w najistotniejszym obszarze zbieżności.

5. Wybrane zastosowania finansowe samoorganizujących się systemów

Samoorganizujące się systemy rozproszone o wysokiej wydajności umożliwiają implementację skomplikowanych algorytmów sztucznej inteligencji w sferze finansowej. Przykładowo, kluczowym zastosowaniem ewolucyjnych sieci neuronowych jest wyznaczanie wiarygodności kredytowej.⁴⁷ Ze względu na czynniki losowe nie jest możliwa precyzyjna ocena wiarygodności kredytowej. Z tego powodu próbuje się dokonać oceny na podstawie występowania pewnych cech obserwowanych wśród podmiotów spłacających zobowiązania.⁴⁸ Zastosowanie metod ewolucyjno-neuronowych do oceny wiarygodności kredytowej ma wiele zalet, jak chociażby znaczne przyspieszenie procesu kredytowego. Minimalizowana jest również rola czynnika ludzkiego.

⁴⁷ Matuszczyk A., *Credit Scoring*, Wydawnictwo CeDeWu, Warszawa 2012, s. 33.

⁴⁸ Balicki J.: *Wybrane paradygmaty sztucznej inteligencji w informatycznych systemach finansowych*, „Współczesna Gospodarka”, Vol. 6, No. 4, 2015, s. 7.

Ocena wiarygodności kredytowej wyłania tych, którzy najlepiej rokują w zakresie spłacania kredytów, co zazwyczaj występuje także przed wydaniem karty kredytowej.⁴⁹ W metodzie *k*-najbliższych sąsiadów *kNN* potencjalny kredytobiorca jest kwalifikowany w zależności od klasy, która przeważa wśród *k* ocen znajdujących się w jego najbliższym sąsiedztwie.⁵⁰ Do oceny wykorzystuje się także algorytmy ewolucyjne, drzewa decyzyjne oraz systemy eksperckie.⁵¹ Aplikacje oceny wiarygodności kredytowej korzystają także z pojedynczych ANN,⁵² statystycznych modeli dyskryminacyjnych⁵³ oraz modeli logitowych z regresją liniową.⁵⁴ Do oceny wiarygodności kredytowej zastosowano: perceptron wielowarstwowy z radialną funkcją aktywacji, sieć jednokierunkową z liniową funkcją aktywacji oraz sekwencyjne połączenie dwóch ANN, z których pierwsza oblicza wstępny współczynnik wiarygodności kredytowej. Z drugiej strony wspomniane metody można zastosować do wyselekcjonowania kredytobiorców, którzy w najbliższej przyszłości mogą przestać spłacać zaciągnięte zobowiązania.

Do oceny wiarygodności kredytowej w coraz większym stopniu są wykorzystywane media społecznościowe. Serwis *Facebook* opatentował metodę badania zdolności kredytowej opartą na charakterystyce znajomych potencjalnego kredytobiorcy w sieci społecznościowej. Dla znajomych aplikanta kredytowego obliczana jest ich wiarygodność kredytowa. Jeżeli średnia wartość wiarygodności kredytowej znajomych jest poniżej wartości progowej, to wniosek potencjalnego kredytobiorcy powinien zostać odrzucony.⁵⁵

W wypadku kredytów udzielanych osobom fizycznym zazwyczaj danymi wejściowymi do oceny wiarygodności są: wiek, stan cywilny, posiadanie mieszkania lub innej nieruchomości, miesięczny dochód, fakt prowadzenia działalności gospodarczej, informacje o posiadanych zobowiązaniach, liczba dzieci oraz okres zatrudnienia u obecnego pracodawcy. Przy badaniu zdolności kredytowej firm wykorzystuje się wskaźniki finansowe obliczone na podstawie bilansu firmy. Wachlarz cech można wzbogacić – w przypadku, gdy firma jest klientem instytucji finansowej, w której zamierza zaciągnąć zobowiązanie – o analizę dotychczasowych wzorców zachowań finansowych.

Bardzo istotnym zastosowaniem samoorganizujących się systemów uczących jest przewidywanie kryzysów bankowych. Działania prewencyjne w wypadku wystąpienia kryzysu bankowego polegają na udzieleniu bezpośredniej pomocy finansowej, co może się odbić na osłabieniu gospodarki, gdyż plany ratunkowe są zazwyczaj na poziomie kilkunastu procent PKB dla państw rozwijających się oraz dwudziestu kilku procent dla gospodarek rozwiniętych. Znacznie mniejszego budżetu wymagają działania prewencji antykryzysowej, które opierają się na badaniu wypłacalności banków, prognozowaniu kryzysów lub też na wprowadzeniu innowacji technologicznych obniżających koszty operacyjne.⁵⁶ W ten sposób w Szwajcarii udało się uniknąć kryzysu bankowego w 2009 roku.

⁴⁹ Baesens B., Setiono R., Mues C., Vanthien J.: Using neural network rule extraction and decision tables for credit-risk evaluation. *Management Science*, Vol. 49, No. 3, March 2003, p. 312.

⁵⁰ Henley W.E., Hand D.J., *A k-nearest-neighbour classifier for assessing consumer credit risk*, *The Statistician*, Volume 45, Issue 1 (1996), pp. 75 – 95, s. 77.

⁵¹ Yobas M.B., Crook J.N., Ross P., *Credit scoring using neural and evolutionary techniques*, *IMA Journal of Mathematics Applied in Business and Industry*, (2000) 11, pp. 111-125, s. 121.

⁵² Staniec I., *Zastosowanie sztucznych sieci neuronowych i wybranych metod statystycznych do wspomaganie decyzji kredytowych*, *Zastosowania metod statystycznych w badaniach naukowych II*, StatSoft Polska, Kraków 2003, s. 20.

⁵³ Mylonakis J., Diacogiannis G., *Evaluating the Likelihood of Using Linear Discriminant Analysis as a Commercial Bank Card Owners Credit Scoring Model*, *Int. Business Research*, Vol. 3, No. 2, 2010, s. 43.

⁵⁴ Majer I., *Application scoring: logit model approach and the divergence method compared*, Department of Applied Econometrics Working Papers, Working Paper No. 10-06, 2006, s. 27.

⁵⁵ „Facebook patent: Your friends could help you get a loan - or not”, <http://money.cnn.com/2015/08/04/technology/facebook-loan-patent/>, dostęp 6 marca 2016 r.

⁵⁶ Balicka H., Balicki J., Korłub W., Paluszak J., Zadroga M.: Superkomputery do wspomaganie procesów gospodarczych ze szczególnym uwzględnieniem sektora bankowego. *Współczesna Gospodarka*, Vol. 4, Issue 5, 2014, s. 7.

Do interesujących metod prognozowania kryzysu bankowego można zaliczyć jednokierunkową sieć neuronową z propagacją wsteczną i sieć neuronową z radialną funkcją aktywacji. Sieć neuronowa umożliwia także wyznaczanie parametrów w metodzie wektorów nośnych SVM.^{57 58} Metodę SVM zastosowano do oszacowania systemowego ryzyka chińskiego sektora bankowego. Szacując prawdopodobieństwo utraty płynności finansowej banku, uwzględniono wskaźniki ekonomiczne banku takie, jak: *ROE* (ang. *return on equity*), *ROA* (ang. *return on assets*) czy wskaźniki płynności. Ponadto uwzględniono wskaźniki makroekonomiczne, np. wzrost PKB oraz wskaźniki związane z zależnościami od innych banków. Parametry metody SVM, takie jak współczynniki funkcji bazowej, waga przypisana błędom oszacowań ze zbioru uczącego można wyznaczyć za pomocą sztucznych sieci neuronowych.⁵⁹ Za pomocą testu statystycznego *McNemara* dla małej liczebności próby wykazano, że metoda SVM z ANN cechuje się w porównaniu do ANN precyzyjniejszą predykcją z istotnością 0,1.⁶⁰

Zakończenie

Współczesne systemy rozproszone, takie jak: superkomputery, gridy, chmury i mgły obliczeniowe mogą, wykorzystując uczenie maszynowe, podejmować autonomicznie decyzje gospodarcze w ciągu milisekund, co powoduje, że organizacje wprowadzające takie rozwiązania uzyskują przewagę konkurencyjną. Metody sztucznej inteligencji, w tym programowanie genetyczne oraz sztuczne sieci neuronowe otwierają perspektywy do efektywnej predykcji inwestycji giełdowych, oceny zdolności kredytowych klientów indywidualnych i firm, weryfikacji wydawania kart kredytowych, szacowania ratingów i prognoz rozwoju przedsiębiorstw, a nawet ostrzegania przed kryzysami bankowymi i finansowymi.

Metody sztucznej inteligencji również wydobywają wiedzę w systemach medycznych, wojskowych czy zarządzającymi miastami. W najbliższym czasie należy spodziewać się licznych inteligentnych innowacji w transporcie i przemyśle. Wprowadzenie autonomicznych samochodów poprawi komfort jazdy, zmniejszy liczbę wypadków, a nawet zredukuje korki na drogach.

Wielkie projekty badawcze *IBM Watson IoT*, *AlphaGo*, *Blue Brain* czy *Spaun* generują liczne innowacyjne zastosowania w obszarze platform giełdowych, bankowości czy finansów. Wystąpienie *Flash Crash* na giełdzie nowojorskiej oraz milionowe straty platformy animatora giełdowego *Knight Capital Group* obrazują, jak ważną rolę pełni sztuczna inteligencja wbudowana w autonomiczne systemy finansowe, których ludzie nie są już w stanie skutecznie zastąpić, a także szybko naprawiać. Korzystanie z tradycyjnej technologii internetowej to obecnie stanowczo zbyt mało, a bankowość internetowa czy mikroprocesorowe karty zbliżeniowe nie stwarzają przewagi konkurencyjnej w dłuższej perspektywie.

W kraju banki powinny zatem postawić na rozwój mgieł i chmur prywatnych, *Internetu Rzeczy* oraz inteligentnego oprogramowania o wysokiej precyzji klasyfikacji i predykcji. Chyba jednak najtrudniejsze zadanie stoi przed Giełdą Papierów Wartościowych, gdzie należy się spodziewać intensywniejszej modernizacji infrastruktury informatycznej w celu wprowadzenia sztucznej inteligencji do realizacji transakcji wysokiej częstotliwości. Zwiększone są także wymagania odnośnie szybszego udostępnienia obszerniejszego zestawu danych o realizacji zleceń. Oczekiwać należy intensywnej działalności animatorów rynków finansowych i giełd, którzy powinni dysponować wysokiej jakości platformami programistycznymi.

⁵⁷ Larousse D. T.: *Metody i modele eksploracji danych*. Wydawnictwo Naukowe PWN, Warszawa 2008, s. 223.

⁵⁸ Shouwei L., Mingliang W., Jianmin H.: *Prediction of Banking Systemic Risk Based on Support Vector Machine*. *Mathematical Problems in Engineering*, Vol. 2013, April 2013, s. 5.

⁵⁹ Zan H. et al.: *Credit rating analysis with support vector machines and neural networks: a market comparative study*. *Decision Support Systems*, vol. 37, 2004, s. 544

⁶⁰ Shouwei L., Mingliang W., Jianmin H., *op. cit.*, s. 4

Interesującym kierunkiem dalszych badań jest opracowanie nowych modeli sieci neuronowych w mgłach obliczeniowych wykorzystujących *Internet Rzeczy* do prognozowania finansowego. W rezultacie należy się spodziewać poprawy jakości klasyfikacji i prognoz.

Literatura

1. Angeline P. J., *Subtree crossover: Building block engine or macromutation?*, Genetic Programming 1997: Proceedings of the Second Annual Conference, Stanford University, CA, USA, Morgan Kaufmann, 1997, s. 9-17.
2. Baesens B., Setiono R., Mues C., Vanthien J.: *Using neural network rule extraction and decision tables for credit-risk evaluation*. Management Science, Vol. 49, No. 3, March 2003, pp. 312–320.
3. Balicka H., Balicki J., Korłub W., Paluszak J., Zadroga M.: *Superkomputery do wspomagania procesów gospodarczych ze szczególnym uwzględnieniem sektora bankowego*. Współczesna Gospodarka, Vol. 4, Issue 5, 2014, ss. 1-15.
4. Balicki J., Dryja P., Korłub W., Przybyłek P., Tyszka M., Zadroga M., Zakidalski M.: *Metody neuronowe do prognozowania finansowego*, Współczesna Gospodarka, Vol. 7, No. 2, 2016, ss. 21-36.
5. Balicki J., Korłub W., Krawczyk H., Paluszak J.: *Genetic programming with negative selection for volunteer computing system optimization*, The 6th IEEE International Conference on Human System Interaction (HSI), Sopot, Poland, June 6-8, 2013, pp. 271 – 278.
6. Balicki J., Kuchta J. (red.): *Obliczenia rozproszone w systemach komputerowych o architekturze klasy grid*, Wyd. Politechniki Gdańskiej, Gdańsk 2012.
7. Balicki J., Przybyłek P., Zadroga M., Zakidalski M.: *Sztuczne sieci neuronowe oraz metoda wektorów wspierających w bankowych systemach informatycznych*, Współczesna Gospodarka, Vol. 4, 2013, ss. 1-14.
8. Balicki J.: *Wybrane paradygmaty sztucznej inteligencji w informatycznych systemach finansowych*, Współczesna Gospodarka, Vol. 6, No. 4, 2015, ss. 1-23.
9. Bartz-Beielstein T., Branke J., Mehnen J., Mersmann O.: *Evolutionary Algorithms, WIREs Data mining knowledge discovery*, vol. 4, John Wiley & Sons, New York, 2014, s. 178–195.
10. Bechler A.: *Porównanie efektywności sieci neuronowych i modeli ekonometrycznych we wspomaganiu decyzji kredytowych*, Zastosowania metod statystycznych w badaniach naukowych, StatSoft Polska, Kraków 2003.
11. Brown C.: *Technical analysis for the trading professional, Second Edition: Strategies and techniques for today's turbulent global financial markets*. The McGrawHill Companies, New York 2011.
12. Chaveesuk R., Srivaree-Ratana C., Smith A.E.: *Alternative neural network approaches to corporate bond rating*. Journal of Engineering Valuation and Cost Analysis, vol. 2, 1999, ss. 117-131.
13. Chowdhury A., Tripathi P.: *A metrics based analysis of cloud resource management techniques*, International Conference on Advanced Communication Control and Computing Technologies (ICACCCT 2014), Ramanathapuram, 2014, pp. 1632-1636.
14. Davis E. P., Karim D.: *Comparing early warning systems for banking crises*. Journal of Financial Stability, vol. 4, no. 2, 2008, pp. 89–120.
15. De Jong K. A., *Evolutionary Computation: a Unified Approach*, MIT Press, Cambridge, 2006.



16. Demircuc-Kunt A., Detragiache E.: *Monitoring banking sector fragility: a multivariate logit approach*. World Bank Economic Review, vol. 14, no. 2, 2000, pp. 287–307.
17. Eliasmith C. et al.: *A large-scale model of the functioning brain*, Science, Vol. 338, 2012, pp. 1202–1205.
18. Estrada T., Wyatt M., Taufer M.: *A genetic programming approach to design resource allocation policies for heterogeneous workflows in the cloud*, Parallel and Distributed Systems (ICPADS), 2015 IEEE 21st International Conference on, Melbourne, VIC, 2015, pp. 372-379.
19. Etemadi H., Rostamy A., Dehkordi H.: *A genetic programming model for bankruptcy prediction: Empirical evidence from Iran*, Expert Systems with Applications, Vol. 36, No. 2, March 2009, pp. 3199-3207
20. Eurostat. Your key to European statistics. <http://ec.europa.eu/eurostat/web/national-accounts/data/main-tables>, dostęp: 7.10.2016 r.
21. *Facebook patent: Your friends could help you get a loan - or not*, <http://money.cnn.com/2015/08/04/technology/facebook-loan-patent/>, dostęp 6 marca 2016 r.
22. Ferdousi Z., Maeda A.: Anomaly detection using unsupervised profiling method in time series data, Proc. of the 10th Conference on Advances in Databases and Information Systems, Tesaloniki 2006.
23. Frankel J. A., Rose A. K.: *Currency crashes in emerging markets: an empirical treatment*. Journal of International Economics, vol. 41, no. 3-4, 1996, pp. 351–366.
24. Gately E. *Sieci Neuronowe. Prognozowanie finansowe i projektowanie systemów transakcyjnych*. WIG-Press, Warszawa 1999.
25. Hanschel E., Monnin P.: *Measuring and forecasting stress in the banking sector: evidence from Switzerland*. Investigating the Relationship between the Financial and Real Economy, BIS Papers, no. 22, 2005, pp. 431-449.
26. Henley W.E., Hand D.J.: *A k-nearest-neighbour classifier for assessing consumer credit risk*, The Statistician, Vol. 45, Issue 1, 1996, pp. 75 – 95.
27. Kaminsky G. L., Reinhart C. M.: *The twin crises: the causes of banking and balance-of-payments problems*. American Economic Review, vol. 89, no. 3, pp. 473–500, 1999.
28. *Knight Shows How to Lose \$440 Million in 30 Minutes*. <http://www.bloomberg.com/bw/articles/2012-08-02/knight-shows-how-to-lose-440-million-in-30-minutes>, dostęp: 10.10.2016.
29. Koza J. R., et al.: *Genetic programming IV. Routine human-competitive machine intelligence*, Kluwer Academic Publishers, New York, 2003.
30. Larousse D. T.: *Metody i modele eksploracji danych*. Wydawnictwo Naukowe PWN, Warszawa 2008.
31. Majer I.: *Application scoring: logit model approach and the divergence method compared*, Department of Applied Econometrics, Working Paper, No. 10-06, 2006.
32. Matuszczyk A., *Credit Scoring*, Wydawnictwo CeDeWu, Warszawa 2012, s. 32 - 40
33. Mylonakis J., Diacogiannis G.: *Evaluating the likelihood of using linear discriminant analysis as a commercial bank card owners credit scoring model*. International Business Research, Vol. 3, No. 2, 2010.
34. Nazari M., Alidadi M.: *Measuring credit risk of bank customers using artificial neural network*. Journal of Management Research, Vol. 5, No. 2, 2013.
35. Oet M., Eiben R., Bianco T., Gramlich D., Ong S., Wang J.: *SAFE: an early warning system for systemic banking risk*. Proc. of the 24th Australasian Finance and Banking Conference, SSRN, 2011.

36. OpenStack, <https://www.openstack.org/>, dostęp: 4 września 2016.
37. Pandey V., Wee-Keong Ng, Ee-Peng Lim: *Financial advisor agent in a multi-agent financial trading system*. Proc. 11th Int. Workshop on Database and Expert Systems Applications, 2000, pp. 482-486.
38. Pietrzak E., Markiewicz M. (red.): *Finanse, bankowość i rynki finansowe*, Wyd. Uniwersytetu Gdańskiego, Gdańsk 2007.
39. Pietrzak J.: *Czynniki przewagi konkurencyjnej na rynku bankowych usług detalicznych*, Wyd. Uniwersytetu Gdańskiego, Gdańsk 2002.
40. Poli R., Langdon W.B., McPhee N.F.: *A Field Guide to Genetic Programming*, <http://www.gp-field-guide.org.uk>, dostęp: 4 września 2016.
41. Potvin J.-Y., Soriano P., Vallée M.: *Generating trading rules on the stock markets with genetic programming*, Computers & Operations Search, Vol. 31, No. 7, June 2004, pp. 1033-1047
42. Rout M., Majhi B., Majhi R., Panda G.: *Novel stock market prediction using a hybrid model of adaptive linear combiner and differential evolution*. Communications in Computer and Information Science, Vol. 142, 2011, pp 187-191.
43. Schwaerzel R.: *Financial time series prediction and evaluation by genetic programming with trigonometric functions and high-order statistics*. Ph.D. Dissertation. The University of Texas at San Antonio, 2006.
44. Sensmeier L.: *Building Stock Trading Strategies: 20% Faster with Hadoop*. <http://hortonworks.com/blog/building-stock-trading-strategies-20-faster-with-hadoop/>, dostęp: 10.10.2016r.
45. Shouwei L., Mingliang W., Jianmin H.: *Prediction of banking systemic risk based on Support Vector Machine*. Mathematical Problems in Engineering, Vol. 2013, April 2013, pp. 1-15.
46. Srivastava R. P.: *Automating judgmental decisions using neural networks: a model for processing business loan applications*, Proc. of the 1992 ACM Annual Conference on Communications, pp. 351-357.
47. Staniec I.: *Zastosowanie sztucznych sieci neuronowych i wybranych metod statystycznych do wspomaganie decyzji kredytowych*, Zastosowania metod statystycznych w badaniach naukowych II, StatSoft Polska, Kraków 2003, s. 20.
48. Statlog (German Credit Data) Data Set, <http://archive.ics.uci.edu/ml/datasets/Statlog+%28German+Credit+Data%29>, dostęp: 2 września 2016 r.
49. Svangard N.; Nordin P.; Lloyd S.; Wihlborg C.: *Evolving short-term trading strategies using genetic programming*. Proc. of the Congress on Evolutionary Computation, vol. 2, 2002, pp. 2006-2010.
50. *The "Big Data" Solution For Wall Street*. Stock Forecast Based On a Predictive Algorithm. <http://iknowfirst.com/the-big-data-solution-for-wall-street>, dostęp: 10.10.2016r.
51. Winiarski J.: *Ryzyko w działalności gospodarczej przedsiębiorstw*. Wyd. Uniwersytet Gdański. Instytut Transportu i Handlu Morskiego, Gdańsk 2014.
52. Yobas M.B., Crook J.N., Ross P.: *Credit scoring using neural and evolutionary techniques*. Journal of Mathematics Applied in Business and Industry, Vol. 11, 2000, pp. 111-125.
53. Zan H. et al.: *Credit rating analysis with support vector machines and neural networks: a market comparative study*. Decision Support Systems, vol. 37, 2004, ss. 543–558.

MACHINE LEARNING FOR THE SELF-ORGANIZATION OF DISTRIBUTED SYSTEMS IN ECONOMIC APPLICATIONS

Summary

In this paper, an application of machine learning to the problem of self-organization of distributed systems has been discussed with regard to economic applications, with particular emphasis on supervised neural network learning to predict stock investments and some ratings of companies. In addition, genetic programming can play an important role in the preparation and testing of several financial information systems. For this reason, machine learning applications have been discussed because some software applications can be automatically constructed by genetic programming. To obtain a competitive advantage, machine learning can be used for the management of self-organizing cloud computing systems performing calculations for business. Also the use of selected economic self-organizing distributed systems has been described, including some testing methods of predicting borrower reliability. Finally, some conclusions and directions for further research have been proposed.

Keywords: prediction of the stock market, smart financial systems, ratings of companies, genetic programming, artificial neural networks.

prof. UW, dr hab. Jerzy Balicki
Politechnika Warszawska, Wydział Matematyki i Nauk Informacyjnych
Zakład Strukturalnych Metod Przetwarzania Wiedzy
ul. Koszykowa 75
00-662 Warszawa
j.balicki@mini.pw.edu.pl

mgr inż. Waldemar Korłub
Politechnika Gdańska, Wydział Elektroniki, Telekomunikacji i Informatyki
Katedra Architektury Systemów Komputerowych
ul. G. Narutowicza 11/12
80-233 Gdańsk
waldemar.korlub@pg.gda.pl

