Uniwersytet Śląski
Wydział Informatyki i Nauki o Materiałach
Instytut Informatyki

Rozprawa doktorska

mgr Barbara Probierz

Automatyczna kategoryzacja wiadomości elektronicznych z zastosowaniem sieci społecznych oraz algorytmów mrowiskowych

Promotor: dr hab. Urszula Boryczka
Promotor pomocniczy: dr Jan Kozak

Sosnowiec, 2017 r.

Najbliższym...

Serdeczne podziękowania
Pani dr hab. Urszuli Boryczce oraz Panu dr Janowi Kozakowi za wielką pomoc, wyrozumiałość i przede wszystkim okazaną cierpliwość podczas pisania pracy.

Spis treści

0

Wstęp

Wiadomość elektroniczna to jedna z najbardziej popularnych w dzisiejszych czasach metoda komunikacji. Jest łatwo dostępna, szybka, tania, a jednocześnie pozwala komunikować się z wieloma osobami oraz na duże odległości bez konieczności wychodzenia z domu czy pracy. Zwykłe listy dostarczane są po kilku dniach lub nawet tygodniach, natomiast wiadomość e-mail przesyłana jest do konkretnego adresata, nawet znajdującego się wiele kilometrów od nas, dosłownie w kilka sekund. Poczta elektroniczna zazwyczaj jest usługą bezpłatną, natomiast rozmowa telefoniczna zwłaszcza zagraniczna może generować duże koszty. Podobnie jest z wysyłaniem przesyłki pocztą tradycyjną. Dodatkowym atutem jest to, że wysyłając e-mail nie trzeba sprawdzać, która jest godzina po drugiej stronie globu, gdyż odbiorca może przeczytać wiadomość w dowolnie wybranym przez siebie momencie, wtedy gdy ma na to czas i ochotę.

Wiadomość elektroniczna jako narzędzie komunikacji może być wykorzystana do różnych celów. Najczęściej jest to chęć przekazania komunikatu, zareklamowania produktu, poinformowania o promocji, przesłania dokumentów czy też utrzymania kontaktu z klientami. To co jest napisane w wiadomościach nie tylko przekazuje konkretne treści, ale też jest swego rodzaju wizytówką i najlepiej świadczy o poziomie edukacji i wychowania nadawcy.

W dzisiejszych czasach media i komunikacja społeczna to jeden z najważniejszych czynników i sposobów kształtowania współczesnej rzeczywistości politycznej, społecznej czy ekonomicznej. Komunikacja przez Internet stanowi niezwykle prosty, a zarazem wygodny sposób przekazywania informacji. Obecnie każdego dnia przez Internet wysyłanych jest miliardy wiadomości, a w ciągu roku jest ich ponad 100 bilionów. Typowy użytkownik dostaje przeciętnie 40-50 wiadomości e-mail każdego dnia. Niektórzy otrzymują ich nawet setki dziennie, przez co użytkownicy znaczną część swojego czasu pracy poświęcają na czytanie i odpowiadanie na otrzymane wiadomości e-mail. Jednocześnie znaczna część przesyłanych wiadomości zawiera informacje zbędne, które powinny być filtrowane.

Jednym z największych problemów użytkowników, zwłaszcza tych, dla których e-mail to podstawa komunikacji, jest odpowiednie uporządkowanie poczty elektronicznej i przypisanie wiadomości do poszczególnych folderów. Zwłaszcza, gdy kategoryzacja ta ma się odbywać w sposób automatyczny. Z tego powodu jest coraz większe zainteresowanie tworzeniem systemów, które w sposób automatyczny mogą pomóc użytkownikom w zarządzaniu pocztą elektroniczną.

Niestety problem automatycznego kategoryzowania wiadomości e-mail jest problemem bardzo spersonalizowanym, gdyż zależy od indywidualnych upodobań poszczególnych użytkowników. Odzwierciedlenie tych upodobań można przedstawić w postaci sieci społecznych, których analiza pozwala lepiej zrozumieć zachowania użytkowników poczty elektronicznej. Natomiast poprzez zastosowanie algorytmów mrowiskowych możliwe jest poszukiwanie większej części przestrzeni rozwiązań z zastosowaniem eksploracji i eksploatacji.

Proces przypisania wiadomości do folderów (tzw. e-mail Foldering Problem) jest problemem złożonym, gdyż automatyczna metoda kategoryzacji może się sprawdzić u jednego użytkownika, a u innego może prowadzić do błędów. Częstym powodem takiej sytuacji jest to, że użytkownicy tworzą nowe foldery, a także przestają korzystać z niektórych folderów utworzonych wcześniej. Jednocześnie foldery nie zawsze odpowiadają tematom przesyłanych wiadomości, czasami mogą dotyczyć zadań do wykonania, grup projektowych, niektórych odbiorców, a inne mają sens tylko w powiązaniu z poprzednimi wiadomościami lub użytkownikami. Dodatkowo wiadomość e-mail ma bardzo skomplikowany format wielowymiarowy, gdyż wiadomości mogą być przesyłane, przekazywane, kopiowane, a także może na nie odpowiadać wiele osób lub grup w różnym czasie. Przesyłane wiadomości mogą zawierać jako załączniki inne wiadomości e-mail lub dokumenty w postaci dołączonych plików. Ponadto informacje uzyskane z tematu mogą mieć inne znaczenie niż informacje uzyskane z treści lub załączników przesłanej wiadomości. Co gorsza, informacje docierają do użytkowników w różnym czasie, co powoduje dodatkowe trudności w zarządzaniu pocztą elektroniczną.

Teza

Zastosowanie algorytmów mrowiskowych i sieci społecznych w problemie automatycznego kategoryzowania wiadomości e-mail pozwala na poprawę trafności przypisywania wiadomości do folderów oraz umożliwia sugerowanie zakładania nowych folderów dla użytkowników.

Cel pracy

Głównym celem pracy jest opracowanie nowego algorytmu, którego zastosowanie pozwala na lepsze dopasowanie wiadomości e-mail do poszczególnych folde-

rów wraz z możliwością sugerowania tworzenia nowych folderów. Zaproponowany algorytm opiera się na metodyce algorytmów mrowiskowych, eksploracji danych oraz sieci społecznych. Chcąc zrealizować główny cel pracy należy m. in.:

- przeanalizować zbiór danych Enron E-mail składający się ze skrzynek pocztowych zawierających wiadomości e-mail, a następnie oczyścić go, dostosować do problemu oraz przekształcić do odpowiedniej struktury;

- stworzyć sieć społeczną opartą na kontaktach pomiędzy nadawcą a odbiorcami wiadomości e-mail, a także na podstawie analizy i obserwacji sieci społecznej wyodrębnić grupy użytkowników posiadających podobną strukturę społeczną;

- opracować własną wersję algorytmu do automatycznego kategoryzowania wiadomości e-mail, w którym klasyfikator oparty jest o algorytmy mrowiskowe, dzięki czemu możliwe jest przeszukanie większej przestrzeni rozwiązań;

- przedstawić mechanizm predykcji folderów dla użytkowników, na podstawie struktury folderów innych użytkowników wyznaczonych przez stworzoną sieć społeczną.

Rozprawa składa się z 9 rozdziałów. Po przedstawieniu tematyki, zakresu oraz postawieniu celów rozprawy opisane są najważniejsze aspekty teoretyczne dotyczące poruszanych zagadnień. Przeanalizowany jest również aktualny stan nauki związanej z problematyką niniejszej rozprawy. Następnie przedstawione są zaproponowane rozwiązania, autorskie algorytmy oraz eksperymenty związane z potwierdzeniem sensowności ich wprowadzenia.

W rozdziale pierwszym pokazane jest jak analizować i interpretować dane. Omówione są najważniejsze rodzaje analiz, takie jak opisowa analiza danych czy analiza eksploatacyjna. Przedstawione i scharakteryzowane są główne modele w analizie danych. Opisane są także dane historyczne i jakościowe. W drugiej części rozdziału wyszczególnione są rodzaje i metody eksploracji danych. Opisane są znane klasyfikatory danych, tj. maszyna wektorów nośnych, naiwny klasyfikator Bayesa, entropia, algorytm Winnow, a także scharakteryzowane są algorytmy genetyczne.

Rozdział 2 poświęcony jest zagadnieniom dotyczącym drzew decyzyjnych oraz algorytmów mrowiskowych do konstruowania drzew decyzyjnych. Przedstawione są w nim najpopularniejsze algorytmy do konstruowania drzew decyzyjnych, tj. algorytm ID3, CART oraz C4.5. Zaprezentowana jest również problematyka związana z pojęciem rodziny klasyfikatorów. Opisane są najczęściej spotykane podejścia do tworzenia rodzin klasyfikatorów, takie jak boosting, bagging i lasy losowe. Następnie omówione są zagadnienia dotyczące algorytmów mrowiskowych

do konstruowania drzew decyzyjnych. Przedstawiona jest idea systemu mrowiskowego, a także omówione jest odniesienie do systemu mrówkowego. Zaprezentowany jest algorytm mrowiskowy do konstruowania drzew decyzyjnych oraz kwestie związane z uaktualnianiem śladu feromonowego i wyznaczaniem wartości funkcji heurystycznej. Opisany jest także algorytm mrowiskowy do konstruowania lasów decyzyjnych.

W rozdziale 3 zaprezentowana jest definicja sieci społecznych oraz różne miary i wskaźniki charakteryzujące sieci społeczne. Opisane są metody analizy sieci społecznych oraz przedstawione są różne modele sieci.

Rozdział 4 realizuje pierwszy cel rozprawy, który związany jest z dokładną analizą zbioru danych Enron E-mail i problemem automatycznego przypisania wiadomości do folderów. Opisany jest proces oczyszczenia danych oraz omówiony jest mechanizm przekształcenia zbioru danych do struktury tabeli decyzyjnej. Dodatkowo przedstawiona jest szczegółowa analiza publikacji dotyczących przeprowadzonych badań na zbiorze danych Enron E-mail.

Rozdział 5 opisuje pierwsze eksperymenty przeprowadzone w celu potwierdzenia słuszności przekształcenia zbioru danych do struktury tabeli decyzyjnej. Do badań stosowane są klasyczne algorytmy eksploracji danych, a otrzymane wyniki poddawane są testom statystycznym.

W rozdziale 6 przedstawione są badania z zastosowaniem algorytmów mrowiskowych. Zaprezentowana jest metoda dotycząca zmodyfikowanej wersji algorytmu mrowiskowego do konstruowania drzew decyzyjnych. Otrzymane wyniki są porównane do wyników algorytmów klasycznych. Dodatkowo opisane są badania związane z zastoswaniem zespołów klasyfikatorów. Omówione są również wyniki testów statystycznych.

Rozdział 7 poświęcony jest realizacji kolejnego celu pracy związanego z utworzeniem sieci społecznej opartej na kontaktach pomiędzy nadawcą a odbiorcami wiadomości e-mail. Dodatkowo przedstawione są analizy stworzonych sieci społecznych, na podstawie których wyodrębnione są grupy użytkowników posiadających podobną strukturę społeczną.

W rozdziale 8 szczegółowo omówiony jest zaproponowany algorytm do automatycznego kategoryzowania wiadomości e-mail do folderów. Ponieważ jest to autorski algorytm, to na początku przedstawiona jest jego idea i schemat działania. W rozdziale zawarty jest opis eksperymentów mających na celu porównanie omawianego autorskiego algorytmu z innymi podejściami. Szczegółowo omówione w nim są wyniki doświadczeń dotyczących porównania z algorytmami, które były inspiracją do jego zaprojektowania. Dodatkowo opisane są wyniki testów statystycznych.

W rozdziale 9 przedstawiony jest mechanizm sugerowania zakładania nowych folderów dla użytkowników, na podstawie struktury folderów innych użytkowników wyznaczonych przez stworzoną sieć społeczną. Rozdział ten jest jednocześnie

realizacją ostatniego z celów pracy.

Rozprawa zakończona jest podsumowaniem wszystkich jej etapów i oceną realizacji założonych celów oraz nawiązaniem do postawionej tezy. Wskazane są w niej również kierunki dalszego rozwoju i możliwych badań dotyczących problemu automatycznego kategoryzowania wiadomości do folderów.

1

Odkrywanie wiedzy z danych

Eksploracja danych (ang. data mining) jest jedną z najdynamiczniej i najintensywniej rozwijanych dziedzin informatyki w ostatnim czasie. Występuje w wielu dyscyplinach takich jak: statystyka, systemy baz danych, sztuczna inteligencja, optymalizacja, obliczenia równoległe. Jest to nauka o wydobywaniu przydatnych informacji z dużych zbiorów danych lub baz danych. Idea eksploracji danych polega na wykorzystaniu szybkości komputera do znajdowania ukrytych dla człowieka, z uwagi na ograniczone możliwości czasowe, prawidłowości w danych zgromadzonych w hurtowniach danych. Termin eksploracja danych jest często używany jako synonim procesu odkrywania wiedzy w bazach danych. W literaturze czasami jednak rozróżnia się te dwa pojęcia. Termin odkrywanie wiedzy odnosi się do całego procesu, natomiast eksploracja danych stanowi tylko jeden z etapów tego procesu odnoszący się do generowania reguł. Pozostałe etapy procesu odnoszą się do przygotowania danych, wyboru danych do eksploracji, czyszczenia danych, definiowania dodatkowej wiedzy przedmiotowej, interpretacji wyników eksploracji i ich wizualizacji [46].

1.1 Analiza danych

Analiza i interpretacja danych należy do najważniejszych etapów badań analitycznych. Polega ona na wykorzystaniu różnych mierników i wskaźników oraz parametrów opisowych zbiorowości statystycznej, takich jak: średnie, przeciętne pozycyjne, miary zmienności, asymetrii, spłaszczenia, a także analizy korelacji i regresji. W bardziej zaawansowanych badaniach korzysta się również z analiz wielowymiarowych, takich jak: analiza czynnikowa i głównych składowych, wielowymiarowa analiza regresji, analiza dyskryminacji, analiza korespondencji, analiza koincydencji oraz analiza skalowania wielowymiarowego.

Słowo „statystyki” użyte w liczbie mnogiej oznacza mniej więcej to samo, co

dane, a dokładniej oznacza to samo, co „liczby". Statystyka, jako nauka nie jest jednak nauką o liczbach. Jednym z głównych zadań statystyki jest wydobywanie informacji z liczb, a ogólniej z zaobserwowanych danych. Samo pojęcie „statystyka" wprowadził G. Achenwall, określając ją jako naukę ułatwiającą rządzenie państwem [2].

Z historycznego punktu widzenia łatwo wyróżnić wyraźne okresy w rozwoju odrębnych metod analiz. W pierwszym okresie, trwającym do lat dwudziestych dwudziestego wieku, głównie zajmowano się opisem zebranych danych. Wyjaśniając, opisowa analiza danych polega na syntetycznym, zwięzłym opisie tych danych bez wyciągania jakichkolwiek wniosków o mechanizmach ich powstania. Zebrane lub dostarczone dane można pogrupować, uporządkować itp., można z nich także określić wiele charakterystyk.

Na początku lat dwudziestych dwudziestego wieku R. A. Fisher zapoczątkował drugi okres rozwoju metod analizy danych [38]. Cechą charakterystyczną nowych metod było to, że analizę danych dokonuje się za pomocą modeli stochastycznych. Polega to na tym, że zakłada się, że posiadane dane zostały „wyprodukowane" przez odpowiednio zdefiniowany mechanizm lub maszynę. Ten właśnie mechanizm generujący dane nazywamy modelem.

Zebrane dane statystyczne przedstawiane są zwykle w postaci tablic danych, zwanych macierzami [65]. Wyróżnić można trzy rodzaje takich tablic:

- tablice prostokątne $n \times N$ typu cecha - obiekt, zawierające wartości cech dla poszczególnych obiektów,
- tablice kwadratowe $n \times N$ typu cecha - cecha, zawierające informacje o powiązaniu cech (np. zależności, prawdopodobieństwa współwystępowania itp.),
- tablice kwadratowe $N \times N$ typu obiekt - obiekt, to zwykłe tablice odległości lub podobieństwa między badanymi obiektami.

Opisowa analiza danych

Głównym celem opisowej analizy danych jest „streszczenie" danych w postaci niewielkiej liczby prostych wskaźników, a także uporządkowanie, grupowanie oraz takie przedstawienie danych, które umożliwiałoby ich analizę wizualną.

Jednowymiarowe (jednocechowe) dane statystyczne uzyskuje się, jako realizację losowej próby n-elementowej pobranej z populacji, w której badana jest cecha X, a więc uzyskuje się je, jako realizację ciągu n niezależnych zmiennych losowych, z których każda ma taki sam rozkład, jak cecha X w populacji. W ten sposób jednocechowe dane można przedstawić jako ciąg:

$$x_1, x_2, \ldots, x_n, \tag{1.1}$$

gdzie $x_i (i = 1, \ldots, n)$ oznacza wartość cechy X zaobserwowaną dla $i - tego$ elementu próby ($i - tego$ obiektu). Często równocześnie rozważa się większą liczbę cech statystycznych. Wtedy wygodniej jest przedstawić dane w formie macierzy danych. Jeśli na przykład rozważa się łączenie p zmiennych losowych $X_1, X_2, \ldots, X_n$, to wygodną formę reprezentacji danych statystycznych będących realizacją próby n-elementowej pobranej z populacji, w której badana jest p-wymiarowa zmienna losowa $(X_1, X_2, \ldots, X_n)$ uzyskuje się przez utworzenie macierzy:

$$\mathbf{X} = \begin{bmatrix} x_{11} & x_{12} & \ldots & x_{1p} \\ x_{21} & x_{22} & \ldots & x_{2p} \\ \vdots & \vdots & \ddots & \ddots \\ x_{n1} & x_{n2} & \ldots & x_{np} \end{bmatrix}, \tag{1.2}$$

gdzie x_{ij} oznacza wartość cechy $X_j (j = 1, \ldots, p)$ zanotowaną dla $i - tego$ $(i = 1, \ldots, n)$ obiektu [22].

Dane statystyczne dotyczące jednej cechy X streszczane są zazwyczaj przez podanie średniej arytmetycznej $\overline{x}$ i wariancji s^2 otrzymanych wartości $x_1, x_2, \ldots, x_n$, przy czym:

$$\overline{x} = \frac{1}{n} \sum_{i=1}^{n} x_i, \tag{1.3}$$

$$s^2 = \frac{1}{n} \sum_{i=1}^{n} (x_i - \overline{x})^2. \tag{1.4}$$

Średnia arytmetyczna i wariancja z próby są estymatorami punktowymi wartości oczekiwanej i wariancji zmiennej losowej X w populacji.

Dane statystyczne wielocechowe streszcza się w analogiczny sposób przez obliczenie wektora średnich arytmetycznych i macierzy kowariancji. Jeżeli wartości cech $X_1, X_2, \ldots, X_p$ w próbie n-elementowej zamieszczone są w macierzy danych, to średnią arytmetyczną cechy $X_j (j = 1, \ldots, p)$ oblicza się według wzoru:

$$\overline{x_j} = \frac{1}{n} \sum_{i=1}^{n} x_{ij}, \tag{1.5}$$

a wariancję tej cechy oblicza się według wzoru:

$$s_j{}^2 = \frac{1}{n} \sum_{i=1}^{n} (x_{ij} - \overline{x_j})^2, \tag{1.6}$$

natomiast kowariancję cech X_j oraz X_k w próbie oblicza się według wzoru:

$$c_{ij} = \frac{1}{n} \sum_{i=1}^{n} (x_{ij} - \overline{x_j})(x_{ik} - \overline{x_k}). \tag{1.7}$$

Oprócz średniej arytmetycznej i wariancji można wyznaczyć inne estymatory używane jako miary położenia i zmienności. Niech $x_1, \ldots, x_n$ będzie n-elementową próbką pochodzącą z badanej populacji, na jej podstawie można wyznaczyć wielkości charakteryzujące populację [42].

Wyróżnić można następujące miary położenia:

- średnia arytmetyczna:

$$\overline{x} = \frac{1}{n}(x_1 + \cdots + x_n), \tag{1.8}$$

- średnia ważona:

$$\overline{x_w} = w_1 x_1 + \cdots + w_n x_n, \tag{1.9}$$

gdzie $w_i \geqslant 0$ dla każdego $1 \leqslant i \leqslant n$ oraz $w_1 + \cdots + w_n = 1$. Gdy $w_1 = \frac{1}{n}$ to otrzymuje się średnią arytmetyczną.

- mediana - najpierw należy uporządkować próbkę, ustawiając jej elementy w kolejności niemalejącej:

$$M_e = x_{([0,5n])}. \tag{1.10}$$

Mediana jest środkową wartością ciągu obserwacji, zdefiniowaną jako:

$$M_e = \begin{cases} x_{(m+1)}, & \text{gdy } n \text{ jest nieparzyste } (n = 2m+1) \\ 0,5(x_{(m)} + x_{(m+1)}), & \text{gdy } n \text{ jest parzyste } (n = 2m) \end{cases}, \tag{1.11}$$

- kwantyl q-ty:

$$Q_q = x_{([qn])}, \tag{1.12}$$

- (α, β) - obcięta średnia:

$$\overline{x_{\alpha,\beta}} = \frac{1}{r-m} \sum_{j=m+1}^{r} x_{(j)}, \tag{1.13}$$

gdzie $0 \leqslant \alpha, \beta \leqslant 0,5$, $m = [\alpha n]$ oraz $r = n - [\beta n]$. Wyznaczanie obcięcia średniej polega na odrzuceniu $(\alpha 100)\%$ najmniejszych i $(\beta 100)\%$ największych obserwacji, a następnie obliczeniu zwykłej średniej arytmetycznej.

- średnia typu (α, β) - Winsora:

$$W_{\alpha,\beta} = \frac{1}{n}\Big(\sum_{j=m+1}^{r} x_{(j)} + m x_{(m+1)} + (n-r) x_{(r)}\Big), \tag{1.14}$$

gdzie $0 \leqslant \alpha, \beta \leqslant 0,5$, $m = [\alpha n]$ oraz $r = n - [\beta n]$. W tym przypadku średnia jest obliczana z całej próbki, gdzie $(\alpha 100)\%$ najmniejszych i $(\beta 100)\%$ największych obserwacji jest zastąpionych przez odpowiednio $m+1$ i r-tą obserwację.

Wyróżnić także należy miary zmienności tj:

- odchylenie standardowe:

$$s = \frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^2, \tag{1.15}$$

- odchylenie przeciętne:

$$d = \frac{1}{n}\sum_{i=1}^{n}|x_i - \overline{x}|, \tag{1.16}$$

- medianowe odchylenie bezwzględne:

$$MAD = Me(|x_i - Me(x_j)|), \tag{1.17}$$

  gdzie $Me(x_j)$ oznacza medianę zbioru danych $x_1, \ldots, x_n$,

- odchylenie ćwiartkowe:

$$q = 0,5(Q_{0,75} - Q_{0,25}), \tag{1.18}$$

  jest to połowa różnicy między 0,75-tym a 0,25-tym kwantylem, czyli różnicy między trzecim, a pierwszym kwantylem.

- liniowa kombinacja kwantyli:

$$q_{\alpha,\beta} = c(q(\beta) - Q(\alpha)), \tag{1.19}$$

  gdzie $0 \leqslant \alpha \leqslant \beta \leqslant 1$. Zwykle zakłada się, że kwantyle są symetryczne, czyli $\alpha = 1 - \beta$, a stała c jest tak drobna, aby spełnione były warunki regularności otrzymanych estymatorów [42].

- (α, β)-obcięte standardowe odchylenie:

$$s_{\alpha,\beta} = \frac{1}{r-m}\sum_{j=m}^{r}(x_{(j)} - \overline{x_{\alpha,\beta}}), \tag{1.20}$$

  gdzie $0 \leqslant \alpha, \beta \leqslant 0,5$, $m = [\alpha n]$ oraz $r = n - [\beta n]$.

Wśród wielu miar odległości dwóch obiektów i oraz j, gdzie $i, j = 1, \ldots, n$, na uwagę zasługują m.in. tzw. odległość euklidesowa oraz odległość miejska (metropolitalna, taksówkowa, city-block, Manhattan) [22]. Pierwsza z nich wyraża się wzorem:

$$d_{ij} = \sqrt{\sum_{k=1}^{p}(x_{ik} - x_{jk})^2}, \tag{1.21}$$

podczas, gdy odległość miejska określona jest wzorem:

$$d_{ij} = \sum_{k=1}^{p} |x_{ik} - x_{jk}|. \tag{1.22}$$

Jedną z wielu miar odległości między populacjami jest tzw. odległość Penrose'a [69]. Odległość ta może być stosowana jedynie wtedy, gdy każda z m macierzy kowariancji $\sum_1, \sum_2, \ldots, \sum_n$ ma taką samą główną przekątną tzn. wtedy, gdy dla każdego $j(j = 1, \ldots, p)$ zachodzi $\sigma_{j1}^2 = \sigma_{j2}^2 = \cdots = \sigma_{jm}^2 = \sigma_j^2$. Oznacza to, że każda z p cech $X_1, X_2, \ldots, X_p$ ma jednakową wariancję we wszystkich m populacjach. Odległość Penrose'a między populacjami o numerach k oraz l, gdzie $(k, l = 1, \ldots, m)$ wyraża się wówczas wzorem:

$$d_{kl} = \frac{1}{p} \sum_{j=1}^{p} \frac{(\mu_{jk} - \mu_{lk})^2}{\sigma_j^2}. \tag{1.23}$$

Konstrukcja miary uwzględnia wariancje $\sigma_1^2, \sigma_2^2, \ldots, \sigma_p^2$ cech $X_1, X_2, \ldots, X_p$, pomijając przy tym wpływ wzajemnego skorelowania poszczególnych cech względem odległości między dwiema populacjami. Wpływ ten oddziałuje na wartość miary zwłaszcza wtedy, gdy wśród cech $X_1, X_2, \ldots, X_p$, znajdują się cechy istotnie skorelowane. Jeśli wśród p cech znajduje się podgrupa q silnie skorelowanych cech, które mierzą właściwie to samo lub pokrewne zjawisko oraz pewna cecha X_i niezależna od wymienionych q cech, to wpływ q redundantnych cech na wartość miary jest mniej więcej q razy większy od wpływu pojedynczej cechy X_i. W ten sposób odległość Penrose'a dwóch populacji scharakteryzowanych przez cechy silnie skorelowane rośnie proporcjonalnie do liczby cech redundantnych.

Opisanej wady nie ma miara odległości między populacjami, zwana odległością Mahalanobisa [69]. Odległość ta może być stosowana przy założeniu, że rozkłady zmiennej losowej $(X_1, X_2, \ldots, X_p)$ mają jednakową macierz kowariancji we wszystkich populacjach, tzn. gdy $\sum_1 = \sum_2 = \cdots =_s um_n$. Odległość Mahalanobisa pomiędzy populacjami o numerach k oraz l $(k, l = 1, \ldots, m)$ wyraża się wzorem:

$$d_{kl} = (\mu_k - \mu_i)^T \sum^{-1} (\mu_k - \mu_l), \tag{1.24}$$

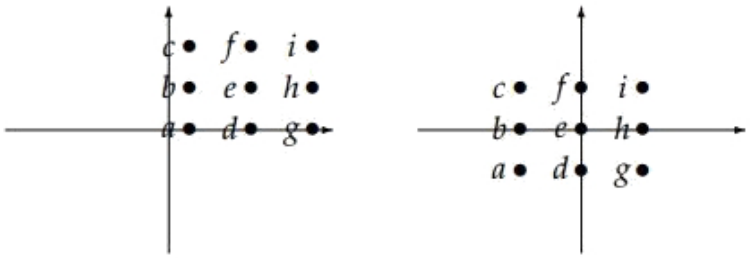
gdzie μ_ki μ_l są wektorami wartości oczekiwanych w populacji o numerach k oraz l, zaś $\sum = \sum_1 = \sum_2 = \cdots = \sum_n$ jest wspólną dla m populacji macierzą kowariancji p-wymiarowych rozkładów $(X_1, X_2, \ldots, X_p)$ w poszczególnych populacjach [65].

Eksploracyjna analiza danych

Pojęcie eksploracyjnej analizy danych pojawiło się w statystyce na początku lat siedemdziesiątych dwudziestego wieku. Jej głównym celem była zmiana dotychczasowej metodyki stosowanej do rozwiązywania problemów przy użyciu metod statystycznych. Typowe podejście statystyczne polega na zbieraniu danych

statystycznych, po czym postuluje się pewien ogólny mechanizm losowy oparty na tych danych do wyekstrahowania nowych danych. W terminologii statystycznej postulowanie mechanizmu generującego dane nazywa się przyjęciem założenia o postaci modelu, jego sprecyzowanie nazywa się estymacją, a zastosowanie go do generowania nowych danych nazywa się wnioskowaniem. Celem analizy eksploracyjnej jest uzasadnienie wyboru teorii (modelu) do analizy posiadanych danych. Eksploracyjna analiza danych jest bardziej ukierunkowana na stawianie pytań niż udzielanie odpowiedzi. W wyniku przeprowadzonej analizy generowane są hipotezy, które następnie są analizowane, badane i weryfikowane [67]. Większość rozumowań na etapie analizy eksploracyjnej jest oparta na tzw. zdrowym rozsądku, gdzie wnioskowanie przeprowadzane jest dość prosto, pod warunkiem, że istnieje możliwość obejrzenia danych.

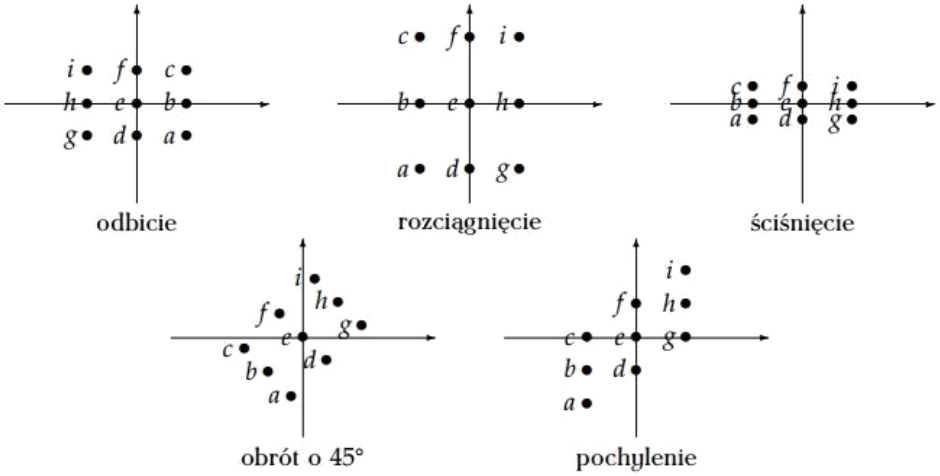
Jednym z podstawowych założeń prawie wszystkich metod analizy danych jest założenie o możliwości interpretowania tych danych jako zbiorów w przestrzeni liniowej, a ściślej w n-wymiarowej przestrzeni euklidesowej. Istnieje wtedy możliwość wykorzystania do analizy danych całego aparatu analitycznego algebry liniowej, w szczególności możliwość zastosowania przekształceń liniowych. Przykłady przekształceń liniowych pokazano na rysunkach 1.1 i 1.2. Zapisywane są one w postaci macierzy zwanych macierzami przekształcenia. Szczególnie często stosowane w analizie danych są przekształcenia polegające na obrocie punktów. Odpowiednia zmiana bazy może w istotny sposób ułatwić interpretację wyników. Innym przekształceniem jest centrowanie. Polega ono na odjęciu wartości średniej. Na rys. 1.1. przedstawiony jest zbiór X oraz jego wycentrowanie.



Rysunek 1.1: Dane oryginalne i przesunięte do środka

Każda analiza danych wielowymiarowych, jeśli nie ma żadnych dodatkowych informacji dotyczących ich struktury, powinna być poprzedzona analizą graficzną. Inspekcja wzrokowa zebranych danych możliwa jest w przestrzeni R_n w przypadku, gdy $n < 3$, gdyż najwygodniej „ogląda" się punkty umieszczone na płaszczyźnie. Nikt nie potrafi zobaczyć zbioru punktów w przestrzeni R_n , jeśli $n > 3$. Dla tablic dwudzielczych można wykreślić rysunki perspektywiczne rozkładów badanych cech odpowiadające wykresom kolumnowym lub histogramom dla danych jednowymiarowych. Wykres kolumnowy wykonuje się wtedy, gdy dane są typu

kategorialnego lub cechy są skokowe. Wykres histogramowy tworzony jest dla obu cech ciągłych [42].



Rysunek 1.2: Przykłady przekształceń liniowych

Innym sposobem graficznego przestawienia danych jest ciąg wykresów rozkładów drugiej cechy wykonanej dla każdej kategorii wartości pierwszej cechy. Można tu dokładnie zobaczyć wartości liczebności, a nie tylko rzut perspektywiczny. Zaletą histogramu trójwymiarowego jest zwięzłe przedstawienie całej tabeli.

Do innej kategorii metod graficznych często używanych przy prezentacji danych wielowymiarowych zalicza się tzw. wykresy symboliczne. Ich istota polega na przedstawieniu każdej obserwacji w postaci symbolicznego rysunku. Do budowy wielu z nich wykorzystuje się jedną i tę samą zasadę, choć wynikiem jest inne rozwiązanie graficzne. Warunkiem wspólnej prezentacji wszystkich obserwacji wielowymiarowych jest ich przeskalowanie przez unormowanie bądź standaryzację. Wspólną częścią tych rysunków jest narysowanie wychodzących z jednego punktu p promieni, rozłożonych przy zachowaniu jednakowych kątów między nimi. Kąt ten jest równy $2\pi \backslash p$. Każdy promień symbolizuje jedną ze składowych wektora obserwacji [67]. Do najczęściej używanych wykresów należą:

- wykres gwiazdowy – gdzie długość każdego promienia jest proporcjonalna do wartości j-tej cechy dla i-tej obserwacji ($j = 1, \ldots p, i = 1, \ldots, n$);

- wykres rytów Andersona – jest jednym z wariantów wykresu gwiazdowego, przy czym zamiast wspólnego środka wykreślane jest koło. Z tego koła wyprowadzane są promienie w postaci pęczków. Długość promienia jest proporcjonalna do wartości j-tej cechy dla i-tej obserwacji. Do pęczka promieni zalicza się te cechy, które są ze sobą najmocniej skorelowane;

- wykres promieniowy – gdzie długość każdego promienia jest równa 1 lub $6s_j$ w zależności od przyjętego sposobu skalowania. Na każdym promieniu odkładana jest od wspólnego punktu przeskalowana wartość j-tej cechy (w przypadku danych standaryzowanych do każdej wartości cechy j należy dodać uprzednio $3s_j$). Następnie należy połączyć kolejno końce wyznaczonych odcinków na promieniach;
- wykres wielokątowy – wykonuje się tak samo jak wykres promieniowy, ale po jego wykreśleniu wymazuje się promienie pozostawiając zaznaczony punkt położenia początków promieni, oraz zamalowuje się powstały wielokąt;
- wykres torowy – gdzie tor reprezentowany jest przez powierzchnię koła, którą za pomocą promieni należy podzielić na kawałki o kącie proporcjonalnym do odpowiednio przeskalowanych danych (w przypadku danych standaryzowanych do każdej wartości cechy j należy dodać uprzednio $3s_j$);
- Metoda krzywych Andrewsa – w której każda obserwacja jest reprezentowana przez oddzielną krzywą: $\phi_i(t)$, gdzie $i = 1, \ldots, n$, okresową dla $t \in < -\pi, \pi >$. Krzywe te mają postać:

$$f_i(t) = \frac{1}{\sqrt{2}} x_n + \sum_{j=1}^{k} (\sin(jt) x_{i,2j} + \cos(jt) x_{i,2j+1}), \tag{1.25}$$

gdzie:

$$k = \begin{cases} \frac{p}{2}, & \text{dla } p \text{ parzystych} \\ \frac{p-1}{2}, & \text{dla } p \text{ nieparzystych} \end{cases} . \tag{1.26}$$

1.2 Modele w analizie danych

Model jest wysokopoziomowym, ogólnym opisem zbioru danych. Przyjmuje perspektywę dużej próbki. Może być opisowy, podsumowujący dane w dogodny i zwięzły sposób, lub też indukcyjny, pozwalający formułować wnioski o populacji, z której pobrane zostały dane lub o prawdopodobnych wartościach przyszłych danych. Model jest zatem uproszczonym odwzorowaniem rzeczywistości. Jest pojęciem abstrakcyjnym, będącym pomostem między abstrakcyjnymi sposobami myślenia a realnie istniejącą rzeczywistością. Dobrze skonstruowany model jest kompromisem między dwoma skrajnościami: nadmiernego uproszczenia oraz nadmiernego nagromadzenia szczegółów.

Zgodność i różnice między modelem a modelowaną rzeczywistością powinny być określone w sposób dokładny i nie budzący wątpliwości. Model jest nie tylko środkiem uzyskiwania informacji, ale także narzędziem za pomocą którego można rozwinąć teorię badanej dziedziny. Studiowanie modelu umożliwia wysuwanie wniosków o przedstawianych obserwacjach.

Głównie modele dzielimy na:

- modele materialne – przedstawiają realnie istniejące obiekty, są ich zminiaturyzowaną kopią o pełnej lub ograniczonej funkcjonalności,
- modele niematerialne – mają ściśle teoretyczny charakter i istnieją jedynie w ludzkiej świadomości.

Wśród modeli drugiego rodzaju można wyróżnić:

- model konceptualny – zbudowany według przyjętych opisów jakościowych dotyczących rzeczy, oddziaływań, relacji, struktur, zjawisk, sposobów funkcjonowania za pomocą hipotetycznych konstrukcji lub idealizacji danych prowadzących do werbalnego przedstawienia badanych zjawisk,
- model symboliczny – którego celem jest przedstawienie opisów za pomocą symboli mogących być elementami języka (tzw. model lingwistyczny) oraz grafiką (tzw. model ikonograficzny),
- model formalny – polega na przedstawieniu badanych zjawisk w sformalizowanym języku matematyki.

Modele formalne

Pojęcie modelu formalnego jest również wieloznaczne. Najczęściej przez model rozumie się pewną teorię, niekiedy jej matematyczne przedstawienie, ale często także jest to jakiś kwantyfikowalny opis pewnych aspektów danych. Model formalny musi w obiektywny sposób opisywać zjawiska i procesy świata rzeczywistego. Wśród modeli formalnych odróżnić należy modele deterministyczne od modeli stochastycznych.

Jeśli w każdej konkretnej sytuacji spełnione są przyjęte założenia i warunki, to deterministyczne związki mają wagę praw. Formalny model deterministyczny w wyniku daje ciąg liczb nielosowych. W deterministycznym modelu znana jest nie tylko struktura, która może być opisana za pomocą równań algebraicznych lub różniczkowych, ale znane są także wartości parametrów [65].

Modele stochastyczne

Jeżeli do równania wprowadzi się element nieokreśloności, mówi się wtedy o modelu stochastycznym. W odróżnieniu od modelu deterministycznego model stochastyczny dla konkretnej sytuacji generuje losowe odpowiedzi zgodnie z pewnym rozkładem prawdopodobieństwa. Modele stochastyczne pozwalają na stosunkowo dokładne odtworzenie za pomocą aparatu matematycznego związków między cechami, gdy w danych obserwuje się te związki, przynajmniej w przybliżeniu. Jako definicję modelu stochastycznego można przyjąć pojedyncze równanie

lub układ równań zawierający zmienne losowe, zmienne nielosowe i parametry. Zmiennymi losowymi są te zmienne, których wartości nie są znane przed eksperymentem, zmiennymi nielosowymi są zmienne o kontrolowanych wartościach, natomiast parametrami są znane lub częściowo nieznane stałe.

Jeżeli do budowy modelu stochastycznego zastosuje się ujecie dedukcyjne, to w wyniku otrzyma się model probabilistyczny. Jest on odtworzeniem charakterystyk probabilistycznych badanego zjawiska, obiektu lub procesu, które otrzymano za pomocą metod rachunku prawdopodobieństwa na podstawie znajomości charakterystyk losowych elementów składowych relacji między nimi. W przypadku ujęcia indukcyjnego otrzymuje się model statystyczny. Jest on układem współzależności opisujących pewien obiekt, proces lub zjawisko, którego parametry estymuje się na podstawie rzeczywistych danych za pomocą metod statystycznych. Proces budowy modelu rozpada się na dwa wzajemnie powiązane etapy:

- określenie ogólnej postaci powiązań modelu i występujących w nich zmiennych, oraz
- statystyczna estymacja wartości parametrów na podstawie zaobserwowanych danych.

Modele ekonometryczne

Wśród modeli statystycznych można wyróżnić modele ekonometryczne, przedstawiające układ jednoczesnych równań regresyjnych wiążących wektor zmiennych endogenicznych z wektorem odwzorowań zmiennych egzogenicznych i elementami losowymi. Grupę zmiennych endogenicznych tworzą wszystkie zmienne objaśniane danego modelu, które w niektórych funkcjach modelu mogą występować jako zmienne objaśniające, z kolei zmienne egzogeniczne mogą występować tylko jako zmienne objaśniające.

Modele ekonometryczne dzielą się na modele statyczne, dotyczące wzajemnych powiązań między zmiennymi modelu, które w sposób jawny nie zależą od czasu, i modele dynamiczne, gdzie uwzględniany jest moment pojawienia się obserwacji. Powiązanie zmiennych i parametrów wynikające z postaci funkcji modelu pozwala wyróżnić pewne klasy modeli, takie jak:

- model liniowy,
- modle nieliniowy sprowadzany do modelu liniowego,
- model nieliniowy.

Podział ten ma znaczenie przy wyborze metod estymacji odpowiednich parametrów do klasy modelu. Uproszczenia najczęściej są związane z postulatem liniowości, a czasem ze stopniem nieliniowości modelu, gdyż większość systemów czy procesów ma charakter nieliniowy, co znacznie ogranicza stosowanie modelu

liniowego. W zależności od tego czy problem można zapisać w postaci jednej funkcji, czy ich układu, rozróżnić można modele jednorównaniowe i wielorównaniowe, które dzielą się na modele proste, rekurencyjne, składające się z określonej liczby modeli jednorównaniowych powiązanych ze sobą zależnością jednostronną, i o równaniach współzależnych, gdzie występują sprzężenia zwrotne między zmiennymi endogennymi.

Z kolei modele dynamiczne można dodatkowo podzielić na:

- modele autoregresyjne – zawierające w równaniu oprócz zmiennej endogenicznej funkcję wartości tej zmiennej z okresów wcześniejszych,
- modele adaptacyjne – inaczej modele samokorygujące się, samonastrajające się, zdolne odzwierciedlać zmieniające się w czasie warunki, uwzględniające informacyjną wartość różnych składników szeregu,
- modele trendu – pozwalające na uniknięcie problemu z doborem zmiennych do modelu, gdyż jedyną zmienną jest w nich czas t,
- modele przyczynowo-skutkowe – posiadające rozszerzoną listę zmiennych w modelu o takie, które mogą mieć wpływ na zmienną endogeniczną,
- modele harmoniczne.

Istnieją dwie główne kategorie modeli rozkładu i gęstości, a mianowicie modele parametryczne i nieparametryczne. W modelach parametrycznych znana jest postać modelu z dokładnością do nieznanych parametrów podlegających estymacji, natomiast w modelach nieparametrycznych estymowana jest postać modelu najczęściej za pomocą metod estymacji gęstości rozkładu prawdopodobieństwa. W modelach parametrycznych przyjmuje się jedną z postaci funkcyjnych. Dla zmiennych o wartościach rzeczywistych funkcja jest często opisywana przez parametr położenia (wartość średnią) i parametr skali (opisujący zmienność). Zaletą modeli parametrycznych jest ich prostota, gdyż są łatwe do estymacji i interpretacji, ale mogą mieć stosunkowo duże obciążenie, ponieważ rzeczywiste dane mogą nie spełniać przyjętej postaci funkcyjnej. Natomiast w modelach nieparametrycznych rozkład lub gęstość są szacowane na podstawie danych, a stosunkowo mało założeń o postaci funkcyjnej przyjmuje się a priori [67].

Dane historyczne i jakościowe

Zmienne rejestrowane w równych odstępach czasu, np. codziennie, co tydzień, co miesiąc, co kwartał lub co rok tworzą szeregi czasowe, czyli ciągi obserwacji dotyczące cechy X, w których kolejne obserwacje odnoszą się do następujących po sobie kolejnych momentów lub okresów czasu. Szereg czasowy cechy X można przedstawić jako ciąg zmiennych losowych $X_1, X_2, \ldots, X_n$, gdzie X_t oznacza rozkład cechy X w momencie lub okresie t, ponieważ rozkład cechy X zmienia się

na ogół miarę w upływu czasu. Jeżeli przy tym znane są wartości $x_1, x_2, \ldots, x_n$ zmiennych losowych $X_1, X_2, \ldots, X_n$ to wówczas ciąg $x_1, x_2, \ldots, x_n$ nazywany jest realizacją szeregu czasowego. Jeżeli obserwacje cechy X dokonywane są w czasie ciągłym, to ciąg zmiennych losowych X_t, gdzie t należy do pewnego zbioru T liczb rzeczywistych, nazywa się procesem stochastycznym oznaczanym symbolami:

$$\{X_t - X_t(\omega) \mid t \in T, \omega \in \Omega\} \tag{1.27}$$

lub krócej:

$$\{X_t \mid t \in T\}, \tag{1.28}$$

gdzie t należy do zbioru liczb naturalnych lub całkowitych.

Prognozowanie szeregów czasowych ma na celu oszacowanie przyszłych wartości rozważanej cechy X, czyli wartości $x_{n+1}, x_{n+2}, \ldots$ zmiennych losowych $X_{n+1}, X_{n+2}, \ldots$ na podstawie dotychczas zaobserwowanych wartości $x_1, x_2, \ldots, x_n$ badanego szeregu czasowego. Prognozowanie szeregów czasowych jest niekiedy bardziej efektywne dzięki zastosowaniu tzw. analizy składowych szeregu czasowego, czyli przedstawieniu cechy X_t jako iloczynu lub sumy pewnej liczby składowych, takich jak: trend, składowa cykliczna, składowa sezonowa czy składowa losowa. Uzyskana w ten sposób wiedza o kształtowaniu się poszczególnych składowych szeregu w czasie jest stosowana do konstrukcji trafnej prognozy. W procesie prognozowania można wyróżnić następujące etapy:

- zbieranie i weryfikacja danych,
- konstrukcja reguły prognozowania,
- ocena adekwatności modelu,
- wybór reguły prognozowania dokonany na podstawie analizy prawidłowości w zebranych danych,
- podział szeregu czasowego na część początkową – ciąg o długości m i część testową – ciąg o długości n,
- dopasowanie parametrów wybranej reguły prognostycznej do części początkowej,
- zastosowanie wybranej reguły prognozowania do wyznaczania prognoz dla części testowej i wyznaczenie miary błędu prognozy dla zadanej reguły prognostycznej,
- podjęcie decyzji o przydatności danej reguły prognozowania na podstawie obliczonej miary błędu prognozy,
- wyznaczenie prognozy.

Wśród różnego rodzaju danych, które są pozyskiwane w licznych badaniach wyróżnia się dane jakościowe. Ich główną cechą charakterystyczną jest to, że nie można ich wyrazić bezpośrednio za pomocą jednostek miary, tak jak np. wiek (w latach), wzrost (w centymetrach) czy zarobek (w złotówkach). Cechami jakościowymi są np. płeć, kolor włosów, kolor oczu, gdzie każdą z cech można podzielić na poszczególne kategorie takie jak: płeć: męska, żeńska, kolor oczu: zielony, piwny, niebieski, kolor włosów: blond, rudy, brąz itd. Dane jakościowe najłatwiej i najczęściej są przedstawiane w formie tablicy, którą nazywa się tablicą kontyngencji [42].

1.3 Metody eksploracji danych

Eksploracja danych jest najważniejszym etapem w procesie odkrywania wiedzy z danych. Odkrywanie wiedzy w bazach danych (ang. knowledge discovery in databases) w skrócie oznaczane jako KDD, to złożony proces mający doprowadzić użytkownika do nowych, nieznanych wcześniej czytelnych informacji i wniosków, które jednocześnie zachowują status globalnych i wiarygodnych prawidłowości występujących w badanym zbiorze danych. Ogólnie rzecz biorąc proces ten polega na określeniu problemu, przygotowaniu danych, ich eksploracji oraz interpretacji wyników.

Określenie problemu należy rozumieć jako fazę zdefiniowania problemu, czyli postawienia pytań, na które ma być poszukiwana odpowiedź. Przygotowanie danych, czyli przetwarzanie wstępne danych, składa się przede wszystkim z procesów:

- czyszczenia danych, procesu odpowiedzialnego za wyeliminowanie zbędnej redundancji oraz dokonującego uspójnienia danych w hurtowniach danych. Ma szczególne zastosowanie w przypadku korzystania z kilku systemów źródłowych;
- transformacji danych, a więc doprowadzanie zbiorów do postaci adekwatnej do etapu eksploracji. Konwersja typów atrybutów, definicja atrybutów wywiedzionych, dyskretyzacja wartości ciągłych;
- selekcji danych, czyli wyboru ważnych w kontekście badanego problemu danych (relacji i krotek przeznaczonych do eksploracji).

Eksploracja danych to etap, w którym przy użyciu zewnętrznych metod dąży się do wydobycia zależności z badanych zbiorów danych. Interpretacja wyników, czyli przetwarzanie końcowe, obejmuje wybór najbardziej interesujących informacji, filtrowanie oraz wizualizację i interpretację uzyskanych wyników.

Nie trudno zauważyć, że najważniejszym etapem odkrywania wiedzy jest moment eksploracji, przez co staje się głównym elementem odpowiedzialnym za

efektywne przeprowadzenie całego procesu. Genezy tego można się doszukiwać jeszcze na etapie powstawania obydwóch terminów, kiedy to odkrywanie wiedzy zostało przedstawione jako poszukiwanie wiedzy w posiadanych zbiorach danych, a eksploracja odnosiła się przede wszystkim do technik i narzędzi (algorytmów) stosowanych podczas poszukiwania informacji. W wielu publikacjach pojęcia te przedstawiane są czasem jako synonimy, co jednak nie jest najlepszym wyjściem, ponieważ rozróżnienie ich pozwala na wprowadzenie pewnej zależności i przejrzystości podczas odkrywania wiedzy w bazach danych.

Ze względu na swoją wszechstronność eksploracja danych, w zależności od celu, do jakiego dąży oraz typu powiązań posługuje się różnymi technikami. Wyróżnić należy kilka podstawowych technik eksploracji danych [62], tj.:

- odkrywanie klasyfikacji,
- odkrywanie asocjacji,
- klastrowanie,
- odkrywanie wzorców sekwencji,
- odkrywanie podobieństw w przebiegach czasowych,
- wykrywanie zmian i odchyleń.

Odkrywanie klasyfikacji

Celem klasyfikacji jest predykcja wartości danego atrybutu w oparciu o zadany zbiór danych treningowych, czyli znajdowanie sposobu odwzorowania danych w zbiór predefiniowanych klas. Klasyfikację realizuje się przeważnie przy użyciu drzew decyzyjnych, które buduje się na podstawie danych uczących. Tak zbudowane drzewa decyzyjne pozwalają klasyfikować nowe obiekty w bazie. Klasyfikacja danych została szerzej omówiona w rozdziale 1.4.

Odkrywanie asocjacji

Odkrywanie asocjacji (kojarzenie) jest jedną z najciekawszych i najbardziej popularnych technik eksploracji danych. Celem procesu odkrywania asocjacji jest znalezienie interesujących i nieznanych zależności lub korelacji, nazywanych ogólnie asocjacjami, pomiędzy danymi w dużych zbiorach danych. Odkrywaniu zależności mogą towarzyszyć miary statystyczne pozwalające określić wsparcie i ufność znalezionych zależności. Wynikiem procesu odkrywania asocjacji jest zbiór reguł asocjacyjnych opisujących znalezione zależności lub korelacje pomiędzy danymi. Odkrywanie reguł asocjacji polega na znajdowaniu związków pomiędzy występowaniem grup atrybutów w bazie danych [61, 62].

Klastrowanie

Klastrowanie ma na celu znalezienie skończonego zbioru obiektów o podobnych cechach, zapisanych w bazie danych [61]. Klastrowanie powoduje zgrupowanie obiektów w klasy tak, aby podobieństwo wewnątrzklasowe było możliwie duże, a podobieństwo pomiędzy klasami obiektów możliwie małe.

W przypadku klastrowania proces grupowania obiektów opiera się na analizie danych o wszystkich możliwych do opisania obiektach. Odróżnia to w znacznym stopniu technikę klastrowania od klasyfikowania, w którym konkretne klasy wyznaczone zostają jeszcze przed rozpoczęciem grupowania. Można więc powiedzieć, że w przypadku klastrowania poszukujemy klas obiektów o zbliżonych cechach [61].

Odkrywanie wzorców sekwencji

Metoda odkrywania wzorców sekwencji polega na analizie danych zawierających informacje o zdarzeniach, które wystąpiły w określonym przedziale czasu. Celem jest znalezienie zależności pomiędzy występowaniem określonych zdarzeń w czasie tworząc czasowe wzorce zachowań. Należy podkreślić fakt mówiący, że zdarzenia znalezionej sekwencji czy zależności czasowej nie muszą występować bezpośrednio po sobie, ale mogą być rozdzielane innymi zdarzeniami, zachowując jednak określone właściwości [61, 62].

Metoda odkrywania wzorców sekwencji znalazła zastosowanie w wielu dziedzinach tj.: telekomunikacja, medycyna, ubezpieczenia i bankowość. Metodę odkrywania wzorców sekwencji stosuje się także w analizie kataklizmów (trzęsienia ziemi, huragany, erupcje wulkanów) w celu określenia typowej sekwencji zdarzeń prowadzących do wystąpienia kataklizmu. Metody te są również szeroko stosowane w bankowości i ubezpieczeniach do analizy zachowań klientów i wykrywania oszustw ubezpieczeniowych.

Wreszcie, metody te znalazły szerokie zastosowanie do analizy efektywności i organizacji serwerów Web, np. można zreorganizować sposób nawigacji po stronach WWW (poprawić i uprościć strukturę połączeń pomiędzy skorelowanymi stronami), czy też poprawić sposób prezentacji reklam dostosowując je do określonych grup użytkowników [57].

Pozostałe metody

Poza wymienionymi powyżej podstawowymi technikami eksploracji danych istnieje jeszcze kilka innych częściej lub rzadziej stosowanych technik, których nie sposób wszystkich wymienić. Można zwrócić jednak uwagę na takie metody jak [61, 98]:

- Wykrywanie punktów osobliwych – obejmuje metody wykrywania (znajdowania) obiektów osobliwych, które odbiegają od ogólnego modelu danych (klasyfikacja i predykcja) lub modeli klas (analiza skupień).
- Analiza przebiegów czasowych – obejmuje metody analizy przebiegów czasowych w celu znalezienia: trendów, podobieństw, anomalii oraz cykli.
- Opisy koncepcji/klas – zawierają je metody znajdowania zwięzłych opisów lub podsumowań ogólnych własności klas obiektów. Znajdowane opisy mogą mieć postać reguł charakteryzujących lub reguł dyskryminacyjnych. W tym drugim przypadku, opisują różnice pomiędzy ogólnymi własnościami tak zwanej klasy docelowej (klasy analizowanej) a własnościami tak zwanej klasy (zbioru klas) kontrastującej (klasy porównywanej).
- Analiza trendów i odchyleń – obejmuje metody analizy danych zmiennych w czasie w celu znalezienia różnic pomiędzy aktualnymi a oczekiwanymi wartościami danych, anomalnych zmian wartości danych w czasie.
- Odkrywanie podobieństw w przebiegach czasowych – polega na znajdowaniu podobieństw w przebiegach czasowych opisujących określone procesy.
- Wykrywanie zmian i odchyleń – polega na znajdowaniu różnic pomiędzy aktualnymi a oczekiwanymi wartościami danych. Przykładem może być znajdowanie anomalnych zachowań klientów ubezpieczalni, właścicieli kart kredytowych, czy klientów firm telekomunikacyjnych.
- Regresja – funkcyjaą zależność zmiennej losowej od innej zmiennej z dokładnością do błędu losowego o wartości oczekiwanej równej zero [74]. Innymi słowy w wyniku regresji otrzymuje się funkcję przyporządkowującą konkretną wartość danemu elementowi z badanego zbioru. Tak więc regresja w pewien sposób stara się odwzorować dane w wartości liczbowe [98, 53].

1.4 Klasyfikacja

Słowo „klasyfikacja" pochodzi od łacińskich słów „classis" i „facio" oznaczających odpowiednio „oddział" oraz „czynić". Pojęcie klasyfikacja (ang. classification) nie jest jednoznaczne i bywa rozważane zarówno na płaszczyźnie semantycznej, jak i logicznej, czy teoriomnogościowej. Można tu wymienić trzy podstawowe znaczenia słowa klasyfikacja:

- metoda podziału zbioru obiektów na klasy,
- czynność przydzielania obiektów do klas,
- zbiór klas będący wynikiem grupowania.

Biorąc pod uwagę te trzy znaczenia klasyfikacja może być rozumiana jako systematyczny podział przedmiotów (zjawisk) na klasy, działy i poddziały, dokonywany według określonej zasady. Klasyfikacja umożliwia poznanie rzeczywistości poprzez redukcję, entropię i dlatego jest podstawowym narzędziem odkrywania praw przyrody oraz konstruowania teorii naukowych. Klasyfikacja ma szerokie zastosowanie w wielu dziedzinach, na przykład stosowana jest w biologii przy klasyfikowaniu roślin i zwierząt na rodzaje, rasy czy gatunki. W ekonomii klasyfikacji podlegają kraje, czy przedsiębiorstwa, a w medycynie pacjenci i choroby.

Klasyfikacja jest jedną z najpopularniejszych technik eksploracji danych. Polega na stworzeniu modelu, który umożliwia przypisanie nowego, wcześniej niewidzianego obiektu, do jednej ze zbioru predefiniowanych klas. Model umożliwiający takie przypisanie nazywa się klasyfikatorem. Poprzez klasyfikator dokonywane jest przypisanie do klasy na podstawie doświadczenia nabytego podczas trenowania i testowania na zbiorze uczącym. W trakcie wieloletnich prac prowadzonych nad klasyfikatorami i ich zastosowaniem w statystyce, uczeniu maszynowym, czy sztucznej inteligencji, zaproponowano bardzo wiele metod klasyfikacji [61, 62].

Najczęściej stosowane techniki to klasyfikacja bayesowska, klasyfikacja na podstawie k najbliższych sąsiadów, drzewa decyzyjne, sieci neuronowe, sieci bayesowskie, czy algorytmy SVM (ang. support vector machines). Popularność technik klasyfikacji wynika przede wszystkim z faktu szerokiej stosowalności tego modelu wiedzy. Klasyfikatory mogą być stosowane do oceny ryzyka związanego z udzieleniem klientowi kredytu, wyznaczeniem prawdopodobieństwa przejścia klienta do konkurencji, czy znalezienia zbioru klientów, którzy z największym prawdopodobieństwem odpowiedzą na ofertę promocyjną.

Podstawową wadą wielu technik klasyfikacji jest konieczność starannego wytrenowania klasyfikatora i trafnego wyboru rodzaju klasyfikatora w zależności od charakterystyki przetwarzanych danych. Te czynności mogą wymagać od użytkownika wiedzy technicznej, zazwyczaj wykraczającej poza sferę kompetencji analityków i decydentów.

Techniką podobną do klasyfikacji jest regresja (ang. regression). Różnica między dwiema technikami polega na tym, że w przypadku klasyfikacji przewidywana wartość jest kategoryczna, podczas gdy w regresji celem modelu jest przewidzenie wartości numerycznej. Często, metody wykrywania punktów osobliwych stanowią integralną część innych metod eksploracji danych, na przykład, metod grupowania [64].

Klasyfikacja jest procesem polegającym na przypisaniu każdego dokumentu d_i z danego zbioru treningowego:

$$D_{train} = \{(d_1, c_1), \ldots, (d_m, c_m)\} \tag{1.29}$$

do jednej z predefiniowanych klas w oparciu o zbiór wartości atrybutów opisujących dany dokument. Tak więc dla danego dokumentu d_i reprezentowanego przez

wektor cech $(x_1, \ldots, x_n)$ można znaleźć odwzorowanie przypisujące mu jedną klasę ze zbioru $C = \{c_1, \ldots, c_m\}$. Odwzorowanie:

$$f : R^n \ni (x_1, \ldots, x_n) \rightarrow c_m \in C \tag{1.30}$$

nazywamy klasyfikatorem albo odwzorowaniem klasyfikacyjnym. Celem kategoryzacji tekstu jest nauczenie algorytmu generowania klasyfikatora na podstawie zbioru treningowego. Jednak do stworzenia odwzorowania klasyfikacyjnego niezbędne są dodatkowe informacje. Zazwyczaj przyjmują one formę profilu (prototypu klasy) zawierającego typowe, charakterystyczne cechy odróżniające daną kategorię od innych lub formę zbioru przykładów dokumentów należących do poszczególnych kategorii, który może posłużyć bezpośrednio do budowy odwzorowania klasyfikacyjnego, lub pośrednio do wygenerowania profilu.

Maszyna Wektorów Nośnych

Maszyna Wektorów Nośnych (ang. Support Vector Machine, SVM) to technika uczenia maszynowego pozwalająca na analizowanie danych i określanie wzorców w celu klasyfikacji, która polega na określeniu, do której z dwóch klas należy przypisać zbiór danych wejściowych. Do procesu uczenia maszyny wektorów nośnych wymagany jest zbiór uczący, w którym każdy element tego zbioru ma oznaczenie, do której klasy należy. Uzyskany model SVM reprezentuje dane ze zbioru uczącego oddzielone od siebie granicą z najszerszym możliwym marginesem, czyli odległością od tej hiperpłaszczyzny. Maszyna Wektorów Nośnych została po raz pierwszy przedstawiona przez Vladimira Vapnika w pracy [26].

Wśród SVM rozróżnić można klasyfikator liniowy oraz nieliniowy, a także klasyfikację C-SVC. Istotą metody SVM jest konstrukcja optymalnej hiperpłaszczyzny, której zadaniem jest rozdzielenie danych, należących do przeciwnych klas, z możliwie największym marginesem zaufania. W klasyfikacji C-SVC celem jest znalezienie maksimum funkcji określającej margines między wyznaczonymi klasami poprzez rozwiązanie zadania optymalizacji kwadratowej. Dodatkowo wprowadzona jest funkcja kary za błąd w określeniu klasy. Dla danego zbioru treningowego $T = \{(x_1, c_1), \ldots, (x_n, c_n)\} \subseteq R^k \times \{C_1, C_2\}$, poprzez (zazwyczaj nieliniową) transformację $\phi : R^k \rightarrow \tau$ do przestrzeni τ pwstaje nowe zadanie klasyfikacji:

$$T' = \{(\phi(x_1), y_1), \ldots, (\phi(x_n), y_n)\} \subseteq \tau \times \{-1, 1\}, \tag{1.31}$$

gdzie:

$$y = \begin{cases} 1, & \text{o ile } c_i = C_1 \\ -1, & \text{w przeciwnym przypadku.} \end{cases}. \tag{1.32}$$

Celem klasyfikacji jest znalezienie takiego wektora w, dla którego:

$$y_i(w \cdot \phi(x_i) + b) \geqslant 1, \quad i = 1, \ldots, n, \tag{1.33}$$

przy założeniu separowalności klas
lub:

$$y_i(w \cdot \phi(x_i) + b) \geqslant 1 - \xi_i, \quad \xi_i \geqslant 0, \quad i = 1, \ldots, n \tag{1.34}$$

dla danych nieseparowalnych.

Liczba niezerowych ξ_i jest wówczas liczbą błędów popełnianych w klasyfikacji wektorów zbioru treningowego. Zatem minimalizację liczby błędów można uzyskać poprzez minimalizację:

$$\sum_{i=1}^{n} \xi_i. \tag{1.35}$$

W związku z powyższym szukanie optymalnego modelu SVM można zdefiniować jako problem minimalizacji:

$$\min_{w,b,\xi} \left[\frac{1}{2} ||w||^2 + D \sum_{i=1}^{n} \xi_i \right], \tag{1.36}$$

gdzie:

D jest stałą nośności danych,

w jest wektorem współczynników,

b jest stałą reprezentującą parametry przenoszenia danych wejściowych.

Naiwny klasyfikator Bayesa

Naiwny klasyfikator Bayesa (ang. naive Bayes classiffier) jest jedną z metod uczenia maszynowego, stosowaną do rozwiązywania problemu sortowania i klasyfikacji. Zadaniem klasyfikatora Bayesa jest przyporządkowanie nowego przypadku do jednej z klas decyzyjnych, przy czym zbiór klas decyzyjnych musi być skończony i zdefiniowany apriori.

Naiwny klasyfikator Bayesa jest statystycznym klasyfikatorem, opartym na twierdzeniu Bayesa, które pokazuje w jaki sposób obliczyć prawdopodobieństwo warunkowe $P(C_i/X)$ tego, że obiekt o właściwościach X należy do klasy C_i. Prawdopodobieństwa $P(X/C_i)$, $P(C_i)$, $P(X)$ można bezpośrednio wyliczyć z danych zgromadzonych w treningowym zbiorze danych. Naiwny klasyfikator Bayesa zakłada, że wartości atrybutów w klasach są niezależne. Założenie to jest zwane założeniem o niezależności warunkowej klasy (ang. class conditional independence).

W naiwnym klasyfikatorze Bayesa każdy obiekt traktowany jest jako wektor X wartości atrybutów $A = \{A_1, \ldots, A_n\}$ takich, że $X = (x_1, x_2, \ldots, x_n)$. W naiwnej klasyfikacji Bayesa obiekt X przypisany jest do tej klasy, do której prawdopodobieństwo warunkowe przynależności X jest największe. Dlatego X jest przypisany do klasy C_i jeżeli $P(C_i/X) \geqslant P(C_k/X)$, dla każdego k, takiego, że $1 \leqslant k \leqslant m$, gdzie $k \neq i$.

Klasyfikacja Bayesa oparta jest na maksymalizowaniu prawdopodobieństwa:

$$P(C_i/X) = \frac{P(X/C_i) \cdot P(C_i)}{P(X)}, \tag{1.37}$$

gdzie:
$P(X/C_i)$ – prawdopodobieństwo aposteriori, że X należy do klasy C_i,
$P(C_i)$ – prawdopodobieństwo apriori wystąpienia klasy C_i,
$P(X)$ – prawdopodobieństwo apriori wystąpienia przykładu X.

Prawdopodobieństwo $P(X)$ jest stałe, w związku z czym wystarczy maksymalizować iloczyn $P(X/C_i) \cdot P(C_i)$. Ponadto przyjmuje się, że:

$$P(C_i) = s_1/s, \tag{1.38}$$

gdzie:
s – oznacza liczbę obiektów w zbiorze treningowym,
s_i – oznacza liczbę obiektów w klasie C_i.

Dla $X = (x_1, x_2, \ldots, x_n)$ wartość $P(X/C_i)$ oblicza się jako iloczyn:

$$P(X/C_i) = P(x_1/C_i) \cdot P(x_2/C_i) \ldots P(x_n/C_i), \tag{1.39}$$

przy czym:

$$P(x_k/C_i) = s_{ik}/s_i, \tag{1.40}$$

gdzie:
s_{ik} – oznacza liczbę obiektów klasy C_i, dla których wartość atrybutu A_k jest równa x_k,
s_i – oznacza liczbę wszystkich obiektów w klasie C_i w zadanym zbiorze treningowym.

Entropia

Entropia (ang. Entropy) to wielkość $S(p_1, p_2, \ldots, p_n)$ służąca do pomiaru niepewności wystąpienia danego zdarzenia elementarnego w następnej chwili. Niech zbiór

$$A_i = \{A_1, A_2, \ldots, A_n\} \tag{1.41}$$

reprezentuje możliwe wyniki pewnego eksperymentu, tj. możliwe wartości pewnej zmiennej losowej. Natomiast rozkład prawdopodobieństwa

$$P(A_i) = p_i, \qquad \sum_{i=1}^{n} p_i = 1, \tag{1.42}$$

opisuje tę zmienną, gdy wartość średnia tego rozkładu jest znana i wynosi

$$\langle A \rangle = \sum_{i=1}^{n} A_i p_i. \tag{1.43}$$

Entropia S powinna być ciągłą funkcją swoich argumentów $\{p_1, p_2, \ldots, p_n\}$, tj. małe zmiany prawdopodobieństw P_i powinny skutkować małymi zmianami entropii. Dodatkowo powinna być funkcją symetryczną, w tym sensie, że zamiana miejscami p_i oraz p_j , dla $i \neq j$, nie powinna zmieniać wartości tej funkcji.

W przypadku dodania do zbioru wyników $A_i = \{A_1, A_2, \ldots, A_n\}$ nowego wyniku A_{n+1} o prawdopodobieństwie $p_{n+1} = 0$ wartość entropii nie powinna się zmieniać. Ta własność entropii wynika stąd, że zdarzenia niemożliwe o prawdopodobieństwie wystąpienia równym zeru, nie powinny mieć wpływu na prawdziwy wynik eksperymentu.

Gdy wynik eksperymentu jest pewny, tzn. gdy prawdopodobieństwo pewnej wartości A_i wynosi $p_i = 1$, wtedy niepewność $S(p_1, p_2, \ldots, p_n)$ powinna być najmniejsza. Natomiast wartość entropii jest maksymalna wtedy, gdy niepewność związana z wynikiem eksperymentu jest największa. Oczywiście z największą niepewnością mamy do czynienia wtedy, gdy każdy wynik jest tak samo prawdopodobny. Wtedy maksymalna wartość entropii S_{max} powinna być rosnącą funkcją n. Dlatego zasada maksymalnej entropii oznacza, że spośród wielu możliwych rozkładów prawdopodobieństwa $P(A_i)$, unormowanych (1.42) i spełniających warunek (1.43) należy wybrać taki rozkład, który jest obarczony największą niepewnością, w związku z czym:

$$S(p_1, p_2, \ldots, p_n) = -p_i \sum_{i=1}^{n} \ln p_i \tag{1.44}$$

Algorytm Winnow

Winnow to algorytm podobny do perceptronu, lecz z zasady lepiej odrzuca nieistotne elementy. Jest to w efekcie korelator ze sztywnym progiem odrzucania. Normalnie wyjście z tego algorytmu jest binarne – pasuje lub nie. Każda cecha w algorytmie Winnow ma własną wagę.

Detekcja polega na sumowaniu wag wszystkich wykrytych cech zgodnie z wagami dla wszystkich obiektów. Niech k_{ij} będzie i-tą cechą j-tego obiektu. Wtedy:

$$w_j = \sum_{i=0}^{n} w_{ij} \times x_{ij}, \tag{1.45}$$

gdzie:

$$x_{ij} = \begin{cases} 0 \iff k_{ij} \in K_j \\ 1 \iff k_{ij} \notin K_j \end{cases}, \tag{1.46}$$

gdzie w_j to wynikowa waga danego obiektu, w_{ij} to waga danej cechy dla zadanego obiektu, K_j to zbiór cech występujących w j-tym obiekcie. Wynik detekcji to:

$$R_j = \begin{cases} 0 \iff w_j > \Theta \\ 1 \iff w_j \leqslant \Theta \end{cases} \tag{1.47}$$

gdzie Θ to wartość progowa. Dobre wyniki uzyskuje się dla $\Theta = W_j/2$, gdzie W_j to średnia waga j-tego obiektu we wszystkich treningach.

Standardowy algorytm Winnow korzysta z prostej maszyny stanu do treningu. Niech R_j będzie wynikiem detekcji j-tego obiektu, n będzie identyfikatorem pożądanego obiektu, l oznacza liczbę cech przyporządkowanych j-temu obiektowi, $i \in \{1 \dots l\}$, wtedy:

$$n = j \wedge R_j = 0 \Rightarrow w_{ij} = w_{ij} \times \alpha \tag{1.48}$$

$$n \neq j \wedge R_j = 1 \Rightarrow w_{ij} = 0 \tag{1.49}$$

Jeżeli problem jest liniowo separowalny, błąd algorytmu Winnow wynosi maksymalnie $\alpha \log_\alpha(\Theta + 1) + \frac{n}{\Theta}$. Algorytm Winnow2 zawiera następującą modyfikację algorytmu uczącego:

$$n = j \wedge R_j = 0 \Rightarrow w_{ij} = w_{ij} \times \alpha \tag{1.50}$$

$$n \neq j \wedge R_j = 1 \Rightarrow w_{ij} = w_{ij}/\alpha \tag{1.51}$$

Wszystkie warianty algorytmu Winnow ignorują kolejność cech, która nie wpływa na wynik detekcji.

1.5 Algorytmy genetyczne

Pierwsze próby połączenia teorii ewolucji z informatyką przeprowadzono na przełomie lat pięćdziesiątych i sześćdziesiątych. Początkowy brak sukcesów wynikał z naśladownictwa ówczesnych podręczników biologii, które kładły większy nacisk na rolę mutacji jako źródła zmienności genetycznej w porównaniu z reprodukcją płciową. Jednak widocznym przełomem była zaproponowana przez J. H. Hollanda [47] w połowie lat sześćdziesiątych technika programowania uwzględniająca ewolucję zarówno przez mutację, jak i krzyżowanie się. W kolejnych latach zakres stosowania tego algorytmu został poszerzony o kod genetyczny pozwalający reprezentować strukturę każdego problemu. W ten sposób powstał uniwersalny algorytm znany pod nazwą algorytmu genetycznego, który przedstawiony jest w postaci pseudokodu jako alg.1.

Algorytm 1: Pseudokod algorytmu genetycznego

```
1  t = O
2  inicjacja P^0
3  ocena P^0
4  while (not warunek stopu) do
5    T^t = reprodukcja_P^t;
6    O^t = krzyżowanie_i_mutacja_T^t;
7    ocena O^t;
8    P^{t+1}=O^t;
9    t =t + 1;
10 endWhile;
```

Zgodnie z definicją w [47] algorytmy genetyczne to algorytmy poszukiwania, które w rozwiązaniu zadań stosują zasady doboru naturalnego i dziedziczenia, posługują się populacją potencjalnych rozwiązań, zawierają pewien proces selekcji oparty na dopasowaniu osobników oraz zawierają pewne operatory genetyczne. Każde rozwiązanie ocenia się na podstawie pewnej miary jego dopasowania zwane funkcją przystosowania (celu). Im większa jest wartość tej funkcji tym dany osobnik jest lepiej przystosowany. Nową populację w kolejnej iteracji tworzy się przez selekcje osobników najlepiej przystosowanych.

Selekcja polega na wybraniu osobników, które będą brały udział w tworzeniu potomków następnego pokolenia. Wybór ten odbywa się na podstawie obliczonych wartości funkcji przystosowania, zatem największą szansę na wybranie mają osobniki o największej wartości funkcji przystosowania [58].

Istnieje wiele metod selekcji, jednak najbardziej popularną jest metoda ruletki. Polega ona na tym, że każdemu osobnikowi przydzielany jest wycinek koła ruletki o wielkości proporcjonalnej do wartości funkcji przystosowania danego osobnika. Zatem im większa jest wartość funkcji przystosowania, tym większy jest wycinek na kole ruletki. Całe koło ruletki odpowiada sumie wartości funkcji przystosowania wszystkich osobników danej populacji. Każdemu osobnikowi oznaczonemu przez t_i, dla $i = 1, 2, \ldots, n$, gdzie n jest wielkością populacji, odpowiada wycinek koła $v(t_i)$ stanowiący część całego koła, wyrażony w procentach zgodnie ze wzorem:

$$v(t_i) = p_s(t_i) \cdot 100\%, \tag{1.52}$$

przy czym

$$p_s(t_i) = \frac{f(t_i)}{\sum_{i=1}^{n} f(t_i)}, \tag{1.53}$$

gdzie:

$f(t_i)$ oznacza wartość funkcji przystosowania osobnika t_i,

$p_s(t_i)$ jest prawdopodobieństwem selekcji osobnika t_i.

Selekcja może być rozumiana jako obrót kołem ruletki, w wyniku czego zostaje wybrany osobnik należący do wylosowanego w ten sposób wycinka koła ruletki. Prawdopodobieństwo wybrania danego osobnika jest tym większe, im większy jest wycinek koła, czyli im większa jest jego wartość przystosowania. W wyniku procesu selekcji utworzona zostaje populacja rodzicielska zwana też pulą rodzicielską o liczebności takiej samej jak liczebność bieżącej populacji, czyli równej n.

Inną, także popularną metodą selekcji jest metoda rankingowa. Polega ona na tym, że na początku obliczana jest wartość funkcji przystosowania (inaczej funkcja oceny) dla każdego osobnika z populacji. Następnie tworzona jest lista rankingowa osobników, gdzie na początku listy znajdują się osobniki najlepsze, posiadające największą wartość funkcji oceny, a na końcu osobniki najgorsze. No-

wa populacja tworzona jest z osobników najlepszych, natomiast osobniki z końca listy są usuwani.

Krzyżowanie jest to jeden z dwóch podstawowych operatorów stosowanych w algorytmie genetycznym. W klasycznym algorytmie genetycznym krzyżowanie występuje prawie zawsze. Proces krzyżowania polega na wybraniu pary osobników z populacji rodzicielskiej, utworzonej zgodnie z metodą selekcji, a następnie wymianie części informacji zawartej w genach rodziców i utworzeniu potomstwa.

Wybór pary do krzyżowania dokonywany jest w sposób losowy, zgodnie z prawdopodobieństwem krzyżowania p_c, gdzie $0,5 \leqslant p_c \leqslant 1$. Następnie dla każdej pary, wybranych w ten sposób rodziców, losuje się pozycję genu w chromosomie określającą tzw. punkt krzyżowania l_k, który jest liczbą naturalną z przedziału $[1, L-1]$, gdzie L jest liczbą genów w chromosomie każdego z rodziców. W wyniku krzyżowania dwóch osobników z populacji rodzicielskiej otrzymuje się dwóch potomków. Pierwszy potomek ma chromosom składający się z genów na pozycjach od 1 do l_k pierwszego rodzica, oraz genów od l_{k+1} do L pochodzących od drugiego rodzica. Natomiast drugi potomek otrzymuje pozostałe geny.

Mutacja to drugi podstawowy operator genetyczny, jednak prawdopodobieństwo wystąpienia mutacji p_m zawiera się w przedziale od 0 do 0,1. proces mutacji polega na sporadycznej i przypadkowej zamianie wartości genu w chromosomie na wartość przeciwną. Dokonanie mutacji zgodnie z prawdopodobieństwem p_m polega na wylosowaniu liczby z przedziału $[0, 1]$ dla każdego genu i wybraniu do mutacji tych genów, dla których wylosowana liczba jest mniejsza lub równa prawdopodobieństwu p_m. W algorytmie genetycznym mutacja chromosomu może być dokonywana na populacji rodziców przed operacją krzyżowania lub na populacji potomków utworzonych w wyniku krzyżowania.

2

Drzewa decyzyjne oraz algorytmy mrowiskowe

Drzewa decyzyjne (ang. decision trees) są ważnym narzędziem w uczeniu maszynowym i eksploracji danych, a dokładniej w klasyfikacji danych. Posiadają drzewiastą strukturę, która złożona jest z gałęzi oraz węzłów, spośród których szczególnie można wyróżnić korzeń i liście.

Drzewa decyzyjne to acykliczne grafy skierowane, w których wierzchołkami są węzły, a powiązania między węzłami to gałęzie. Węzły odpowiadają testom, które przeprowadzane są na wartościach atrybutów warunkowych, krawędzie odpowiadają wynikom testów, a liście - etykietom kategorii. Wierzchołki bez potomków to liście, a wierzchołek bez rodzica to korzeń.

W celu zaklasyfikowania atrybutu w drzewie, jest on przesuwany odgórnie od węzła głównego (korzenia) w kierunku węzłów liści poprzez węzły wybierane zgodnie z wynikami testów reprezentowanymi przez węzły wewnętrzne, dopóki nie zostanie osiągnięty ostatni węzeł zwany liściem. W tym momencie, etykieta związana z węzłem liścia jest etykietą klasy przewidzianą dla atrybutu.

Podejście do automatycznego konstruowania drzew decyzyjnych opiera się na metodzie „dziel i zwyciężaj", która polega na iteracyjnej procedurze odgórnego wyboru najlepszego atrybutu do znakowania wewnętrznych węzłów drzewa.

W pierwszym kroku wybierany jest atrybut do reprezentowania korzenia drzewa. Po wybraniu pierwszego atrybutu tworzona jest gałąź dla każdej możliwej wartości atrybutu. Zestaw danych dzielony jest na podzbiory według wartości wybranego atrybutu. Procedura selekcji jest następnie stosowana rekurencyjnie do każdego kanału za pomocą węzła odpowiedniego podzbioru atrybutów, które mają podobne wartości. Obiekty z podzbioru mają tę samą etykietę klasy lub gdy inne kryterium zatrzymania jest spełnione, tworzą węzeł liści do reprezentowania etykiety klasy, którą należy przewidzieć.

2.1 Algorytmy do konstruowania drzew decyzyjnych

W literaturze istnieje wiele algorytmów do konstruowania drzew decyzyjnych. Niektóre z nich powstały w osiemdziesiątych latach dwudziestego wieku, ale ze względu na swoją prostotę i popularność są one stosowane do dnia dzisiejszego. Do takich algorytmów należą algorytm ID3, algorytm C 4.5 oraz algorytm CART.

Algorytm ID3

Algorytm ID3 to jeden z najprostszych algorytmów tworzących drzewa decyzyjne. Algorytm pochodzi z 1986 r., a jego twórcą jest Ross Quinlan. Alg. 2 przedstawia pseudokod algorytmu konstrukcji drzewa decyzyjnego ID3 [63].

Algorytm 2: Pseudokod algorytmu ID3

```
1  Wejście: zbiór treningowy D, zbiór atrybutów warunkowych A, metoda wyboru punktu
   podziału SS.
2  Wyjście: drzewo decyzyjne ukorzenione w wierzchołku N.
3
4  procedure BuildTree(D,A,SS):
5  utwórz wierzchołek drzewa decyzyjnego N;
6  if wszystkie rekordy zbioru D należą do tej samej klasy C then
7      return wierzchołek N jako liść drzewa decyzyjnego i przypisz temu wierzchołkowi
8      etykietę klasy C;
9  endIf
10 if lista_atrybutów_A jest pusta then
11     return wierzchołek N jako liść drzewa decyzyjnego i przypisz temu wierzchołkowi
12     etykietę klasy dominujące w zbiorze treningowym D;
13 endIf
14 zastosuj metodę SS w celu wybrania atrybutu − podziaowego ze zbioru A;
15 przypisz wierzchołkowi N etykietę atrybutu-podziałowego;
16 for all wartości a_i atrybutu-podziałowego do
17     S_i – zbiór rekordów D o wartości atrybutu-podziałowego=a_i ;
18     N_i – BuildTree(S_i, (lista_atrybutów A) - (atrybut podziałowy), SS);
19     utwórz krawędź z N do N_i etykietowaną wartością a_i;
20 endFor
21 return wierzchołek N;
```

Algorytm ID3 działa w oparciu o zasadę maksimum zysku informacyjnego, a także preferuje prostsze drzewa, gdyż bazuje na najprostszych hipotezach. Algorytm tworzenia drzewa decyzyjnego w ID3 wygląda następująco:

- wybrana zostaje pewna cecha X, która najlepiej odróżnia próbki (daje największy zysk informacji);
- cecha ta staje się kryterium podziału w korzeniu drzewa;
- dla każdej wartości y_i cechy X stworzona zostaje nowa gałąź na podstawie testu $X = y_i$;

- algorytm jest powtarzany w dół – tworzy kolejne węzły dla gałęzi wygenerowanych w poprzednim kroku.

Zaletą ID3 oprócz jego prostoty jest to, iż w przypadku braku szumu w zestawie testowym, ID3 daje poprawny wynik dla wszystkich kategorii z zestawu treningowego. Ten prosty algorytm nie radzi sobie z ciągłymi dziedzinami atrybutów (zakłada, że wartości atrybutów są dyskretne), ani tym bardziej w przypadku niepełnych danych. W ID3 nie jest stosowane przycinanie drzewa, stąd też mimo stosowania prostych hipotez istnieje ryzyko przerostu drzewa.

Algorytm C4.5

Algorytm C4.5 zaproponowany przez Quinlana [72] jest prawdopodobnie najbardziej znanym algorytmem do konstruowania drzew decyzyjnych, który stosuje kryterium entropii opartej na wyborze najlepszego atrybutu do utworzenia węzła. Algorytm C4.5 jest udoskonaloną wersją wcześniejszego algorytmu ID3 [71].

W porównaniu do algorytmu ID3 poprawione zostało m.in. kryterium podziału, tak aby uzyskiwane podziały dla większych zbiorów danych generowały mniejszy błąd klasyfikacji i możliwa była klasyfikacja obiektów z brakującymi wartościami atrybutów. W algorytmie ID3 jako kryterium podziału stosowana jest reguła zysku informacji:

$$zyskInf(a_i, S) = entropia(y, S) - \sum_{k=1}^{K} \frac{|S_k|}{|S|} \cdot entropia(y, S_k) \tag{2.1}$$

natomiast w C4.5 reguła względnego zysku:

$$\underset{a_j \leqslant a_j^R, j=1,\ldots,M}{\arg\max} \left(\frac{zyskInf(a_i, S)}{entropia(a_i, S)} \right), \tag{2.2}$$

gdzie:

$zyskInf(a_i, S)$ jest zyskiem informacji (2.1),

$entropia(a_i, S)$ jest entropią rozkładu danych ze zbioru S na podstawie wartości atrybutu a_i zgodnie ze wzorem:

$$entropia(y, S) = \sum_{j=1}^{|y|} -\frac{|S_j|}{|S|} \cdot \log_2 \frac{|S_j|}{|S|}. \tag{2.3}$$

Ponadto w algorytmie C4.5 wprowadzono przycinanie drzewa. Początkowo była to podstawowa metoda przycinania pesymistycznego (ang. pessimistic pruning), która następnie podlegała stopniowym udoskonaleniom (ang. error-based pruning). Podczas procesu uczenia się oraz klasyfikacji istnieje możliwość pracy z obiektami nie posiadającymi wartości wszystkich atrybutów (dane z brakującymi wartościami atrybutów), dodatkowo algorytm C4.5 dostosowany jest do pracy z ciągłymi wartościami atrybutów [73].

Algorytm CART

Algorytm CART (ang. Classification and Regression Trees) to algorytm stosowany do konstruowania drzew decyzyjnych. Po raz pierwszy algorytm ten został zaproponowany przez Breimana i innych w 1984 r. [17]. Dla algorytmu CART, Breiman i in. zaproponowali dwa kryteria podziału: Giniego oraz podziału na dwie części.

Kryterium podziału ma za zadanie znalezienie najlepszego testu, który podzieli dane analizowane w węźle na dwie, maksymalnie jednorodne (pod względem klasy decyzyjnej) części. Jest to zdecydowanie najtrudniejszy i najbardziej złożony etap konstruowania drzew decyzyjnych.

Kryterium Giniego oparte zostało na indeksie Giniego, czyli mierze koncentracji zmiennej losowej. Nadrzędnym celem w tym przypadku jest dokonanie podziału na możliwie jednorodne przypadki w węzłach potomnych. Warunek, według którego dokonuje się podziału jest wyznaczany na podstawie wzoru:

$$\underset{a_j \leqslant a_j^R, j=1,\ldots,M}{\arg\max} \left(-\sum_{k=1}^{K} p^2(k|m_p) + P_l \sum_{k=1}^{K} p^2(k|m_l) + P_r \sum_{k=1}^{K} p^2(k|m_r) \right) \tag{2.4}$$

gdzie:

$p(k|m_p)$ – prawdopodobieństwo wystąpienia klasy decyzyjnej k w węźle m_p,
$p(k|m_l)$ – prawdopodobieństwo wystąpienia klasy decyzyjnej k w węźle m_l,
$p(k|m_r)$ – prawdopodobieństwo wystąpienia klasy decyzyjnej k w węźle m_r,
P_l – prawdopodobieństwo przejścia obiektu do węzła m_l (lewe poddrzewo),
P_r – prawdopodobieństwo przejścia obiektu do węzła m_r (prawe poddrzewo),
K – liczba wartości klas decyzyjnych.

Kryterium podziału na dwie części (ang. twoing rule) przede wszystkim dokonuje podziału danych na dwie możliwie równe części. Jednorodność klasy decyzyjnej jest w tym przypadku mniej znacząca niż podczas stosowania kryterium Giniego, choć odgrywa pewną rolę. Warunek, według którego dokonuje się podziału jest wyznaczany na podstawie wzoru:

$$\underset{a_j \leqslant a_j^R, j=1,\ldots,M}{\arg\max} \left(\frac{P_l P_r}{4} \left[\sum_{k=1}^{K} |p(k|m_l) - p(k|m_r)| \right]^2 \right). \tag{2.5}$$

Dokładna interpretacja obydwu kryteriów oraz statystyczne uzasadnienie takiego rozwiązania podane zostało przez Breimana i in. w [17]. Alg. 3 przedstawia pseudokod algorytmu konstrukcji drzewa decyzyjnego CART [63].

Algorytm 3: Pseudokod algorytmu CART

```
1  Wejście: zbiór treningowy D, lista_atrybutów A, metoda wyboru punktu podziału SS.
2  Wyjście: drzewo decyzyjne ukorzenione w wierzchołku N.
3
4  procedure BuildTree(D,A,SS):
5  utwórz wierzchołek drzewa decyzyjnego N;
6  if wszystkie rekordy zbioru D należą do tej samej klasy C then
7      return wierzchołek N jako liść drzewa decyzyjnego i przypisz temu wierzchołkowi
8      etykietę klasy C;
9  endIf
10 if lista_atrybutów_A jest pusta then
11     return wierzchołek N jako liść drzewa decyzyjnego i przypisz temu wierzchołkowi
12     etykietę klasy dominujące w zbiorze treningowym D;
13 endIf
14 zastosuj metodę SS w celu wybrania kryterium podziałowego;
15 przypisz wierzchołkowi N etykietę kryterium podziałowego;
16 zastosuj kryterium podziałowe do podziału zbioru D na partycje D_1 i D_2;
17 S_i – zbiór rekordów D należących do partycji D_i;
18 N_i – BuildTree(S_i, (lista_atrybutów A) - (atrybut podziałowy), SS);
19 utwórz krawędź z N do N_i etykietowaną wartością kryterium podziałowego;
20 return wierzchołek N;
```

2.2 Zespoły klasyfikatorów

Zespół klasyfikatorów (ang. Ensamble Method, EM), zwany również rodziną klasyfikatorów, to połączone ze sobą pojedyncze klasyfikatory, które poprzez wielokrotne uruchomienie algorytmu budują wiele hipotez na podstawie zróżnicowanych próbek danych. Ostateczna decyzja zespołu klasyfikatorów podejmowana jest na podstawie decyzji określonych przez wszystkie pojedyncze klasyfikatory. Istnieją dwa główne podejścia do projektowania algorytmów opartych na zespołach klasyfikatorów. Pierwsze podejście to skonstruowanie każdej hipotezy niezależnie w taki sposób, że uzyskany zestaw hipotez jest dokładny i zróżnicowany. Drugie podejście do projektowania zespołów jest skonstruowanie hipotez połączonych w taki sposób, że ważony głos hipotezy daje dobre dopasowanie do danych.

Definicja zespołu klasyfikatorów wyraża się wzorem:

$$EM = \{d_j : X \to \{1, 2, ..., g\}\}_{j=1,2,...,J}, \tag{2.6}$$

gdzie J jest liczbą pojedynczych klasyfikatorów j ($J \geqslant 2$).

W metodzie opartej na zespołach klasyfikatorów, klasyfikacja odbywa się za pomocą prostego głosowania. Każdy klasyfikator głosuje na jedną decyzję dotyczącą danej próbki, jednak ostatecznie zostaje wybrana decyzja z największą liczbą głosów. Klasyfikator tworzący zespół klasyfikatorów EM, oznaczony jako $dEM : X \to 1, 2, ..., g$, posługuje się następującą zasadą głosowania:

$$dEM(x) := \arg\max_k N_k(x), \tag{2.7}$$

gdzie $N_k(x)$ jest liczbą głosów dla próbki $x \in X$ klasyfikacji w klasie k, tak, że $N_k(x) := \#\{j : d_j(x) = k\}$, natomiast k to klasa decyzji, taka, że $k \in \{1, 2, \ldots, g\}$.

Prace nad zespołem klasyfikatorów rozpoczęto w latach 90-tych XX wieku. Pierwszą i najprostszą rodziną klasyfikatorów jest metoda bagging [35]. Metoda ta została przedstawiona przez Leo Breimana [15] w 1996 roku. Kolejna metoda to boosting [82], która jest ulepszeniem metody bagging, choć powstawała niezależnie od niej. Pierwszymi, którzy zastanawiali się nad możliwością wzmocnienia „słabego" algorytmu uczącego, byli Michael Kearns i Leslie Valiant [50]. Natomiast Yoav Freund i Robert Schapire [40] w 1995 roku przedstawili algorytm boostingu, który pozwolił już rozwiązać większość praktycznych problemów jakimi były obciążone wcześniejsze jego wersje.

Boosting

Dyskretny adaptacyjny algorytm uczący tzw. boosting to metoda zaproponowana przez Schapire'a w 1990 roku [82] zainspirowanego przez Kearnsema [50], a dokładniej jego propozycją związaną z tym, aby na podstawie słabych zbiorów uczących spróbować utworzyć dobry zbiór uczący. Nowsza wersja boostingu (dyskretny adaptacyjny boosting – algorytm AdaBoost) powstała ok. 1995 roku i zaprezentowana została przez Freunda i Schapirea m.in. w publikacjach z 1996 i 1997 roku [40, 41]. Algorytm ten jest w dalszym ciągu rozwijany przez Schapirea, co zostało zaprezentowane m.in. w [78].

Do każdego elementu zbioru uczącego przypisywana jest waga (początkowo równa $\frac{1}{n}$), która określa prawdopodobieństwo, z jakim element powinien zostać wylosowany do pseudopróby. Następnie budowany jest klasyfikator. W kolejnym etapie, dla elementów, które przypisane zostały do błędnej klasy decyzyjnej zwiększona zostaje waga, a więc podczas tworzenia kolejnej pseudopróby zwiększone zostaje prawdopodobieństwo wyboru elementów, które wcześniej zostały źle sklasyfikowane. W algorytmie AdaBoost wagi w_i elementów zbioru uczącego (początkowo równe $\frac{1}{n}$) modyfikowane są w zależności od błędu klasyfikacji uzyskanego przez powstały klasyfikator. Błąd klasyfikacji klasyfikatora j, to suma wag elementów, które zostały przez niego źle sklasyfikowane:

$$\epsilon(j) = \sum_{x_i} w_i[k_i \neq k_i^j], \tag{2.8}$$

gdzie we_i, to waga obiektu (elementu, przykładu) x_i, a k_i^j to klasa decyzyjna, do której sklasyfikowano obiekt x_i. Jeśli błąd klasyfikacji jest mniejszy lub równy 0,5 następuje odpowiednia modyfikacja wagi w_i, w przeciwnym przypadku wagi mnożone są przez współczynnik zapisany wzorem 2.9 i normalizowane.

$$\kappa(j) = \frac{1 - \epsilon(j)}{\epsilon(j)} \tag{2.9}$$

Bagging

Bagging (ang. Bootstrap Aggregating [35]) jest jedną z pierwszych rodzin klasyfikatorów bazującej na bootstrapowej agregacji, zaproponowaną przez Breimana [15] w 1996 r. Metoda ta często pozwala poprawić klasyfikację oraz modele regresyjne pod względem stabilności i dokładności poprzez obniżenie wariancji. Polega na wielokrotnym budowaniu klasyfikatora na podstawie zbioru próbek bootstrapowych utworzonych z całego zbioru treningowego.

Biorąc pod uwagę zestaw danych treningowych n, bagging wybiera w każdej iteracji zbiór uczący o rozmiarze n przez próbkowanie równomierne z pełną wymianą z oryginalnego zbioru danych. Każdy element takiego zbioru może zostać wybrany dokładnie z tym samym prawdopodobieństwem równym $\frac{1}{n}$. Zakładając, że decyzje klasyfikatorów bazowych są niezależne od siebie, każdy z klasyfikatorów bazowych oddaje dokładnie jeden głos, a decyzja o klasyfikacji próbki oparta jest na prostym głosowaniu, zgodnie ze wzorem (6.1).

Lasy losowe

Niektóre zespoły klasyfikatorów, takich jak lasy losowe są szczególnie przydatne do wysoko wymiarowych zbiorów danych z powodu zwiększonej dokładności klasyfikacji. Można to osiągnąć poprzez generowanie wielu modeli predykcyjnych, każdy z innego podzbioru danych treningowych składających się z podzbiorów atrybutów [16].

Breiman zapewnia ogólne ramy dla zespołów drzewa o nazwie „lasy losowe" [16]. Każde drzewo, zależy od wartości losowo wybranych atrybutów, niezależnych dla każdego węzła lub drzewa z tej samej dystrybucji dla wszystkich drzew. W ten sposób losowy las jest klasyfikatorem (zespołem), który składa się z wielu drzew decyzyjnych. Każda reguła dzielenia jest wykonywana niezależnie dla różnych podzbioru atrybutów. W efekcie może zostać wybranych m atrybutów p próbek uczących. Zakładając, że $m \ll p$, i zgodnie z przeprowadzonymi doświadczeniami, dobre wyniki uzyskuje się, gdy $m = \sqrt{p}$. Zakładając, że$\frac{1}{3}$ próbek nie może być wybrana do próbki uczącej (zgodnie z prawdopodobieństwem równym $(1-n)^n \approx e^{-n}$), więc tylko $\frac{1}{3}$ drzew w analizowanym lesie będą konstruowane bez tej próby. W tej sytuacji, Breiman zaproponował, że dobrym rozwiązaniem będzie zastosowanie nieograniczonego estymatora prawdopodobieństwa błędnej klasyfikacji uzyskanej przez drzewa decyzyjne [16].

2.3 Algorytmy mrowiskowe

Algorytmy mrowiskowe (ang. Ant Colony Optimization, ACO) stanowią metaheurystyczne podejście do rozwiązywania wielu problemów optymalizacyjnych wykorzystując idee zachowań komunikacyjnych występujące w koloniach mró-

wek. Inteligentne zachowanie kolonii mrówek wynika z komunikacji pośredniej pomiędzy mrówkami poprzez niewielkie modyfikacje środowiska, które nazywa się stygmergią.

Wiele gatunków mrówek, nawet z ograniczonymi możliwościami wizualnymi lub całkowicie niewidomych, znajduje najkrótszą ścieżkę między mrowiskiem a źródłem pokarmu za pomocą feromonu, jako mechanizmu komunikacji. Podczas poszukiwania pożywienia mrówki tworzą ścieżki, na których odkładają ślad feromonowy. Pozwala im to na szybki powrót do mrowiska i przekazanie informacji innym mrówkom o miejscu, w którym znajduje się pożywienie. Stężenie feromonów na ścieżce wpływa na wybór dokonywany przez mrówki, dla których ścieżka staje się bardziej atrakcyjna, gdy ma większe stężenie feromonów. Ostatecznie dzięki oddziaływaniu sprzężenia zwrotnego tworzone są najkrótsze ścieżki łączące mrowisko z pożywieniem, na których odłożona jest duża wartość śladu feromonowego.

Chęć poznania, w jaki sposób owady takie jak mrówki są w stanie odnaleźć najkrótszą drogę z mrowiska do pożywienia była pierwszą inspiracją do powstania algorytmów mrowiskowych (ACO). Badania i eksperymenty wykonywane przez S. Goss, J. L. Deneubourg i innych opisane w pracach [9, 85, 88], dotyczące zrozumienia sposobu realizacji tego zadania przez naturę, były pierwszym krokiem do zaimplementowania tego rozwiązania w algorytmice. Jednak dopiero podjęte przez M. Dorigo [25, 32, 31, 29, 30, 33, 34] próby stworzenia sztucznego systemu mrówkowego oraz zastosowania go do znalezienia najkrótszej drogi pomiędzy wierzchołkami dla zadanego grafu były kluczowym krokiem do powstania algorytmów ACO.

Algorytm ACO jest algorytmem populacyjnym, w którym kolejne generacje wirtualnych agentów-mrówek poszukują rozwiązań dobrej jakości. Agenty-mrówki w każdej iteracji algorytmu kierują się śladem feromonowym pozostawionym przez agenty-mrówki w poprzednich iteracjach. Dzięki temu przeszukiwanie przestrzeni rozwiązań skupia się na obszarach zawierających odkryte wcześniej rozwiązania dobrej jakości.

Zastosowanie algorytmów ACO w kontekście eksploracji danych koncentruje się na odkrywaniu zasad klasyfikacji, co przedstawione zostało w pracach [39, 56], gdzie w przeprowadzonych badaniach stosowany jest algorytm Ant-Miner [68] z wieloma odmianami. Algorytm ACO opisany w pracy [66] buduje drzewo decyzyjne zamiast zbioru zasad, co bardzo różni się od Ant-Miner i jego odmian.

Izrailev i Agrafiotis w swojej pracy [48] zaproponowali metodę opartą na kolonii mrówek do budowy drzew regresyjnych. Regresja polega na znalezieniu modelu, który mapuje dane wejściowe na przewidywane wartości (czyli docelowy atrybut przyjmuje wartości ciągłe), podczas gdy klasyfikacja polega na znalezieniu modelu, który mapuje podane dane do jednego z predefiniowanego zbioru etykiet dyskretnych lub nominalnych klas (czyli docelowy atrybut przyjmuje dyskretne

lub nominalne wartości). Drzewo regresyjne może być postrzegane jako szczególny przypadek drzewa decyzyjnego, gdzie wartość przewidywana dla atrybutu docelowego w każdym węźle liści drzewa jest wartością ciągłą zamiast wartością dyskretną lub nominalną. W prezentowanej metodzie, mrówka reprezentuje drzewa regresyjne, a macierz feromonowa jest reprezentowana przez drzewo binarne odpowiadające topologii wszystkich zbudowanych drzew. W związku z tym, przedstawiony sposób ma dwa istotne ograniczenia. Po pierwsze, używa tylko ciągłych atrybutów dotyczących węzłów decyzyjnych (tj. nie może poradzić sobie z dyskretnym lub nominalnym atrybutem – predyktorem). Po drugie, nie korzysta z funkcji heurystycznych, które są powszechnie stosowane w algorytmach ACO.

2.3.1 Algorytm mrowiskowy do konstruowania drzew decyzyjnych

Jednym z algorytmów mrowiskowych stosowanych w eksploracji danych jest algorytm Ant Colony Decision Tree (ACDT). Algorytm ten łączy idee algorytmów mrowiskowych oraz algorytmu CART i jak pokazały badania osiąga on bardzo dobrej jakości klasyfikatory dla wielu standardowych problemów z dziedziny eksploracji danych [11]. Algorytm ACDT oparty jest na zastosowaniu algorytmów mrowiskowych w procesie optymalizacji budowy drzew decyzyjnych. Wykonywanie algorytmu polega na wyborze testu dla każdego węzła, na podstawie dwóch czynników. Pierwszym czynnikiem jest maksymalna wartość zgodna z kryterium podziału algorytmu CART, a drugim dodatkowa informacja zapisana w postaci śladu feromonowego [11, 12].

Podczas pracy algorytmu każdy agent-mrówka w populacji konstruuje drzewo decyzyjne. Po wykonaniu pracy przez całą populację, mrówka z najlepszym drzewem odkłada feromon, co zapisane zostało w linii 12. Feromon odkładany jest dla każdego podziału wybranego podczas konstruowania drzewa, wraz z informacją o podziale dokonanym w węźle nadrzędnym. Dzięki takiemu zastosowaniu algorytm stara się budować kolejne drzewa z uwzględnieniem struktury poprzednich drzew, modyfikując pojedyncze węzły. Wynikiem pracy algorytmu jest najlepsze drzewo decyzyjne.

Wartość funkcji heurystycznej wyznaczana jest na podstawie kryterium podziału stosowanego w algorytmie CART, zgodnie ze wzorem:

$$\underset{a_j \leqslant a_j^R, j=1,\ldots,M}{\arg\max} \left(\frac{P_l P_r}{4} \left[\sum_{k=1}^{K} |p(k|m_l) - p(k|m_r)| \right]^2 \right), \tag{2.10}$$

gdzie:
$p(k|m_l)$ – prawdopodobieństwo wystąpienia klasy decyzyjnej k w węźle m_l,
$p(k|m_r)$ – prawdopodobieństwo wystąpienia klasy decyzyjnej k w węźle m_r,
P_l – prawdopodobieństwo przejścia obiektu do węzła m_l (lewego poddrzewa),

P_r – prawdopodobieństwo przejścia obiektu do węzła m_r (prawego poddrzewa),
K – klasy decyzyjne.

Natomiast prawdopodobieństwo wyboru testu w węźle (wzór (2.11)) jest standardowym prawdopodobieństwem wykorzystywanym w systemach mrowiskowych.

$$p_{i,j} = \frac{\tau_{m,m_{L(i,j)}}(t)^{\alpha} \cdot \eta_{i,j}^{\beta}}{\sum_{i}^{a} \sum_{j}^{b_i} \tau_{m,m_{L(i,j)}}(t)^{\alpha} \cdot \eta_{i,j}^{\beta}}, \tag{2.11}$$

gdzie:

$\eta_{i,j}$ – współczynnik informacji heurystycznej dla testu atrybutu i o wartości j,
$\tau_{m,m_{L(i,j)}}$ – ślad feromonowy w czasie t dla krawędzi prowadzącej od węzła m do węzła m_L (dla testu atrybutu i o wartości j),
α i β – parametry określające względną wagę funkcji heurystycznej i śladu feromonowego.

Wartość początkowa śladu feromonowego jest taka sama dla wszystkich gałęzi i jest zależna od liczby klas decyzyjnych oraz wartości wszystkich atrybutów. Natomiast aktualizacja feromonu (wzór (2.12)) polega na odłożeniu pewnej wielkości feromonu przez najlepsze drzewo w populacji oraz odpowiednim odparowaniu. Podczas odkładania feromonu wartość dla każdej pary węzłów (rodzic – potomek) zwiększana jest o wartość odpowiadającą jakości drzewa decyzyjnego:

$$\tau_{m,m_L}(t+1) = (1-\gamma) \cdot \tau_{m,m_L}(t) + Q, \tag{2.12}$$

gdzie:

Q oznacza ocenę jakości drzewa decyzyjnego (wzór (2.13)),
γ jest parametrem określającym prędkość wyparowywania feromonu.

$$Q(T) = \phi \cdot w(T) + \psi \cdot a(T,P), \tag{2.13}$$

gdzie:

$w(T)$ – wielkość (liczba węzłów) drzewa T,
$a(T,P)$ – dokładność klasyfikacji obiektów ze zbioru testowego P przez drzewo T,
ϕ i ψ – stałe określające względną ważność wartości $w(T)$ i $a(T,P)$.

Najważniejszymi regułami zachowania agenta-mrówki są reguły aktualizacji śladu feromonowego i funkcja przejścia między stanami. Każda decyzja dotycząca wyboru kolejnego kroku podejmowana jest przez sztuczną mrówkę zgodnie ze wzorem:

$$j = \begin{cases} \arg\max\limits_{r \in J_i^k} \{[\tau_{ir}(t)] \cdot [\eta_{ir}]^{\beta}\}, & \text{jeśli } q \leqslant q_0 \\ p_{ij}^k(t), & \text{w przeciwnym razie,} \end{cases} \tag{2.14}$$

gdzie:

η_{ir} – wartość heurystycznie oszacowanej jakości przejścia ze stanu i do

stanu r,
τ_{ir} – wartość nagrody, czyli stopień użyteczności branej pod uwagę decyzji,
β – parametr określający ważność wartości η_{ir},
$p_{ij}^k(t)$ – kolejny krok (decyzja) wylosowana z zastosowaniem prawdopodobieństw:

$$p_{ij}^k(t) = \begin{cases} \frac{\tau_{ij}(t)\cdot[\eta_{ij}]^\beta}{\sum_{r\in J_i^k}\tau_{ir}(t)\cdot[\eta_{ir}]^\beta}, & \text{jeśli } j \in J_i^k \\ 0, & \text{w przeciwnym razie,} \end{cases} \tag{2.15}$$

gdzie J_i^k jest zbiorem decyzji, jakie mrówka k może podjąć będąc w stanie i. Po przejściu całej trasy przez pojedynczą sztuczną mrówkę, nakładany jest ślad feromonowy na każdej odwiedzonej krawędzi (i, j). Niech τ_{ij} oznacza wielkość śladu feromonowego na krawędzi (i, j):

$$\tau_{ij}(t+1) = (1-\rho)\cdot\tau_{ij}(t) + \rho\cdot\tau_0, \tag{2.16}$$

gdzie ρ to współczynnik wyparowania śladu, $0 \leqslant \rho \leqslant 1$, natomiast τ_0 to wartość równa śladowi inicjalnemu. Po upływie całej iteracji algorytmu składającej się z n cykli, na krawędziach należących do tejże trasy, wielkość śladu feromonowego jest modyfikowana zgodnie ze wzorem (2.17) w celu znalezienia optymalnego rozwiązania ${L_k}^{-1}$.

$$\tau_{ij}(t,t+n) = (1-\alpha)\cdot\tau_{ij}(t) + \Delta\tau_{ij}(t,t+n), \qquad dla\ \Delta\tau_{ij}(t,t+n) = {L_k}^{-1}, \tag{2.17}$$

gdzie α to współczynnik zwany poświatą feromonu, taki, że $(1-\alpha)$ reprezentuje wyparowanie feromonu.

2.3.2 Algorytm mrowiskowy do konstruowania lasów decyzyjnych

Algorytm mrowiskowy do konstruowania lasów decyzyjnych (ang. Ant Colony Decision Forest, ACDF) jest oparty na dwóch rozwiązaniach: zespołów klasyfikatorów opisanych w rozdziale 2.2 oraz algorytmu mrowiskowego do konstruowania drzew decyzyjnych (ang. Ant Colony Decision Trees, ACDT) zaprezentowanego w rozdziale 6.1. Algorytm ACDF może być stosowany do analizy trudnych zbiorów danych poprzez dodanie losowości w procesie wyboru, gdzie zestawy cech lub atrybutów będą różne podczas budowy drzew decyzyjnych [13].

W przypadku algorytmu ACDF, agenty-mrówki tworzą zbiór hipotez w sposób przypadkowy. Podjęte wyzwanie polega na wprowadzeniu nowej metody losowości podprzestrzeni do budowy zbiorów drzew decyzyjnych - oznacza to, że agent mrówki mogą tworzyć zbiór hipotez z przestrzeni hipotez za pomocą zasady losowej proporcji. W każdym węźle drzewa agent-mrówka może wybierać spośród losowego podzbioru (pseudo-losowych próbek) atrybutów, a następnie ograniczyć hipotezę wyboru do tego podzbioru. Ze względu na losowość w przedstawionym podejściu zaproponowano rezygnację z posiadania różnych podgrup

cech wskazanych dla każdego agenta-mrówki lub kolonii na rzecz większej stabilności podejmowanych hipotez. Jest to konsekwencja propozycji użycia metody lasów losowych.

Oryginalne prawdopodobieństwo wyboru było równe $\frac{1}{n}$. W następującej populacji wirtualnych mrówek wartość tego prawdopodobieństwa zależy od wagi obiektu. W przypadku błędnej klasyfikacji, współczynnik ten zostanie zwiększony zgodnie z wzorem:

$$we_i = \begin{cases} 1, & \text{jeśli obiekt jest dobrze sklasyfikowany} \\ 1 + \lambda \cdot n, & \text{w przeciwnym przypadku,} \end{cases} \tag{2.18}$$

Natomiast, prawdopodobieństwo wyboru obiektu x_i obliczono według wzoru:

$$pp(x_i) = \frac{we_i}{\sum_{j=1}^{n} we_j}. \tag{2.19}$$

3

Sieci społeczne

Sieć (ang. Network) to reprezentacja struktury składającej się z dwóch elementów – węzłów oraz powiązań, które określają relacje pomiędzy tymi węzłami. Natomiast sieci społeczne (ang. Social Network, SN) to sieci, w których węzłami są osoby lub grupy osób (np. zespoły, organizacje), a powiązania między tymi osobami dotyczą np. relacji znajomości, komunikacji czy zależności w pracy.

Analiza sieci społecznej (ang. Social Network Analysis, SNA) to interdyscyplinarne podejście badawcze dostarczające technik analizy danych o charakterze relacyjnym, w którym podstawowym sposobem reprezentacji rzeczywistości jest sieć. Jest ono tworzone i rozwijane na pograniczu teorii grafów, algebry macierzowej, informatyki i statystyki, a wykorzystywane m.in. w ekonomii, socjologii, fizyce i biologii [92].

Zainteresowanie tak różnych dyscyplin naukowych analizą sieciową jest związane z coraz powszechniejszą koniecznością prowadzenia badań procesów masowych ujawniających obecnie swe znaczenie już nie tylko w systemach naturalnych czy technicznych, lecz także ekonomicznych, prawnych czy poznawczych.

3.1 Definicja sieci

Sieć społeczna (ang. Social Network) to wielowymiarowa struktura złożona ze zbioru jednostek społecznych oraz połączeń między nimi. Jednostki społeczne to osoby funkcjonujące w danej sieci, natomiast połączenia odwzorowują różnorodne relacje społeczne pomiędzy poszczególnymi osobami. Pierwsze badania sieci społecznych przeprowadził w 1923 r. Jacob L. Moreno, który uznawany jest za jednego z założycieli dyscypliny analizy sieci społecznych. Jest to gałęź socjologii, która zajmuje się ilościową oceną roli jednostki w grupie lub społeczności przez analizę sieci powiązań między jednostkami. Jego książka „Who Shall Survive?" z 1934 r. zawiera pierwsze graficzne przedstawienia sieci społecznych, a także

definicje kluczowych terminów w analizie sieci społecznych i sieci socjometrycznych [59, 60].

Sieć społeczną przedstawia się w postaci grafu. Zgodnie z matematyczną definicją, graf to uporządkowana para, taka, że:

$$G = (V, E), \tag{3.1}$$

gdzie V jest skończonym zbiorem wierzchołków grafu $V = \{1, \ldots, n\}$, przy czym $\overline{\overline{V}} \geqslant 2$, natomiast E jest skończonym zbiorem wszystkich dwuelementowych podzbiorów zbioru V zwanych krawędziami, łączącymi poszczególne wierzchołki, takim, że:

$$E \subseteq \{\{u, v\} : u, v \in V, u \neq v\}. \tag{3.2}$$

W grafie wierzchołki reprezentują obiekty, natomiast krawędzie obrazują relacje między tymi obiektami. W zależności od tego czy relacja ta ma charakter symetryczny, czy też nie, graf wykorzystywany do opisu sieci może być grafem nieskierowanym lub grafem skierowanym. Krawędzie w sieci społecznej reprezentują interakcję, przepływ informacji i dóbr, podobieństwo, afiliację lub związki społeczne. Miarami siły powiązania są częstotliwość, wzajemność oraz rodzaj interakcji lub przepływu informacji, ale także siła powiązania zależna jest od atrybutów łączonych węzłów (np. stopień pokrewieństwa) oraz struktury sąsiedztwa tych węzłów (np. liczba wspólnych sąsiadów).

3.2 Modele sieci

Modele sieci służą do przedstawiania i badania relacji między obiektami. Pierwsze modele powstały prawdopodobnie już w XVIII w. Są to przede wszystkim Model Bernoulliego, Model Erdosa-Renyiego, a także model dotyczący problemu mostów królewieckich. Prawie trzysta lat później powstał model „małego świata" oraz model Barabasi-Alber dotyczący sieci WWW.

Model Bernoulliego – pierwsze próby sieci

Znany szwajcarski matematyk, Johan Bernoulli, był pierwszą osobą, która wyraziła odsetek osób podatnych na zakażenie endemiczne pod względem siły zakażenia i długości życia i w tym celu przyjął dane opisujące epidemię czarnej ospy we Wrocławiu. Bernoulli oparł się w swojej pracy na danych zebranych przez Edmunda Halley'a [45]. Z zachowaniem wszystkich wymagań dokładności i szczerości, zapisane tam zostały dane dotyczące wieku i płci każdego zmarłego w danym miesiącu, które porównano z liczbą urodzin z pięciu lat (od 1687 do 1691). Głównym celem Bernoulli'ego było policzenie skorygowanej tabeli życia. Na podstawie swoich rozważań Bernoulli skonstruował tabelę opisującą zmiany populacji we Wrocławiu i otrzymał średnią długość życia w stanie naturalnym jako 26 lat i 7 miesięcy, a dla stanu bez ospy 29 lat i 9 miesięcy.

Problem mostów królewieckich

W 1736 r. Leonhard Euler zainteresował się problemem mieszkańców Królewca (miasta przedzielonego rzeką na dwie wyspy połączone siedmioma mostami), którzy zastanawiali się czy można tak dobrać trasę spaceru, aby po wyjściu z domu przejść przez każdy z mostów dokładnie jeden raz, a następnie powrócić do punktu wyjścia. Euler wykazał, że rozwiązanie tego problemu jest niemożliwe, co opisał w pracy [37] jako problem mostów królewieckich, używając przy tym pierwszy raz specyficznego modelu, który nazwany został grafem.

Model Erdosa-Renyiego - sieć losowa

W 1959 r. Paul Erdös i Alfréd Rényi zaproponowali model grafu losowego (model ER) [36]. Model ten charakteryzuje się tym, że każda para wierzchołków (spośród N) łączona jest krawędzią z prawdopodobieństwem p.

Rozkład $P(k)$ stopnia k-tego wierzchołka w takim modelu jest najczęściej dwumianowy:

$$P(k) = \binom{N-1}{k} p^k (1-p)^{N-1-k} \tag{3.3}$$

lub Poissona:

$$P(k) = \frac{(pN)^k}{k!} e^{-pN} = \frac{\overline{k}^k}{k!} e^{-\overline{k}} \tag{3.4}$$

gdzie: k – średnia stopni wierzchołków w grafie. Podsumowując, graf losowy ma niski współczynnik gronowania gc_v oraz małą średnią wartość długości dróg L.

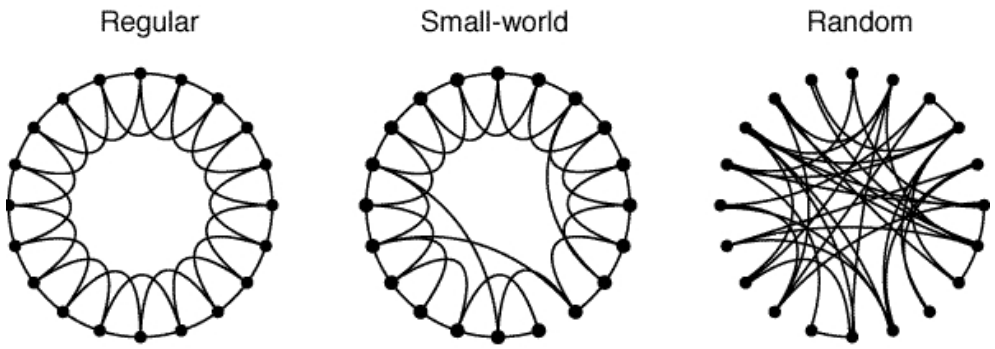
Sieć małego świata

W roku 1998 Duncan J. Watts i Steven Strogatz opublikowali pracę [93], w której zaproponowali model „małego świata" (ang. Small World, $W - S$, mały świat). Model $W - S$ powstaje z sieci regularnej, w której losowo wybrane krawędzie są „przepinane" z prawdopodobieństwem p. W związku z tym w takiej sieci pojawiają się krótsze drogi między odległymi parami wierzchołków, co przedstawione jest na rys. 3.1.

Średnia odległość (średnia długość dróg definiowana wzorem 3.12) między dwoma dowolnymi wierzchołkami jest w tych sieciach mała w porównaniu z liczbą wierzchołków i słabo zależy od rozmiaru sieci gdyż:

$$L \sim \frac{\ln N}{\ln \overline{k}} \tag{3.5}$$

Zgodnie ze spostrzeżeniami Wattsa i Strogatza, sieć może być uważana za sieć małego świata wtedy, gdy średnia odległość L w tej sieci jest porównywalna ze średnią odległością w sieci losowej oraz posiada znacznie większą wartość średniego współczynnika gronowania gc_v.

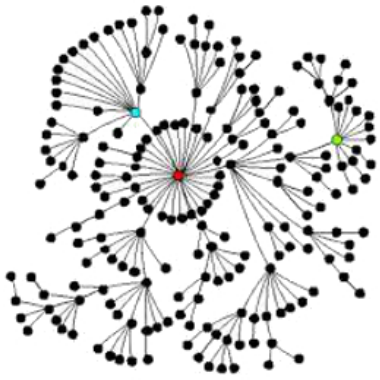


Rysunek 3.1: Przykłady sieci (od lewej): sieć regularna, sieć typu Small World, sieć losowa

Model Barabasi-Alber - sieć bezskalowa

W 1999 r. Albert-László Barabási i Réka Albert podjęli jedną z pierwszych prób zbadania struktury sieci WWW, podczas których zaobserwowali potęgowy rozkład prawdopodobieństwa opisujący strukturę sieci WWW. Swoje obserwacje opisali w pracy [7], w której zaproponowali model sieci „bezskalowej" (model $B-A$, ang. Scale Free).

Pojęcie bezskalowości zdefiniowane zostało jako niemierzalność, tzn., że obiekty w każdej skali wyglądają tak samo, a decyduje o tym prawo potęgowe opisujące ich strukturę. Przykład sieci bezskalowej został przedstawiony na rys. 3.2.



Rysunek 3.2: Przykładowa sieć bezskalowa

Model $B-A$ charakteryzuje się:

- stałym wzrostem rozmiaru sieci (ewolucja, spontaniczność)
- preferencyjnym dołączaniem nowych wierzchołków do sieci

- potęgowym rozkładem stopnia (k) wierzchołka wyrażanego wzorem:

$$P(k) \sim k^{-\gamma} \tag{3.6}$$

gdzie γ jest parametrem rozkładu, który w praktyce w sieciach rzeczywistych w zasadzie nie przekracza wartości 3.

3.3 Właściwości sieci

Do opisywania sieci społecznych stosuje się różne współczynniki oraz metody [76], które często pod swoimi nazwami kryją szereg różnych algorytmów. W celu identyfikacji wzorców zachowań w sieci opracowano wiele schematów, które w większości bazują na kilku podstawowych obserwacjach.

Głównymi wskaźnikami charakteryzującymi daną sieć społeczną są stopnie wierzchołków oraz centralność wg tych stopni. **Stopień wierzchołka** v (stopień wejściowy, stopień wyjściowy) to liczba krawędzi wchodzących lub wychodzących z danego węzła i określana jest wzorem:

$$deg(v) = \sum_{u=1}^{n} k_{v,u}, \tag{3.7}$$

gdzie k_{vu} to krawędź między wierzchołkiem v, a wierzchołkiem u.

Natomiast **centralność wg stopni wierzchołków** służy do określania, które węzły są kluczowe z punktu widzenia rozprzestrzeniania informacji lub wpływania na węzły położone w bezpośrednim sąsiedztwie. Najczęściej centralnym wierzchołkiem określa się te wierzchołki, które mają najwięcej relacji z innymi wierzchołkami (posiadają największą liczbę krawędzi). Maksymalny stopień wierzchołka v w sieci G określony jest wzorem:

$$\Delta(G) = \max\{deg(v) : v \in V(G)\}. \tag{3.8}$$

Wśród dodatkowych wskaźników charakteryzujących sieć społeczną wyróżnić można takie wskaźniki jak promień, bliskość czy pośrednictwo wierzchołka, a także średnią odległość w sieci. Miary te oblicza się wg poniższych wzorów.

Promień rc_v (ang. radius) wierzchołka v:

$$rc_v = \frac{1}{\max_{v \in V} d_{v,u}} \tag{3.9}$$

gdzie d_{vu} to długość najkrótszej drogi w grafie G między wierzchołkami v oraz u (długość drogi w grafie między wierzchołkami v oraz u równa jest liczbie krawędzi na drodze z v do u). Najwyższą ocenę uzyskuje wierzchołek, który jest możliwie najbliżej wszystkich najbardziej wysuniętych wierzchołków sieci (odległość dzieląca go od najdalszego wierzchołka jest najmniejsza).

Bliskość cc_v (ang. closeness) wierzchołka v:

$$cc_v = \frac{V-1}{\sum_{u \in V} d_{v,u}} \tag{3.10}$$

Według tej miary, wierzchołek jest tym bardziej centralny, im jest średnio bliżej wszystkich innych wierzchołków sieci. W efekcie miara ta pozwala stwierdzić, który z dowolnych dwóch wierzchołków wymaga mniej kroków, aby „skomunikować" się z dowolnym innym wierzchołkiem sieci.

Pośrednictwo bc_v (ang. betweenness, load) wierzchołka v:

$$bc_v = \frac{\sum_{w \in V} \sum_{u \neq w \in V} \frac{p_{w,v,u}}{p_{w,u}}}{(V-2)(V-1)} \tag{3.11}$$

gdzie $p_{w,v,u}$ – liczba dróg w grafie G między wierzchołkami w oraz u przechodzących przez v.

Średnia odległość L (średnia długości dróg najkrótszych) w sieci:

$$L = \frac{\sum_{v \neq u \in V} d_{v,u}}{V(n-1)}, \tag{3.12}$$

gdzie d_{vu} to długość najkrótszej drogi w grafie G między wierzchołkami v oraz u.

Sposób grupowania (ang. Clustering) interpretowany jest jako rozmieszczenie blisko siebie obiektów w jaki sposób powiązanych ze sobą, a powstała struktura określana jest jako klaster lub grono. Prawdopodobieństwo, że najbliżsi sąsiedzi wierzchołka v są również swoimi najbliższymi sąsiadami określa współczynnik gronowania gc_v wierzchołka v takim, że:

$$gc_v = \frac{2E_v}{k_v(k_v-1)}, \qquad k_v > 1, \tag{3.13}$$

gdzie E_v to liczba krawędzi k_v między sąsiadami wierzchołka v [99]. Jest to iloraz liczby krawędzi pomiędzy sąsiadami danego wierzchołka do liczby krawędzi, jaki miałby graf pełny składający się z tych sąsiadów. Współczynnik gronowania (klasteryzacji) służy do szacowania, ilu sąsiadów danego wierzchołka jest połączonych każdy z każdym.

3.4 Analiza sieci społecznych

W badaniach nad zbiorami danych zawierających wiadomości e-mail niezwykle ważną rolę odgrywa analiza sieci społecznych (ang. Social Network Analysis, SNA). Jest to przede wszystkim specyficzna perspektywa analizy, która nie skupia się na indywidualnych jednostkach lub makrostrukturach, lecz bada powiązania między poszczególnymi jednostkami czy grupami. Poprzez analizę sieci można badać sieci wielkorozmiarowe i ustalać specyfikę ich topologii i ewolucji [8].

Analiza sieci społecznych ma szeroki zakres zastosowań. Przede wszystkim stosowana jest w dużych organizacjach i firmach jako narzędzie wspierające strategiczne zarządzanie zasobami ludzkimi czy też zarządzanie wiedzą w organizacji. SNA wspiera innowacyjność firmy, a także służy analizie procesów biznesowych oraz analizie potrzeb szkoleń. Dodatkowo wykorzystywana jest przy badaniach marketingowych w tworzeniu mapy sieci społecznej klientów. Analiza sieci społecznych pozwala jednak przede wszystkim kadrze zarządzającej na zapoznanie się z nieformalną strukturą organizacji i przepływu informacji w firmie.

Wiele badań nad sieciami dotyczyło znajdowania korelacji między społeczną strukturą sieci a jej wydajnością [44]. Początkowo analiza sieci społecznych przeprowadzana była na podstawie ankiet wypełnianych ręcznie przez uczestników [28], jednak z czasem popularne stały się badania przeprowadzane z zastosowaniem wiadomości e-mail [6]. W niektórych badaniach stwierdzono, że zespoły badawcze są bardziej kreatywne, gdy posiadają większy kapitał społeczny [43]. Sieci społeczne związane są również z odkrywaniem sieci komunikacji. Takie podejście rozważali m.in. Garnett C. Wilson and Wolfgang Banzhaf, co opisali w swojej pracy [94].

Przyjmuje się, iż za pomocą SNA można poddać analizie sieć na trzech podstawowych poziomach:

- na poziomie całej sieci (np. ilość relacji, tzw. gęstość sieci, dystans pomiędzy elementami sieci, wyodrębnienie rdzenia i peryferia sieci etc.);
- na poziomie poszczególnych części sieci (np. określenie relacji pomiędzy różnymi grupami, wskazywanie centralnych oraz wyizolowanych grup w sieci, wskazanie wąskich gardeł pomiędzy grupami etc.);
- na poziomie poszczególnych węzłów sieci (np. wskazanie elementów będących integratorami sieci, wskazanie głównych pośredników, wskazanie podmiotów peryferyjnych etc.).

4

Przygotowanie danych

Pierwszym krokiem w realizacji głównego celu rozprawy jest przeanalizowanie zbioru danych Enron E-mail, oczyszczenie go, dostosowanie do problemu oraz przekształcenie do odpowiedniej struktury, co zostało opisane w niniejszym rozdziale. Wybór odpowiedniego zbioru danych nie jest tu przypadkowy, gdyż właśnie zbiór danych E-mail Enron jest powszechnie stosowany do badań związanych z eksploracją danych, przetwarzaniem języka naturalnego oraz uczenia maszynowego. Ponadto zbiór ten jest uważany za jeden z cenniejszych zbiorów, gdyż zawiera rzeczywiste wiadomości e-mail dostępne publicznie, co często jest problematyczne z uwagi na prywatność danych z innych zestawów.

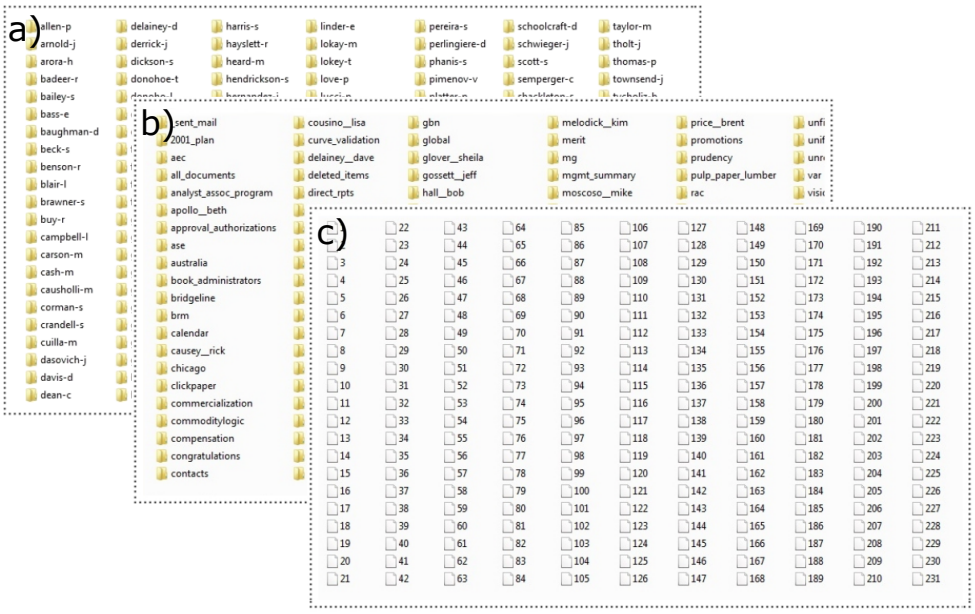
4.1 Analiza zbioru Enron E-mail

Enron E-mail Dataset [1] jest to zestaw danych zebrany i przygotowany przez Projekt CALO (A Cognitive Assistant that Learns and Organizes). Zawiera ponad 600 tys. wiadomości e-mail, które zostały wysłane lub odebrane przez 150 pracowników wyższego szczebla z Enron Corporation. Zbiór danych został przejęty przez Komisję Regulacji Energetyki Federalnej w trakcie dochodzenia po upadku firmy, a następnie został podany do publicznej wiadomości. Kopia bazy danych została wykupiona przez Leslie Kaelbling z Massachusetts Institute of Technology (MIT), po czym okazało się, że w zbiorze są duże problemy związane z integralnością danych. Dzięki pracy zespołu z ośrodka SRI International, zwłaszcza Melinda Gervasio, dane zostały poprawione i udostępnione innym naukowcom do badań.

Wiadomości te są przypisane do kont osobistych i podzielone na foldery. W zbiorze danych nie ma załączników do wiadomości e-mail, a niektóre wiadomości zostały usunięte ze względu na występowanie duplikatów w innych folderach. Brakujące informacje zostały w miarę możliwości uzupełnione na podstawie in-

nych treści, jednak w przypadku, gdy nie było możliwości określenia odbiorcy wprowadzono frazę no_address@enron.com.

Każda skrzynka pocztowa pracowników firmy Enron Corporation jest przechowywana w osobnym folderze i oznaczona nazwiskiem danego pracownika. W każdej skrzynce są foldery utworzone automatycznie przez system pocztowy (np. *sent_mail*, *all_documents*, *deleted_items*) oraz foldery utworzone przez użytkowników. Wewnątrz tych folderów są kolejno ponumerowane wiadomości e-mail. Przykładowy fragment zbioru Enron został przedstawiony na rys. 4.1, gdzie w części *a*) są przedstawione skrzynki pocztowe pracowników firmy Enron Corporation, w części *b*) widać foldery dla jednej przykładowej skrzynki pocztowej, natomiast w części *c*) są wiadomości e-mail znajdujące się w wybranym folderze.



Rysunek 4.1: Fragnemt zbioru Enron E-mail

Wszystkie wiadomości w zbiorze Enron E-mail Dataset mają jednakową budowę. Są to pliki tekstowe zawierające w kolejnych liniach szczegółowe informacje tj.: identyfikator wiadomości, data wysłania, adres pocztowy nadawcy, adres pocztowy odbiorcy, temat wiadomości, odbiorcy, do których wysłano kopię wiadomości, imię i nazwisko nadawcy wiadomości, imię i nazwisko odbiorcy wiadomości, nazwa folderu, w którym jest wiadomość, nazwa skrzynki pocztowej, w której jest wiadomość, treść wiadomości. Przykład takiej wiadomości jest przedstawiony na rys. 4.2.

W tab. 4.1 przedstawione zostały parametry dotyczące każdej skrzynki pocztowej ze zbioru Enron. Zawarte są tam dane dotyczące liczby folderów oraz liczby wiadomości e-mail zawartych w skrzynkach pocztowych, jak również dane statystyczne, dotyczące występowania wiadomości w folderach.

```
Message-ID: <29670339.1075849829690.JavaMail.evans@thyme>
Date: Thu, 15 Feb 2001 07:27:00 -0800 (PST)
From: lisa.petruszka@enron.com
To: sally.beck@enron.com
Subject: ASE information for Harvard Business School
Cc: charles.saltsman@us.cgeyc.com, robert.evans@us.cgeyc.com
Mime-Version: 1.0
Content-Type: text/plain; charset=us-ascii
Content-Transfer-Encoding: 7bit
Bcc: charles.saltsman@us.cgeyc.com, robert.evans@us.cgeyc.com
X-From: Lisa Petruszka
X-To: Sally Beck
X-cc: "charles.saltsman@us.cgeyc.com" <robert.evans@us.cgeyc.com,>@SMTP@enronXgate, robert.evans@us.cgeyc.com@SMTP@enronXgate
X-bcc:
X-Folder: \Sally_Beck_Nov2001\Notes Folders\Ase
X-Origin: BECK-S
X-FileName: sbeck.nsf

Sally -

To follow up on our conversation Monday about the Harvard Business School
study and their interest in CGEY's Accelerated Solutions Environment (ASE), I
am including contact information for two key people with the ASE.  Feel free
to pass this information along to the professors at Harvard.

Rob Evans, Global Director for the ASE and Facilitator
Voicemail:  617-859-6757
Cellphone:  617-513-0663

Chip Saltsman, Facilitator
Phone:  410-783-3725
Cellphone:  443-655-5820

Rob Evans is actually based out of our facility in Cambridge, MA.
```

Rysunek 4.2: Przykładowa wiadomość ze zbioru Enron E-mail

Tabela 4.1: Parametry zbioru danych Enron

Skrzynka pocztowa	Liczba wiadomości	Liczba folderów	Liczba wiadomości w folderze		
			średnia	minimalna	maksymalna
allen-p	3034	10	303,40	2	628
arnold-j	4898	46	106,48	1	1047
arora-h	654	14	46,71	1	197
badeer-r	877	16	54,81	1	299
bailey-s	478	5	95,60	4	434
bass-e	7823	15	521,53	1	2037
baughman-d	2759	59	46,76	1	3894
beck-s	11830	137	86,35	1	3137
benson-r	767	10	76,70	6	274
blair-l	3415	69	49,49	1	1120
brawner-s	1026	13	78,92	1	240
buy-r	2429	16	151,81	1	1143
campbell-l	6490	54	120,19	1	1708
carson-m	1400	12	116,67	2	373
cash-m	2969	31	95,77	1	726
causholli-m	943	4	235,75	10	515
corman-s	2024	23	88,00	1	651
crandell-s	519	12	43,25	1	246

Kontynuacja na następnej stronie

Tabela 4.1 – kontynuacja z poprzedniej strony

Skrzynka pocztowa	Liczba wiadomości	Liczba folderów	Liczba wiadomości w folderze		
			średnia	minimalna	maksymalna
cuilla-m	1029	13	79,15	5	421
dasovich-j	28234	64	441,16	1	11896
davis-d	2249	33	68,15	3	478
dean-c	2429	11	220,82	3	971
delainey-d	3566	8	445,75	9	910
derrick-j	1766	12	147,17	1	583
dickson-s	395	3	131,67	7	199
donohoe-t	1015	8	126,88	10	430
donoho-l	1045	6	174,17	1	511
dorland-c	2127	11	193,36	1	587
ermis-f	1230	8	153,75	6	379
farmer-d	13032	40	325,80	1	3660
fischer-m	1498	12	124,83	2	429
forney-j	729	5	145,80	1	432
fossum-d	4796	7	685,14	1	1405
gang-l	590	4	147,50	1	451
gay-r	1415	9	157,22	9	420
geaccone-t	1592	18	88,44	1	569
germany-c	12436	71	175,15	1	3027
gilbertsmith-d	578	23	25,13	1	191
giron-d	4220	11	383,64	1	853
griffith-j	2973	77	38,61	1	924
grigsby-m	2237	9	248,56	3	450
guzman-m	6054	4	1513,50	352	2067
haedicke-m	5246	12	437,17	1	2141
hain-m	3820	5	764,00	5	1347
harris-s	548	2	274,00	56	492
hayslett-r	2554	49	52,12	1	533
heard-m	1623	7	231,86	2	784
hendrickson-s	719	9	79,89	1	180
hernandez-j	3265	16	204,06	1	851
hodge-j	1661	10	166,10	1	639
holst-k	463	3	154,33	36	235
horton-s	2470	9	274,44	22	713
hyatt-k	1794	28	64,07	3	647
hyvl-d	3210	45	71,33	1	1627
jones-t	19950	10	1995,00	2	9304

Kontynuacja na następnej stronie

Tabela 4.1 – kontynuacja z poprzedniej strony

Skrzynka pocztowa	Liczba wiadomości	Liczba folderów	Liczba wiadomości w folderze		
			średnia	minimalna	maksymalna
kaminski-v	28465	61	466,64	1	7174
kean-s	25351	198	128,04	1	4478
keavey-p	2177	10	217,70	2	518
keiser-k	1113	5	222,60	27	393
king-j	462	13	35,54	1	346
kitchen-l	5546	60	92,43	1	1132
kuykendall-t	1120	9	124,44	15	220
lavorato-j	4685	33	141,97	1	853
lay-k	5937	15	395,80	1	1373
lenhart-m	5920	7	845,71	44	2056
lewis-a	2191	10	219,10	7	1359
linder-e	2805	4	701,25	6	941
lokay-m	5567	21	265,10	1	1324
lokey-t	1156	7	165,14	16	742
love-p	5002	21	238,19	1	1035
lucci-p	997	5	199,40	13	569
maggi-m	1991	10	199,10	1	1649
mann-k	23381	34	687,68	2	6647
martin-t	1112	10	111,20	1	465
may-l	1600	9	177,78	1	1087
mccarty-d	691	23	30,04	1	254
mcconnell-m	4542	82	55,39	1	1150
mckay-b	681	8	85,13	24	308
mckay-j	998	19	52,53	2	296
mclaughlin-e	3353	21	159,67	1	720
merriss-s	1627	4	406,75	3	548
meyers-a	1099	3	366,33	11	1066
mims-thurston-p	2038	8	254,75	10	715
motley-m	378	13	29,08	3	125
neal-s	3268	21	155,62	1	657
nemec-g	10655	12	887,92	1	4231
panus-s	437	5	87,40	2	377
parks-j	2284	4	571,00	3	1078
pereira-s	725	9	80,56	1	164
perlingiere-d	4778	11	434,36	1	1905
phanis-s	35	4	8,75	2	17
pimenov-v	642	8	80,25	1	359

Kontynuacja na następnej stronie

Tabela 4.1 – kontynuacja z poprzedniej strony

Skrzynka pocztowa	Liczba wiadomości	Liczba folderów	Liczba wiadomości w folderze		
			średnia	minimalna	maksymalna
platter-p	574	6	95,67	6	393
presto-k	2178	16	136,13	1	1040
quenet-j	395	8	49,38	6	136
quigley-d	1568	17	92,24	1	506
rapp-b	563	7	80,43	1	332
reitmeyer-j	498	4	124,50	1	352
richey-c	581	21	27,67	1	253
ring-a	706	8	88,25	8	137
ring-r	994	14	71,00	1	296
rodrique-r	2766	8	345,75	1	820
rogers-b	8009	23	348,22	1	2112
ruscitti-k	1643	35	46,94	1	351
sager-e	5200	14	371,43	3	1916
saibi-e	1116	8	139,50	2	1008
salisbury-h	1632	5	326,40	9	1152
sanchez-m	256	8	32,00	2	102
sanders-r	7329	51	143,71	1	3035
scholtes-d	646	27	23,93	1	4096
schoolcraft-d	1859	19	97,84	1	608
schwieger-j	738	11	67,09	2	200
scott-s	8022	21	382,00	2	1893
semperger-c	721	13	55,46	1	362
shackleton-s	18687	12	1557,25	1	8158
shankman-j	3856	13	296,62	2	1084
shapiro-r	6071	16	379,44	1	1468
shively-h	1989	14	142,07	2	6618
skilling-j	4139	12	344,92	2	1252
slinger-r	132	3	44,00	35	49
smith-m	1642	13	126,31	1	452
solberg-g	1081	6	180,17	1	851
south-s	248	7	35,43	2	84
staab-t	621	18	34,50	2	284
stclair-c	3030	14	216,43	2	1523
steffes-j	3331	64	52,05	1	1379
stepenovitch-j	1227	7	175,29	76	319
stokley-c	1252	24	52,17	2	515
storey-g	1027	19	54,05	1	327

Kontynuacja na następnej stronie

Tabela 4.1 – kontynuacja z poprzedniej strony

Skrzynka pocztowa	Liczba wiadomości	Liczba folderów	Liczba wiadomości w folderze		
			średnia	minimalna	maksymalna
sturm-f	1169	21	55,67	1	211
swerzbin-m	355	5	71,00	1	265
symes-k	10827	27	401,00	1	3221
taylor-m	13875	88	157,67	1	5229
tholt-j	1885	9	209,44	28	373
thomas-p	1293	6	215,50	1	687
townsend-j	646	9	71,78	2	243
tycholiz-b	1219	9	135,44	2	529
ward-k	2611	49	53,29	1	677
watson-k	2950	69	42,75	1	971
weldon-c	1566	54	29,00	1	304
whalley-g	1878	24	78,25	1	462
whalley-l	3335	9	370,56	3	1083
white-s	3272	34	96,24	1	989
whitt-m	807	3	269,00	37	467
williams-j	1213	12	101,08	3	433
williams-w3	3440	24	143,33	1	1398
wolfe-j	1587	19	83,53	1	348
ybarbo-p	1291	20	64,55	3	612
zipper-a	1563	26	60,12	1	542
zufferli-j	557	6	92,83	2	338

4.2 Przypisanie wiadomości do folderów

Głównym narzędziem do zarządzania wiadomościami e-mail jest klasyfikacja dokumentów (opisana w rozdziale 1.4), gdzie szczególnym przypadkiem jest tzw. „E-mail Foldering Problem", czyli proces przypisania wiadomości e-mail do folderów. Polega on na tym, że użytkownicy tworzą nowe katalogi, a także przestają korzystać z niektórych folderów utworzonych wcześniej. Jednocześnie foldery nie zawsze odpowiadają tematom otrzymywanych wiadomości, czasami mogą dotyczyć zadań do wykonania, grup projektowych, niektórych odbiorców, a inne mają sens tylko w powiązaniu z poprzednimi wiadomościami. E-mail Foldering Problem jest problemem złożonym, gdyż automatyczna metoda klasyfikacji może się sprawdzić u jednego użytkownika, a u innego może prowadzić do błędów.

Dodatkowo wiadomość e-mail ma bardzo skomplikowany format wielowymiarowy, gdyż wiadomości mogą być przesyłane, przekazywane, kopiowane, a także

może na nie odpowiadać wiele osób lub grup w różnym czasie. Przesyłane wiadomości mogą zawierać jako załączniki inne wiadomości e-mail lub dokumenty w postaci dołączonych plików. Ponadto informacje uzyskane z tematu mogą mieć inne znaczenie niż informacje uzyskane z treści lub załączników przesłanej wiadomości. Co gorsza, informacje docierają do użytkowników w różnym czasie, co powoduje dodatkowe trudności w zarządzaniu pocztą elektroniczną.

Rozwiązania problemu folderingu mogą mieć zastosowanie w wielu przypadkach, w szczególności do filtrowania wiadomości na podstawie priorytetu przypisywania wiadomości e-mail do folderów utworzonych przez użytkownika, a także do identyfikacji spamu. Klasyfikacja dokumentów ma szeroki zakres zastosowań, do których przede wszystkim należą filtrowanie spamu polegające na zaklasyfikowaniu dokumentu wiadomości e-mail jako spamu lub jako wiadomości użytecznej dla użytkownika. W tym przypadku mamy zazwyczaj do czynienia z klasyfikacją dokumentu do jednej z dwu kategorii: odpowiadającej tematowi – nieodpowiedniej, użytecznej – spam, relewantnej – nierelewantnej. Kolejnym z typowych zastosowań klasyfikacji dokumentów są katalogi tematyczne polegające na porządkowaniu informacji pod kątem tematycznym, bądź też poprzez znajdowanie wątków dyskusji. Ponadto klasyfikację stosuje się do określania ważności wiadomości e-mail poprzez oznaczenie odpowiedniego priorytetu, a także do ekstrakcji informacji, gdzie wyszukiwane są konkretne informacje w tekście, np. o atakach terrorystycznych, w wyniku czego powstaje zwięzła struktura danych, a nie zbiór dokumentów, jak w przypadku wyszukiwania informacji w Internecie.

4.3 Przegląd prac

Pierwsze badania nad metodami kategoryzacji wiadomości powstały w latach dziewięćdziesiątych. D. Lewis wprowadził model Concept Learning dla systemów klasyfikacji tekstu, w tym systemów do pobierania dokumentów, automatycznego indeksowania czy filtrowania poczty elektronicznej [55]. Większość prac dotycząca klasyfikowania wiadomości e-mail skupia się na filtrowaniu spamu wykorzystując naiwny klasyfikator bayesowski (ang. Naive Bayes) lub maszynę wektorów nośnych (ang. Support Vector Machine, SVM) opisane w rozdziale 1.4.

P. Clark i T. Niblett w 1989 r. przedstawili algorytm indukcji reguł CN2 [23], który razem z algorytmem k-najbliższego sąsiada został zastosowany przez T.R. Payne and P. Edwards w pracy [70] do stworzenia interfejsu, który wykorzystuje element uczenia się do filtrowania wiadomości e-mail. W kolejnych pracach Kiritchenko i Matwin w pracy [51] przeprowadzili badania, z których wynikło, że klasyfikacja za pomocą SVM daje dużo lepsze wyniki niż naiwny klasyfikator bayesowski.

Seongwook Youn i Dennis McLeod [97] zaproponowali adaptacyjne podejście ontologiczne do filtrowania poczty e-mail. Filtr ontologii stale się rozwijał w opar-

ciu o preferencje użytkownika, dzięki czemu był bardziej adaptowalny. Sahami i inni [80] zaproponowali wykorzystanie specyficznych funkcji domenowych, takich jak typy nadawców z klasyfikatorem Bayesa do filtrowania wiadomości – śmieci. Zwroty takie jak „Free Money" zostały uznane za cechy specyficzne dla tych domen. Rozważając te dodatkowe funkcje oraz naturalną treść wiadomości e-mail, poprawiono ich poprawność.

Wang i inni[91] zaproponowali metodę połowową nadzorowaną, obejmującą przetwarzanie informacji zwrotnych od użytkownika. Metoda ta stosuje różne reguły klasyfikacji do każdej sekcji poczty elektronicznej. Na podstawie wyników klasyfikacji e-maile są przypisywane do właściwej grupy.

Saxena i inni [81] zaproponowali podejście oparte na klasyfikacji adresów e-mail w oparciu o Ant Clustering Algorithm. Wykorzystują podejście etapowe z pierwszą fazą szkolenia, obejmującą ręczne sortowanie wiadomości e-mail do folderów przez użytkowników. Następnie jest etap testowania, w którym testowane są e-maile z już określonymi kategoriami i ostatnia faza przetwarzania dokumentów, w której algorytm stosuje się do e-maili, których kategorie muszą zostać określone.

Celem Arey i innych [3] jest zautomatyzowanie procesu klasyfikacji poczty elektronicznej. Podejście oparte jest na hipotezie, że struktura i wzorce mogą być wyodrębniane i używane w klasyfikacji przychodzących e-maili. Wykorzystują graficzną reprezentację struktury wiadomości e-mail (nagłówek, treść) i relacje między różnymi terminami występującymi w strukturze.

Vira i inni [89] zaproponowali algorytm, który używa teorii Bayesowskiej do klasyfikowania e-maili. Prawdopodobieństwo warunkowe jest wykorzystywane w treści tekstowej e-maila przy użyciu słów kluczowych z ręcznie sklasyfikowanych e-maili użytkownika.

Cui i inni [27] zaproponowali metodę klasyfikacji adresów e-mail opartą na sieciach neuronowych. Ta metoda służyła do klasyfikacji osobistych e-maili, które uznano za zwykły tekst i wykorzystywano Personal Component Analysis (PCA) jako preprocesor do sieci neuronowych, co skutkowało redukcją danych ułatwiających proces klasyfikacji.

Klasyfikatory TF-IDF są najczęściej używanymi i popularnymi klasyfikatorami używanymi do klasyfikacji e-maili. Zaproponowano różne typy klasyfikatorów. Segal i Kephart zaproponowali identyfikator TF-IDF w [83], który sugerował, że w górnych kategoriach również nie sklasyfikowano e-maila. Zostało to określone na podstawie zasady TF-IDF do obliczania ciężaru wektora częstotliwości słowa. Kolejny klasyfikator zaproponował Cohen [24], w którym e-mail jest reprezentowany jako ważony wektor. Obliczono wagę TF-IDF. Ustawiono próg, a nowo przychodzący e-mail został sklasyfikowany w określonej kategorii, jeśli wynik podobieństwa (wynikowy produkt adresu e-mail i kategorii) był mniejszy niż próg.

IFile to narzędzie filtrujące adresy e-mail opracowane przez Jasona Rennie'a

[75]. Opiera się on na algorytmie klasyfikacji naiwnego Bayesa i składa się z 3 warstw. Czynność wykonywalna, której zadaniem jest przechowywanie i utrzymywanie modelu klasyfikacji oraz generowanie etykiet klasy e-maili. Warstwy skryptów opakowania filtrują nadchodzące wiadomości e-mail i aktualizują model klasyfikacji. Ostatni kod Tcl jest używany do wyszukiwania interfejsu użytkownika dla IFile przy minimalnej interferencji użytkownika.

Klimt Bryan i Yang Yiming przedstawili zbiór Enron jako nowy obszar badań stosowany do ekstrakcji informacji oraz automatycznej klasyfikacji wiadomości do folderów. Przeanalizowali przydatność zbioru w odniesieniu do przewidywania folderu przy zastosowaniu SVM dla poszczególnych fragmentów maila [52].

Z kolei R. Bekkerman, A. McCallum, G. Huang [10] przedstawili studium przypadku benchmarku e-mail Foldering na przykładzie dwóch zbiorów danych e-mail: Enron i SRI. Wykonali klasyfikację wiadomości z 7 skrzynek pocztowych do folderów tematycznych na postawie 4 klasyfikatorów: Maximum Entropy (MaxEnt), Naive Bayes, Support Vector Machine (SVM) and Wide-margin Winnow. Przed uruchomieniem klasyfikatorów szkoleniowych dane zostały wyczyszczone i ustandaryzowane. Usunięto foldery zawierające małą liczbę wiadomości, oraz foldery uznane za nieaktualne, czyli takie, które zostały automatycznie utworzone z aplikacji poczty elektronicznej (np. skrzynka odbiorcza, elementy wysłane, kosz) oraz te, które zostały zarchiwizowane dla wszystkich użytkowników pewnej organizacji, a które można znaleźć w hierarchii folderów wszystkich byłych pracowników firmy Enron. Jednak pozostawiono foldery, które zostały zarchiwizowane indywidualnie przez poszczególnych pracowników.

W pracy [20] przeanalizowano dane Enron E-mail, aby dowiedzieć się jak są określone struktury w organizacji. Chapanond Anurat, Krishnamoorthy Mukkai S., Yener Bülent przeprowadzili analizę, która opierała się na konstruowaniu wykresu e-mail i badała jego właściwości zarówno teoretyczne, jak i wykres technik analizy widmowej. Analiza teoretyczna wykresu zawierała wyliczenie szeregu wskaźników, takich jak wykres rozkładu stopni, średni wskaźnik odległości czy współczynnik grupowania na wykresie. Wykazano, że wstępne przetwarzanie danych ma znaczący wpływ na wyniki, więc standardowy formularz jest potrzebny do ustalenia poziomu odniesienia danych.

Ke Shih-Wen, Bowerman Chris, Oakes Michael przedstawili hybrydowy model klasyfikacji PERC i zbadali wydajność KNN, SVM i PERC w symulacji sytuacji w czasie rzeczywistym na danych Enron E-mail. Wyniki opisane w pracy [49] pokazują, że PERC jest znacznie lepszy w przypisaniu wiadomości do małych folderów. Zaobserwowano poprawę dokładności poprzez zmniejszenie prawdopodobieństwa tworzenia przez użytkowników zduplikowanych folderów dla niektórych tematów.

Ampazis Nikolaos, Iakovaki Helen, Dounias Georgios w swojej pracy [5] zaproponowali metodę, którą porównali z wydajnością osiąganą przez naiwny klasyfikator Bayesa i SVM. Autorska identyfikacja z OLMAM okazała się znacznie

lepsza w porównaniu z innymi metodami.

M. Wang, Y. He, M. Jiang w swojej pracy [90] przedstawili kategoryzacje wiadomości e-mail opartej na wąskim gardle informacyjnym (information bottleneck, IB) oraz maksymalnej entropii. Metodę wąskiego gardła zastosowano do znajdowania i grupowania słów kluczowych w oparciu o dystrybucję e-maili do różnych folderów, a następnie tematy wiadomości e-mail i grupy adresowe posłużyły jako dodatkowe cechy do klasyfikacji na podstawie treści wiadomości. Natomiast model maksymalnej entropii służył do poprawienia dokładności klasyfikatora.

T.A. Almeida i A. Yamakami omówili siedem różnych wersji klasyfikatora naiwnego Bayesa i porównali go z SVM. Zaproponowali nowy pomiar w celu oceny jakości klasyfikatorów antyspamowych. W ten sposób zbadali korzyści wynikające z zastosowania współczynnika korelacji Matthews jako miary wydajności [4].

C. Priyanka, W. Rajesh i S. Sanyam zastosowali algorytm SVM do opracowania filtrów antyspamowych. Przedstawiony został klasyfikator SVM oparty na różnych funkcjach, które oceniono pod względem wydajności [21]. Natomiast Ke Xu, Cui Wen, Qiong Yuan, Xiangzhu He i Jun Tie w swojej pracy [96] przedstawili równoległy klasyfikator SVM w oparciu o algorytm MapReduce (PSMR) stosowany do klasyfikacji wiadomości e-mail, gdzie omówili także problemy, które wynikają z różnic między foldering e-mail i tradycyjnej klasyfikacji dokumentów.

S. Sayed, S. Abdelrahman i I. Farag zaproponowali metodę klasyfikacji, która przechodzi przez trzy etapy [79]. Pierwszy etap to klastrowanie folderów poczty e-mail przy użyciu algorytmu K-means z ustalonej liczby klas klastrowych. W drugim etapie zastosowano dwa klasyfikatory: Maximum Entropy i Wide-margin Winnow do szkolenia binarnego klasyfikatorów na każdej parze klas w celu zmniejszenia błędów klasyfikacji. Trzecia faza to metoda turnieju 2-warstwowego (2-layer tournament method) polegająca na eliminacji, gdzie kolejno wyłaniany jest zwycięzca na klasę klastra i zwycięzca wszystkich klastrów odpowiednio.

Badania przeprowadzone przez Trivedi Shrawan Kumar, Dey Shubhamoy, Shikhar Prabandh w pracy [86] przedstawiają skutki korzystania z wybranych funkcji dwóch metod selekcji „Genetic Search" oraz „Greedy Stepwise Search" stosowanych na popularnych klasyfikatorach uczenia maszynowego tj. naiwny klasyfikator Bayesa, SVM, algorytm genetyczny. Wyniki pokazują, że „Greedy Stepwise Search" jest dobrą metodą selekcji cech do wykrywania spamu. Wśród innych badanych klasyfikatorów SVM został uznany za najlepszy zarówno pod względem dokładności klasyfikacji jak i liczby wyników fałszywie dodatnich.

W kolejnym roku w pracy [87] przedstawiono skutki oddziaływania pomiędzy różnymi funkcjami jądra i różnych technik selekcji cech dla poprawy zdolności uczenia się maszyny wektorów nośnych (SVM) w wykrywaniu spamu. Interakcja z czterech funkcji jądra SVM czyli „Znormalizowany Wielomian Kernel (NP)", „Wielomian Kernel (PK)", „Radial Basis Function Kernel (RBF)" i „Pearson VII Uniwersalny Kernel (PUK)" oparta na jednej z trzech funkcji techniki selek-

cji czyli „Gain Ratio (GR)", „Chi-kwadrat" , „ukryte indeksowanie semantyczne (LSI)" była testowana na zbiorze Enron E-mail. Wyniki pokazują kilka interesujących faktów dotyczących zmienności wykonywania funkcji jądra z wielu funkcji (lub wymiarów) do danych. NP najlepiej przeprowadza się w szerokim zakresie wymiarowości, na wszystkich badanych technikach selekcji cech. PUK dobrze radzi sobie na danych o niskiej wymiarowości i jest na drugim miejscu pod względem wydajności (po NP), ale wykazuje słabe działanie na danych o wysokiej wymiarowości. LSI wydaje się być najlepszą wśród wszystkich badanych technik selekcji funkcji. Jednak dla dużych danych przestrzennych, wszystkie techniki selekcji cech wykonują się niemal równie dobrze.

4.4 Oczyszczenie danych

Podczas przeprowadzonej analizy zbioru Enron oraz pamiętając o celu pracy, postanowiono w pierwszej kolejności odrzucić foldery, które zostały utworzone w sposób automatyczny przez program pocztowy. Przede wszystkim są to foldery: *Inbox*, *Sent*, *Sent Items*, *Trash*, *Alldocuments*, *Draft*, *Calendar*, *Contacts*, *Discussion Threads* oraz *Deleted Items*. Foldery te występowały we wszystkich skrzynkach pocztowych, a użytkownicy nie mieli wpływu na tworzenie tych folderów.

Po usunięciu tych folderów okazało się, że w 32 skrzynkach nie zostały żadne inne foldery, a w 22 skrzynkach został tylko jeden. W związku z czym ze zbioru danych usunięto wszystkie puste skrzynki oraz te zawierające tylko jeden folder.

W kolejnym kroku spłaszczono drzewiastą strukturę folderów, pozostawiając tylko pierwszy poziom folderów w każdej skrzynce oraz usuwając foldery zagnieżdżone. Wiadomości e-mail, które pierwotnie były w folderach zagnieżdżonych zostały przeniesione do folderów z pierwszego poziomu struktury drzewa.

Ostatnią modyfikacją zbioru danych Enron E-mail było usunięcie folderów zawierających tylko jedną wiadomość. Parametry otrzymanego zbioru danych Enron po oczyszczeniu zostały przedstawione w tab. 4.2.

Tabela 4.2: Parametry zbioru danych Enron po oczyszczeniu

Skrzynka pocztowa	Liczba wiadomości	Liczba folderów	Liczba wiadomości w folderze		
			średnia	minimalna	maksymalna
arnold-j	121	17	7,12	3	16
arora-h	100	2	50,00	10	90
badeer-r	107	6	17,83	6	70
bass-e	33	3	11,00	5	20
baughman-d	1027	38	27,03	3	201
beck-s	1971	101	19,51	3	166

Kontynuacja na następnej stronie

Tabela 4.2 – kontynuacja z poprzedniej strony

Skrzynka pocztowa	Liczba wiadomości	Liczba folderów	Liczba wiadomości w folderze		
			średnia	minimalna	maksymalna
blair-l	1422	39	36,46	3	1120
brawner-s	14	2	7,00	4	10
buy-r	10	2	5,00	3	7
campbell-l	422	29	14,55	3	77
carson-m	109	3	36,33	22	46
cash-m	191	16	11,94	3	26
corman-s	21	3	7,00	5	9
cuilla-m	85	5	17,00	5	46
dasovich-j	794	39	20,36	3	140
davis-d	317	14	22,64	3	64
dean-c	267	3	89,00	3	233
farmer-d	3672	25	146,88	5	1192
fischer-m	55	3	18,33	11	32
geaccone-t	91	8	11,38	4	33
germany-c	1013	19	53,32	3	390
gilbertsmith-d	53	6	8,83	4	22
griffith-j	807	44	18,34	3	223
haedicke-m	112	16	7,00	3	34
hayslett-r	431	29	14,86	3	54
hernandez-j	142	6	23,67	5	53
hyatt-k	744	21	35,43	3	166
hyvl-d	891	37	24,08	3	77
jones-t	403	3	134,33	18	365
kaminski-v	4477	41	109,20	3	547
kean-s	9905	123	80,53	3	4477
king-j	11	2	5,50	4	7
kitchen-l	4015	46	87,28	5	715
lavorato-j	197	14	14,07	3	72
lay-k	51	4	12,75	3	35
lokay-m	2493	11	226,64	6	1159
lokey-t	38	2	19,00	16	22
love-p	309	8	38,63	3	123
lucci-p	59	2	29,50	13	46
mann-k	1616	24	67,33	3	227
mcconnell-m	546	38	14,37	3	133
mckay-j	430	14	30,71	3	103
mclaughlin-e	576	9	64,00	5	195

Kontynuacja na następnej stronie

Tabela 4.2 – kontynuacja z poprzedniej strony

Skrzynka pocztowa	Liczba wiadomości	Liczba folderów	Liczba wiadomości w folderze		
			średnia	minimalna	maksymalna
neal-s	81	6	13,50	3	52
nemec-g	35	2	17,50	16	19
platter-p	12	2	6,00	6	6
presto-k	43	6	7,17	3	14
quigley-d	563	11	51,18	4	146
rapp-b	6	2	3,00	3	3
richey-c	105	8	13,13	3	39
ring-r	568	8	71,00	4	296
rodrique-r	97	3	32,33	17	47
rogers-b	1395	14	99,64	3	489
ruscitti-k	139	9	15,44	3	48
sager-e	137	6	22,83	3	61
salisbury-h	327	2	163,50	9	318
sanders-r	1188	30	39,60	4	420
scholtes-d	504	20	25,20	3	132
schoolcraft-d	70	3	23,33	4	43
schwieger-j	164	4	41,00	3	110
scott-s	641	10	64,10	3	250
semperger-c	64	6	10,67	3	29
shackleton-s	1001	53	18,89	3	259
shankman-j	133	4	33,25	9	64
shapiro-r	1970	75	26,27	3	181
shively-h	36	4	9,00	4	20
smith-m	24	3	8,00	3	15
staab-t	149	13	11,46	3	28
stclair-c	124	10	12,40	3	46
steffes-j	625	23	27,17	3	317
stokley-c	1250	19	65,79	4	515
storey-g	273	9	30,33	9	86
symes-k	770	12	64,17	3	254
taylor-m	656	21	31,24	3	210
ward-k	457	29	15,76	3	73
watson-k	1005	30	33,50	3	614
weldon-c	268	21	12,76	3	64
whalley-g	328	9	36,44	3	172
whalley-l	182	3	60,67	3	172
white-s	933	17	54,88	3	738

Kontynuacja na następnej stronie

Tabela 4.2 – kontynuacja z poprzedniej strony

Skrzynka pocztowa	Liczba wiadomości	Liczba folderów	Liczba wiadomości w folderze		
			średnia	minimalna	maksymalna
williams-j	163	6	27,17	3	50
williams-w3	2769	18	153,83	3	1398
wolfe-j	638	12	53,17	3	113
ybarbo-p	244	12	20,33	3	57
zipper-a	288	11	26,18	3	61

4.5 Zastosowanie tabel decyzyjnych

Realizując jeden z celów pobocznych, a jednocześnie pierwszy etap pracy związanej z rozprawą doktorską dokonano analizy zbioru skrzynek pocztowych pracowników Enron Corporation oraz opracowano metodę umożliwiającą przekształcenie wszystkich wiadomości e-mail ze skrzynek pocztowych na tabele decyzyjne. Każdy wiersz w takiej tabeli decyzyjnej zawiera regułę, która określa decyzje, jakie muszą zostać podjęte, gdy odpowiednie warunki zostaną spełnione. Zasadność tego pomysłu została potwierdzona poprzez przeprowadzone doświadczenia opisane w dalszych rozdziałach rozprawy.

Formalnie tabela decyzyjna przedstawiona jest w postaci:

$$S = (U, A \cup \{dec\}), \tag{4.1}$$

gdzie:

U jest zbiorem obiektów: $U = \{u_1, ...u_n\}$,

A jest zbiorem atrybutów postaci $a_l : U \rightarrow V_l$,

dec jest specjalnym atrybutem zwanym decyzją $dec : U \rightarrow \{1, ..., d\}$.

Przygotowana tabela decyzyjna składa się z sześciu atrybutów warunkowych oraz jednego atrybutu decyzyjnego *category*, który określa do jakiego folderu przypisana zostaje wiadomość. Zestawienie atrybutów i ich objaśnienie znajduje się w tab. 4.3.

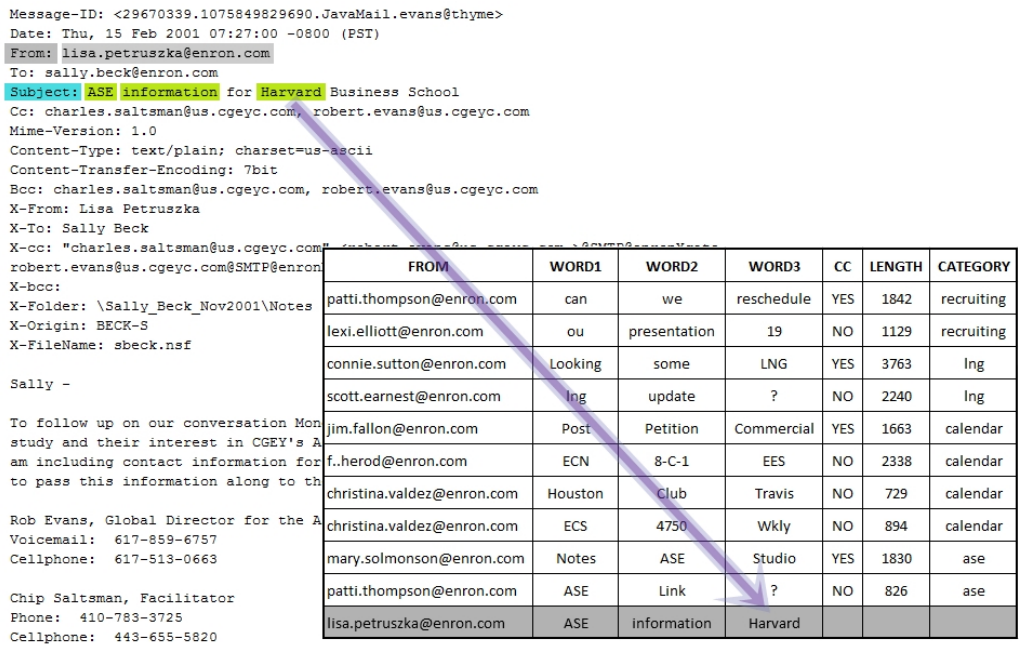
Atrybuty warunkowe wybrano w taki sposób, aby określały najważniejsze informacje o każdej wiadomości. Składają się z nadawcy, trzech pierwszych słów z tematu maila, informacji w postaci wartości boolowskiej, czy osoba, która otrzymała wiadomość była dodana do kopii maila (jeśli nie, to znaczy, że była adresatem) oraz długości maila. Ponadto z tematu maila pominięto podstawowe zwroty oraz łączniki, natomiast dodatkowo wspierano słowa, która należały do zbioru klas decyzyjnych. Opierając się na badaniach przeprowadzonych przez

Tabela 4.3: Zestawienie atrybutów w tabeli decyzyjnej

Nazwa	Opis atrybutu
from	nadawca wiadomości;
*word*1	pierwsze słowo z tematu maila (z wyłączeniem podstawowych słów i łączników), dodatkowo wspierane są słowa, która należą do zbioru klas decyzyjnych;
*word*2	drugie słowo ustalane analogicznie do word1;
*word*3	trzecie słowo ustalane analogicznie do word1 i word2;
cc	wartość boolowska oznaczająca, czy osoba, która otrzymała wiadomość była dodana do kopii maila (jeśli nie, to znaczy, że była adresatem);
length	liczba znaków maila;
category	klasa decyzyjna, folder, do którego przypisana zostaje wiadomość.

J. M. Carmona-Cejudo i in. w pracach [18, 19], postanowiono nie analizować treści wiadomości e-mail, gdyż najważniejsze informacje są zawarte w pierwszych pięciu liniach wiadomości, a badanie na większych fragmentach maila nie poprawiają wydajności.

Sposób przekształcenia wiadomości e-mail do struktury tabeli decyzyjnej został przedstawiony na rys. 4.3. Każdy wiersz takiej tabeli opisuje jedną wiadomość na podstawie zawartych w niej atrybutów.



Rysunek 4.3: Przekształcenie zbioru Enron E-mail do tabeli decyzyjnej

Do badań zostały wybrane takie skrzynki pocztowe, które umożliwiają porównanie otrzymanych wyników z innymi algorytmami. Parametry wybranych skrzynek zostały przeliczone i podane dla każdego zbioru w tab. 4.4. Są to duże zbiory danych o bardzo dużej liczbie klas decyzyjnych i dużej liczbie wartości atrybutów, w większości przypadków o wartościach ciągłych. Liczba klas decyzyjnych zależna jest od analizowanego przypadku i podana została dla każdego zbioru danych w tab. 4.4.

Tabela 4.4: Parametry wybranych zbiorów danych po przekształceniu do tabel decyzyjnych

Skrzynka pocztowa	Liczba obiektów	Liczba klas	Liczba wartości atrybutów					
			from	word1	word2	word3	cc	length
beck-s	1971	101	390	527	670	549	2	1331
farmer-d	3672	25	412	827	985	864	2	1679
germany-c	1013	19	207	382	419	340	2	835
haedicke-m	112	16	64	70	85	67	2	110
kaminski-v	4477	41	821	1231	1304	1058	2	2461
kitchen-l	4015	46	597	1170	1207	996	2	2138
lokay-m	2493	11	295	842	955	863	2	1654
mann-k	1616	24	254	394	490	433	2	1248
rogers-b	1395	14	289	445	521	430	2	1101
sanders-r	1188	30	272	442	485	423	2	1033
scott-s	641	10	135	350	219	166	2	578
shackleton-s	1001	53	158	330	384	357	2	836
shapiro-r	1970	75	325	720	856	754	2	1566
steffes-j	625	23	157	242	341	300	2	555
symes-k	770	12	119	324	346	287	2	685
taylor-m	656	21	173	255	288	245	1	580
williams-w3	2769	18	196	523	597	540	2	1056

5

Analiza wiadomości e-mail z zastosowaniem klasycznych algorytmów eksploracji danych

Pierwsze badania w tej rozprawie zostały przeprowadzone przy użyciu systemu RSES (ang. Rough Set Exploration System) [77]. System ten umożliwia przeprowadzenie eksperymentów na danych tablicowych z zastosowaniem teorii zbiorów przybliżonych. Pomysł na stworzenie systemu powstał w roku 1993 podczas przygotowywania pracy magisterskiej w Zakładzie Logiki Instytutu Matematyki Uniwersytetu Warszawskiego. Rok później, pod kierunkiem zespołu badawczego składającego się z naukowców kilku polskich uczelni wyższych, powstała pierwsza wersja systemu RSES napisanego w języku C++. Obecnie dostępna jest wersja systemu RSES 2.1 napisanego w języku Java. System ten został wyposażony w nowy, poprawiony i bardziej przyjazny interfejs użytkownika, a także dodano do systemu wiele ważnych metod obliczeniowych.

W celu sprawdzenia słuszności przekształcenia zbioru danych do postaci tabeli decyzyjnej podjęto próby przeprowadzenia badań z zastosowaniem algorytmu CART opisanego w rozdziale 2.1. Dodatkowo spośród wielu metod algorytmicznych pozwalających na analizę danych dostępnych systemie RSES wybrane zostały trzy z nich, a mianowicie:

- algorytm wyczerpujący (ang. Exhaustive algorithm) polegający na liczeniu wszystkich reguł z minimalną liczbą deskryptorów wyznaczające wszystkie reguły lokalne;
- algorytm pokryciowy (ang. Covering algorithm) polegający na znalezieniu podzbioru minimalnego rozmiaru, którego elementy pokrywają cały zbiór;
- algorytm genetyczny (ang. Genetic algorithm) opisany w rozdziale 1.5.

5.1 Przeprowadzone eksperymenty

Rozwiązanie proponowanej metody poprawiającej dokładność klasyfikacji wiadomości e-mail do folderów, która opisana została w rozdziale 4.5 zostało zaimplementowane w języku C++. Obliczenia wykonano na komputerze z procesorem Intel Core i5 2.27 GHz z 2.9 GB RAM. Komputer działał pod kontrolą systemu operacyjnego Debian GNU/Linux. Dla algorytmu genetycznego doświadczenia zostały powtórzone 30 razy dla każdego ze zbiorów danych, przy zachowaniu standardowych ustawień parametrów związanych z algorytmami genetycznymi. W przypadku pozostałych trzech algorytmów: Exhaustive, Covering oraz CART, doświadczenia zostały przeprowadzone tylko raz ze względu na deterministyczny charakter tych algorytmów. Wyniki przeprowadzonych badań zostały przedstawione w tab. 5.1, gdzie pogrubioną czcionką zaznaczone zostały najlepsze wyniki.

Tabela 5.1: Porównanie wybranych podejść pod względem dokładności klasyfikacji

Skrzynka pocztowa	Algorytmy z RSES			Alg. CART
	Genetic	Exhaustive	Covering	
beck-s	**0,593**	0,591	0,547	0,574
farmer-d	0,705	0,702	0,663	**0,778**
kaminski-v	0,545	0,542	0,541	**0,670**
kitchen-l	0,472	0,468	0,427	**0,597**
lokay-m	0,731	0,729	0,698	**0,824**
sanders-r	**0,742**	0,739	0,737	0,659
williams-w3	0,895	0,895	0,888	**0,947**

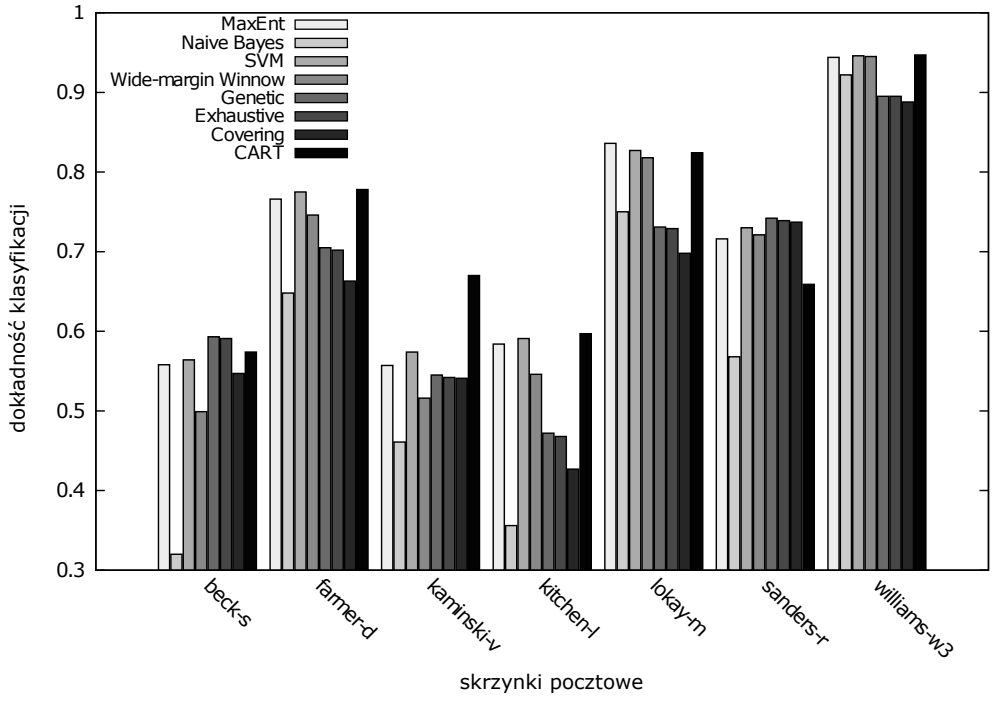
Z dostępnego zbioru danych Enron wybrano siedem skrzynek pocztowych, tak, aby możliwe było porównanie z innymi algorytmami z artykułu [10], które przedstawione zostało w tab. 5.2. Pogrubioną czcionką zaznaczone zostały najlepsze wyniki.

Tabela 5.2: Porównanie z innymi algorytmami z artykułu R. Bekkermana [10]

Skrzynka pocztowa	Dane z artykułu [10]				Alg. CART
	MaxEnt	Naive Bayes	SVM	Winnow	
beck-s	0,558	0,320	0,564	0,499	0,574
farmer-d	0,766	0,648	0,775	0,746	**0,778**
kaminski-v	0,557	0,461	0,574	0,516	**0,670**
kitchen-l	0,584	0,356	0,591	0,546	**0,597**
lokay-m	**0,836**	0,750	0,827	0,818	0,824
sanders-r	0,716	0,568	**0,730**	0,721	0,659
williams-w3	0,944	0,922	0,946	0,945	**0,947**

Uzyskane rezultaty przedstawione w tab. 5.1 i 5.2 oraz na rys. 5.1 wskazu-

ją na znaczną poprawę w przypadku zastosowania proponowanej metody. Jest to szczególnie interesujące ze względu na znaczący proces oczyszczenia zbiorów danych zastosowany w przypadku algorytmów opisanych przez R. Bekkermana [10]. Proponowana metoda na obecnym etapie nie wymaga dużych nakładów pracy związanych z odpowiednim przygotowaniem zbioru danych do badań, a jej adaptacyjność pozwala na uzyskanie stabilnych rezultatów nawet w przypadku nieoczyszczonych, rzeczywistych zbiorów danych.



Rysunek 5.1: Dokładność klasyfikacji proponowanej metody w stosunku do artykułu R. Bekkermana [10]

Proponowane rozwiązanie tylko w jednym przypadku (lokay-m) uzyskuje gorsze wyniki, natomiast we wszystkich pozostałych przypadkach uzyskiwane są lepsze rezultaty. Dla dwóch zbiorów danych (beck-s, sanders-r) poprawa dokładności przypisania wiadomości e-mail do folderu względem najlepszej z porównywanych metod z artykułu [10] uzyskana została przy użyciu algorytmu genetycznego i wynosi 2-3%. Natomiast w przypadku czterech pozostałych zbiorów danych (farmer-d, kaminski-v, kitchen-l, williams-w3) poprawa dokładności uzyskana została przy użyciu algorytmu CART, gdzie najlepszy wynik wynosi 10% dla zbioru kaminski-v. W przeprowadzonych badaniach użyte zostały także algorytmy wyczerpujący (ang. Exhaustive algorithm) oraz pokryciowy (ang. Covering algorithm) i jak przedstawiono w tab. 5.1 zastosowanie tych algorytmów nieznacznie poprawia dokładność klasyfikacji wiadomości e-mail do folderu w stosunku do metod zaproponowanych przez R. Bekkermana [10] w dwóch przypadkach (beck-s, sanders-r),

jednak dokładność ta jest lepsza przy zastosowaniu algorytmu genetycznego.

Zaproponowane podejście wykazało poprawę klasyfikowania wiadomości e-mail do folderów. Przygotowana tabela decyzyjna, w szczególności przy zastosowaniu algorytmu CART, pozwoliła na otrzymanie satysfakcjonujących wyników. Na podstawie przeprowadzonych eksperymentów potwierdzone zostało znaczne poprawienie dokładności klasyfikacji, a więc trafności automatycznej kategoryzacji wiadomości e-mail przy zastosowaniu algorytmu CART.

5.2 Analiza statystyczna

Statystyczna analiza wyników potwierdza opisane rezultaty. Wyniki eksperymentalne proponowanych metod są porównywane przy użyciu testu nieparametrycznej hipotezy statystycznej, tj. Testu Friedmana dla α= 0,05. Parametry związane z testem Friedmana są przedstawione w tabelach 5.3 i 5.5. Dodatkowo przeanalizowano średnie rangi analizowanych podejść (pogrubiona czcionka wskazuje na najlepszy wynik).

Tabela 5.3: Wyniki testu Friedmana i średnie wartości rankingowe dla danych z tab. 5.1

	Wartość
N	7
Chi-Kwadrat	12,826087
Liczba stopni swobody	3
Wartość p jest mniejsza niż	0,0050
5% krytyczna różnica	0,975144
Średnie rangi	
Genetic	1,78571428571
Exhaustive	2,64285714286
Covering	3,85714285714
alg. CART	**1,71428571429**

Tabela 5.4: Statystyczne różnice pomiędzy algorytmami dla danych z tab. 5.1

	Genetic	Exhaustive	Covering	alg, CART
Genetic	–	0,857143	**2,071429**	-0,071429
Exhaustive	-0,857143	–	**1,214286**	-0,928571
Covering	**-2,071429**	**-1,214286**	–	**-2,142857**
alg. CART	0,071429	0,928571	**2,142857**	–

W tabelach 5.4 i 5.6 przedstawiono statystyczne porównanie alg. CART i innych omówionych metod jako różnice w rankingu między porównywanymi algorytmami. Czcionki pogrubione wskazują wartości, które spełniają kryterium

Tabela 5.5: Wyniki testu Friedmana i średnie rangi dla danych z tab. 5.2

	Wartość
N	7
Chi-Kwadrat	20,457143
Liczba stopni swobody	4
Wartość p jest mniejsza niż	0,0004
5% krytyczna różnica	0,977931
Średnie rangi	
MaxEnt	2,85714285714
Naive Bayes	5,0
SVM	1,85714285714
Winnow	3,57142857143
alg. CART	**1,71428571429**

Tabela 5.6: Statystyczne różnice pomiędzy algorytmami dla danych z tab. 5.2

	MaxEnt	Naive Bayes	SVM	Winnow	alg, CART
MaxEnt	–	**2,142857**	**-1,000000**	0,714286	**-1,142857**
Naive Bayes	**-2,142857**	–	**-3,142857**	**-1,428571**	**-3,285714**
SVM	**1,000000**	**3,142857**	–	**1,714286**	-0,142857
Winnow	-0,714286	**1,428571**	**-1,714286**	–	**-1,857143**
alg. CART	**1,142857**	**3,285714**	0,142857	**1,857143**	–

5% krytycznej różnicy. Wartości poniżej zera wskazują, że analizowany algorytm (wiersz w tabeli) jest gorszy niż algorytm porównany (kolumna w tabeli).

6

Analiza wiadomości e-mail z zastosowaniem algorytmów mrowiskowych

Dokładna analiza przeprowadzonych doświadczeń opisanych w rozdziale 5 pozwoliła przypuszczać, że dla zaproponowanej metody tworzenia tabel decyzyjnych możliwe jest uzyskanie wysokiej dokładności klasyfikacji również przy zastosowaniu innych klasyfikatorów opartych na tabelach decyzyjnych. Dodatkowo zaobserwowano, że zastosowanie algorytmów do konstruowania drzew decyzyjnych może przyczynić się do uzyskania jeszcze lepszych rezultatów. Dlatego postanowiono przeprowadzić doświadczenia z zastosowaniem algorytmów mrowiskowych.

6.1 Zastosowanie algorytmu mrowiskowego aACDT

Zaproponowana metoda polega na zastosowaniu zmodyfikowanej wersji algorytmu aACDT oraz przetworzeniu zbioru Enron E-mail do postaci tabeli decyzyjnej, opisanej w rozdziale 4.5, gdzie tabela decyzyjna rozumiana jest jako struktura opisana wzorem (4.1). Dla tak przygotowanego zbioru danych wykonywany został algorytm aACDT z elementami analizy sieci komunikacji polegającej na analizowaniu listy odbiorców.

Modyfikacja algorytmu aACDT polega na badaniu sieci komunikacji (wzór 6.1) pomiędzy osobami w przypadku, kiedy wiadomość została wysłana do grupy osób $cc = true$. Analizowana jest lista wszystkich odbiorców, co ma wpływ na wybraną przez klasyfikator klasę decyzyjną (folder dla wiadomości). Na podejmowaną decyzję wpływ miały preferencje grupy kontaktujących się ze sobą użytkowników, dlatego jeśli użytkownicy kontaktowali się ze sobą z tą samą częstotliwością, to wiadomość jaką otrzymywali była klasyfikowana tak samo, zgodnie ze wzorem:

$$dDF(x) := \arg\max_{c} N_{c_f}(u), \tag{6.1}$$

gdzie c_f jest folderem, do którego przypisano wiadomość e-mail, natomiast $N_{c_f}(u)$ jest liczbą głosów oddanych na e-mail $u \in U$ w klasyfikacji do folderu c_f, taki że:

$$N_{c_f}(u) \ := \{ \ \overline{\overline{vot}} \ : \ p_{vot}(u) = c_f \}, \tag{6.2}$$

gdzie p jest osobą, która otrzymała wiadomość.

Sposób działania tak zaproponowanego algorytmu zaprezentowany jest w postaci pseudokodu alg. 4.

Algorytm 4: Pseudokod proponowanego algorytmu

```
zbiór_danych = przygotuj_tabelę_decyzyjną(osoba)
feromon = inicjalizuj_ślad_feromonowy();
for liczba_iteracji do
    najlepszy_klasyfikator = NULL;
    for liczba_mrówek do
        nowy_klasyfikator = konstruuj_klasyfikator_ACDT(feromon, zbiór_danych);
        nowy_klasyfikator = sprawdź_kontakty_SNA(nowy_klasyfik, zbiór_danych);
        oceń_jakość_klasyfikatora(nowy_klasyfikator);
        if nowy_klasyfikator jest_lepszej_jakości_od najlepszy_klasyfikator then
            najlepszy_klasyfikator = nowy_klasyfikator;
        endIf
    endFor
    aktualizuj_ślad_feromonowy(najlepszy_klasyfikator, feromon);
endFor
wynik = najlepszy_klasyfikator;
```

Doświadczenia zostały powtórzone 30 razy dla każdego ze zbiorów danych, przy zachowaniu standardowych ustawień parametrów związanych z algorytmami mrowiskowymi (przyjętych dla algorytmu ACDT [11]). W związku z wielkością zbioru danych liczba generacji algorytmu mrowiskowego została wstępnie ograniczona do 30 przy populacji 5 mrówek. Czas pracy proponowanego algorytmu wahał się w zależności od zbioru danych, od 7 do 400 sekund na jedno uruchomienie algorytmu. Jest to jednak czas tworzenia klasyfikatora, a sama klasyfikacja realizowana jest w bardzo szybkim czasie.

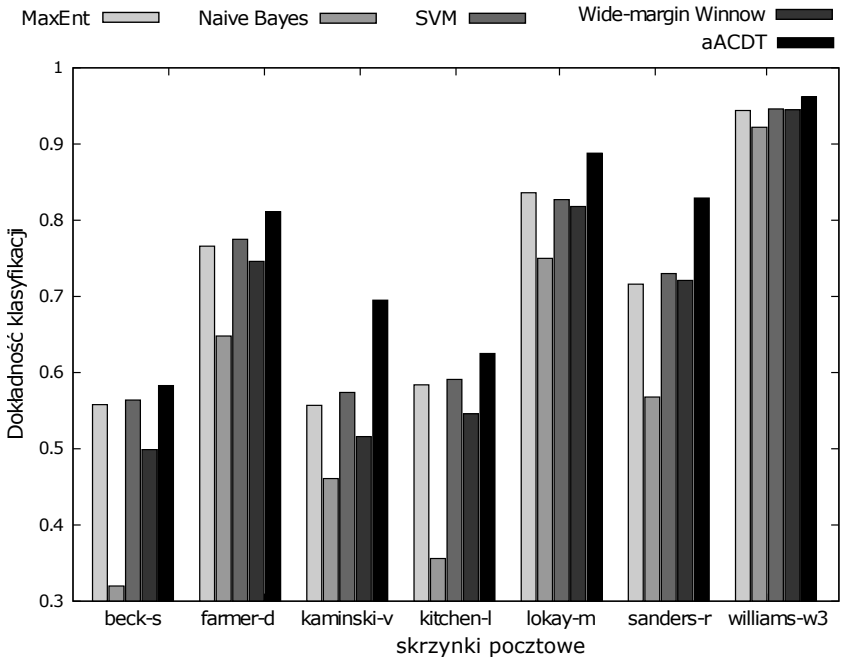
Wyniki badań przedstawione zostały w tab. 6.1 i na rys. 6.1, natomiast wyniki dla pozostałych algorytmów zapożyczone zostały z artykułu R. Bekkermana [10]. Pogrubioną czcionką zaznaczone są najlepsze wyniki.

Dla siedmiu zbiorów danych stworzonych na podstawie siedmiu użytkowników (wybranych tak, aby możliwe było porównanie z innymi algorytmami) proponowany algorytm za każdym razem uzyskuje lepsze rezultaty. W przypadku trzech zbiorów danych (beck-s, farmer-d i williams-w3) poprawa dokładności przypisania folderu do e-mail względem najlepszej z porównywanych metod wynosi 2-3%,

Tabela 6.1: Porównanie wyników z pracy [10] dla algorytmów klasycznych z algorytmem mrowiskowym w połączeniu z analizą odbiorców.

Skrzynka	Dane z artykułu [10]				Alg.
pocztowa	MaxEnt	Naive Bayes	SVM	WMW	aACDT
beck-s	0,558	0,320	0,564	0,499	**0,583**
farmer-d	0,766	0,648	0,775	0,746	**0,811**
kaminski-v	0,557	0,461	0,574	0,516	**0,695**
kitchen-l	0,584	0,356	0,591	0,546	**0,625**
lokay-m	0,836	0,750	0,827	0,818	**0,888**
sanders-r	0,716	0,568	0,730	0,721	**0,829**
williams-w3	0,944	0,922	0,946	0,945	**0,962**

w dwóch innych (kitchen-l i lokay-m) poprawa jest na wysokim poziomie 5%, natomiast w przypadku dwóch zbiorów kaminski-v i sanders-r poprawa jest najlepsza i przekracza 10%.



Rysunek 6.1: Poprawność dokładności klasyfikacji proponowanej metody w stosunku do artykułu [10]

Dla otrzymanych wyników z tab 6.1 przeprowadzono analizę statystyczną przy użyciu testu Friedmana. W tab. 6.2 przedstawiono najważniejsze parametry dla tej statystyki oraz średnie wartości rankingowe dla analizowanych podejść. Najmniejszą wartość rankingu ma algorytm aACDT, co wskazuje na to, że jest on znacznie lepszy niż porównywane metody. W tabeli 6.3 przedstawiono statystyczne porównanie omówionych metod jako różnice w rankingu między porównywa-

nymi algorytmami. Czcionki pogrubione wskazują wartości, które spełniają kryterium 5% krytycznej różnicy dla przeprowadzonego testu. Wartości poniżej zera wskazują, że analizowany algorytm (wiersz w tabeli) jest gorszy niż algorytm porównany (kolumna w tabeli).

Tabela 6.2: Wyniki testu Friedmana i średnie rangi (pogrubioną czcionką zaznaczono najlepszą metodę) dla danych z tab. 6.1

	Wartość
N	7
Chi-Kwadrat	25,942857
Liczba stopni swobody	4
Wartość p jest mniejsza niż	0,0001
5% krytyczna różnica	0,510708
Średnie rangi	
MaxEnt	3,14285714286
Naive Bayes	5,0
SVM	2,14285714286
Winnow	3,71428571429
aACDT	**1,0**

Tabela 6.3: Statystyczne różnice pomiędzy algorytmami (pogrubioną czcionką zaznaczono krytyczne różnice) dla danych z tab. 6.1

	MaxEnt	Naive Bayes	SVM	Winnow	alg, CART
MaxEnt	–	**1,857143**	**-1,0**	**0,571429**	**-2,142857**
Naive Bayes	**-1,857143**	–	**-2,857143**	**-1,285714**	**-4,0**
SVM	**1,000000**	**2,857143**	–	**1,571429**	**-1,142857**
Winnow	**-0,571429**	**1,285714**	**-1,571429**	–	**-2,714286**
aACDT	**2,142857**	**4,0**	**1,142857**	**2,714286**	–

Ze względu na fakt, że autorzy pracy [10] przeprowadzili szereg czynności związany z oczyszczeniem zbioru danych Enron postanowiono przeprowadzić własne doświadczenia z udziałem klasycznych algorytmów. Opierając się na autorskim sposobie tworzenia tabel decyzyjnych wybrano siedem skrzynek pocztowych ze zbioru Enron E-mail w taki sposób, aby otrzymane wyniki można było porównać z poprzednimi badaniami, a także rozszerzono zakres eksperymentów na dziesięć innych zbiorów, których zajętość pamięci skrzynki pocztowej zawierała się w przedziale 10-42MB, co mogło świadczyć o dużej liczbie wiadomości e-mail i folderów. Do przeprowadzenia badań wybrany został zaadaptowany algorytm ACDT [11] oraz niektóre algorytmy z systemu WEKA (ang. Waikato Environment for Knowledge Analysis) [95]. Wyniki eksperymentów zostały przedstawione w tab. 6.4 i na rys. 6.2, a otrzymane rezultaty jednoznacznie potwierdziły, że proponowaną metodę tworzenia tabel decyzyjnych można zastosować z dowolnymi

klasyfikatorami, nawet przy uwzględnieniu nieoczyszczonych zbiorów danych.

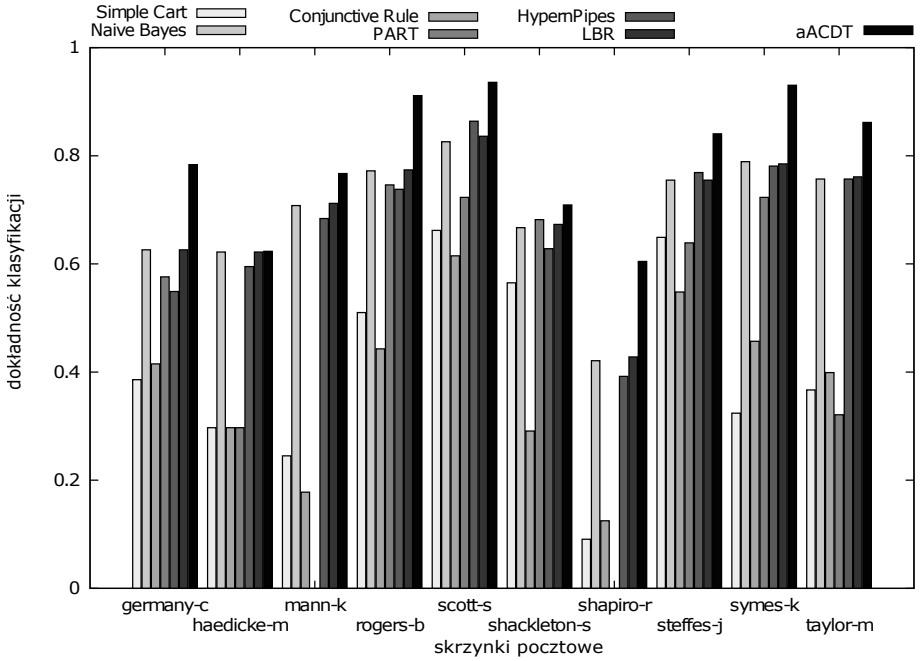
Tabela 6.4: Porównanie wyników algorytmów klasycznych z algorytmem mrowiskowym w połączeniu z analizą odbiorców - badania własne.

Skrzynka pocztowa	Simple Cart	Naive Bayes	Conjunctive Rule	PART	Hyper Pipes	LBR	Alg. aACDT
beck-s	0,332	0,559	0,085	0,553	0,537	0,565	**0,583**
farmer-d	0,643	0,731	0,433	0,699	0,629	0,739	**0,811**
kaminski-v	0,304	0,510	0,131	-	0,488	0,517	**0,695**
kitchen-l	0,255	0,458	0,190	-	0,446	0,504	**0,625**
lokay-m	0,775	0,818	0,550	0,764	0,679	0,818	**0,888**
sanders-r	0,354	0,667	0,391	0,354	0,669	0,667	**0,829**
williams-w3	0,843	0,898	0,868	0,925	0,898	0,901	**0,962**
germany-c	0,386	0,626	0,415	0,576	0,549	0,626	**0,783**
haedicke-m	0,297	0,622	0,297	0,297	0,595	0,622	**0,623**
mann-k	0,245	0,708	0,178	-	0,684	0,712	**0,767**
rogers-b	0,510	0,772	0,443	0,746	0,738	0,774	**0,911**
scott-s	0,662	0,826	0,615	0,723	0,864	0,836	**0,936**
shackleton-s	0,565	0,667	0,291	0,682	0,628	0,673	**0,709**
shapiro-r	0,091	0,421	0,125	-	0,392	0,428	**0,605**
steffes-j	0,649	0,755	0,548	0,639	0,769	0,755	**0,841**
symes-k	0,324	0,789	0,457	0,723	0,781	0,785	**0,930**
taylor-m	0,367	0,757	0,399	0,321	0,757	0,761	**0,862**

Dla wszystkich zbiorów danych stworzonych na podstawie dziesięciu użytkowników proponowany algorytm za każdym razem uzyskuje lepsze rezultaty. W przypadku trzech zbiorów danych (mann-k, scott-s, steffes-j) poprawa dokładności przypisania folderu do e-mail względem najlepszej z porównywanych metod wynosi 5-7%, w trzech innych (rogers-b, symes-k, taylor-m) poprawa jest na wysokim poziomie 10-14%, natomiast w przypadku zbiorów germany-c i shapiro-r poprawa jest bardzo wysoka i wynosi 16-18%.

W pozostałych dwóch zbiorach (haedicke-m i shackleton-s) dokładność klasyfikacji jest na tym samym poziomie we wszystkich zastosowanych algorytmach. Natomiast w przypadku zbiorów mann-k oraz shapiro-r nie zostały podane wyniki klasyfikacji z użyciem algorytmu PART ze względu za zbyt długi czas pracy tego algorytmu.

Dla otrzymanych wyników z tab 6.4 przeprowadzono analizę statystyczną przy użyciu testu Friedmana. Najważniejsze parametry dla tej statystyki oraz średnie wartości rankingowe dla analizowanych podejść przedstawiono w tab. 6.5. Najlepszą metodą sporód porównywanych podejść jest algorytm aACDT ze względu na najmniejszą wartość rankingową. W tabeli 6.6 przedstawiono statystyczne porównanie omówionych metod jako różnice w rankingu między porównywanymi algorytmami. Czcionki pogrubione wskazują wartości, które spełniają kryterium



Rysunek 6.2: Poprawność dokładności klasyfikacji proponowanej metody

5% krytycznej różnicy dla przeprowadzonego testu. Wartości poniżej zera wskazują, że analizowany algorytm (wiersz w tabeli) jest gorszy niż algorytm porównany (kolumna w tabeli).

Tabela 6.5: Wyniki testu Friedmana i średnie rangi (pogrubioną czcionką zaznaczono najlepszą metodę) dla danych z tab. 6.4

	Wartość
N	17
Chi-Kwadrat	79,914894
Liczba stopni swobody	6
Wartość p jest mniejsza niż	0,0001
5% krytyczna różnica	0,700994
Średnie rangi	
Simple Cart	5,79411764706
Naive Bayes	3,14705882353
Conjunctive Rule	6,23529411765
PART	5,20588235294
Hyper Pipes	4,11764705882
LBR	2,5
aACDT	**1,0**

Tabela 6.6: Statystyczne różnice pomiędzy algorytmami (pogrubioną czcionką zaznaczono krytyczne różnice) dla danych z tab. 6.4

	Simple Cart	Naive Bayes	Conjunctive Rule	PART	Hyper Pipes	LBR	Alg. aACDT
Simple Cart	–	**-2,647059**	0,441176	-0,588235	**-1,67647**	**-3,29411**	**-4,79411**
Naive Bayes	**2,647059**	–	**3,08823**	**2,05882**	**0,97058**	-0,647059	**-2,14705**
Conjunctive Rule	-0,441176	**-3,08823**	–	**-1,02941**	**-2,11764**	**-3,73529**	**-5,23529**
PART	0,588235	**-2,05882**	**1,02941**	–	**-1,08823**	**-2,70582**	**-4,20588**
Hyper Pipes	**1,676471**	**-0,97058**	**2,11764**	**1,08823**	–	**-1,61764**	**-3,11764**
LBR	**3,294118**	0,647059	**3,73529**	**2,70588**	**1,61764**	–	**-1,50000**
aACDT	**4,794118**	**2,14705**	**5,23529**	**4,20588**	**3,11764**	**1,50000**	–

Zaproponowane podejście wykazało znaczną poprawę klasyfikowania wiadomości e-mail do folderów. Już sama analiza specjalnie przygotowanej tabeli decyzyjnej przy zastosowaniu zaadaptowanego algorytmu ACDT pozwoliła na otrzymanie satysfakcjonujących wyników. Dodanie elementów sieci społecznych w postaci analizy komunikacji pomiędzy użytkownikami pozwoliło na znaczne poprawienie wyników.

Uzyskane rezultaty potwierdziły słuszność zastosowania tabel decyzyjnych do budowy klasyfikatorów opartych na algorytmach mrowiskowych. Zastosowanie algorytmu mrowiskowego na obecnym etapie nie wymagało dużych nakładów pracy związanych z odpowiednim przygotowaniem zbioru danych do badań, a jego adaptacyjność pozwala na uzyskanie stabilnych rezultatów nawet w przypadku nieoczyszczonych, rzeczywistych zbiorów danych.

Wspomniane możliwości adaptacji algorytmów mrowiskowych wpływają dodatkowo na możliwość ich pracy, przy bardzo niewielkim oczyszczaniu zbiorów danych. W przypadku Enron E-mail dataset możliwe jest tworzenie tabel decyzyjnych bez dalszego ich oczyszczania - wystarczy proces, który został wykonany przez autorów zbioru danych. Jest to znacząca poprawa w stosunku do porównywanych metod.

6.2 Zastosowanie zespołów klasyfikatorów

Na podstawie obserwacji z poprzednich doświadczeń postanowiono zbadać czy zastosowanie zespołu klasyfikatorów opartych na algorytmach mrowiskowych będzie lepszym rozwiązaniem dla analizowanego problemu niż stosowanie pojedynczego klasyfikatora. W tym celu do przeprowadzenia badań wybrany został algorytm mrowiskowy do konstruowania lasów decyzyjnych (ang. Ant Colony Decision Forest, ACDF) [14] oraz niektóre algorytmy z systemu WEKA (ang. Waikato Environment for Knowledge Analysis) [95].

W systemie WEKA dostępnych jest wiele metod algorytmicznych pozwalających na analizę danych. Jednak do przeprowadzenia badań wybrane zostały cztery zespoły klasyfikatorów, a mianowicie:

- algorytm AdaBoost w połączeniu z algorytmami CART oraz RandomTree,
- algorytm Dagging w połączeniu z algorytmami CART oraz RandomTree,
- algorytm Bagging w połączeniu z algorytmem CART,
- algorytm Random Forest.

Do przeprowadzonych badań zostało wybranych siedem skrzynek pocztowych ze zbioru Enron E-mail. Wszystkie zbiory zawierają nieoczyszczone dane, w związku z czym do badań wzięto wszystkie wiadomości występujące w danym zbiorze, a nie tylko wybrane. Każdy z siedmiu zbiorów danych został podzielony na zbiór

treningowy i zbiór testowy. Na podstawie zbioru treningowego zostały wygenerowane reguły decyzyjne. W kolejnym kroku proponowanej metody została sprawdzona skuteczność tych reguł na podstawie zbioru testowego przy zastosowaniu wybranych algorytmów.

Doświadczenia zostały powtórzone 30 razy dla każdego ze zbiorów danych, przy zachowaniu standardowych ustawień parametrów przyjętych dla algorytmu ACDF [11], które wynoszą odpowiednio: $q_0 = 0,3$, $\alpha = 3,0$, $\gamma = 0,1$, $\phi = 0,05$, $\psi = 1,0$ oraz $\lambda = 0,5$. Każde doświadczenie obejmuje 25 pokoleń o wielkości populacji kolonii mrówek równej 625. W każdej decyzji las składał się z 25 drzew.

W tab. 6.7 przedstawiono wyniki trafności przypisania wiadomości e-mail do folderów dla wybranych zespołów klasyfikatorów oraz dla algorytmu ACDF. Przedstawione wyniki są średnią arytmetyczną z wszystkich uruchomień poszczególnych algorytmów, jednak dla algorytmu ACDF za każdym razem otrzymywano wyniki lepsze niż przy użyciu innych wybranych zespołów klasyfikatorów.

Tabela 6.7: Porównanie wyników z zastosowaniem zespołów klasyfikatorów.

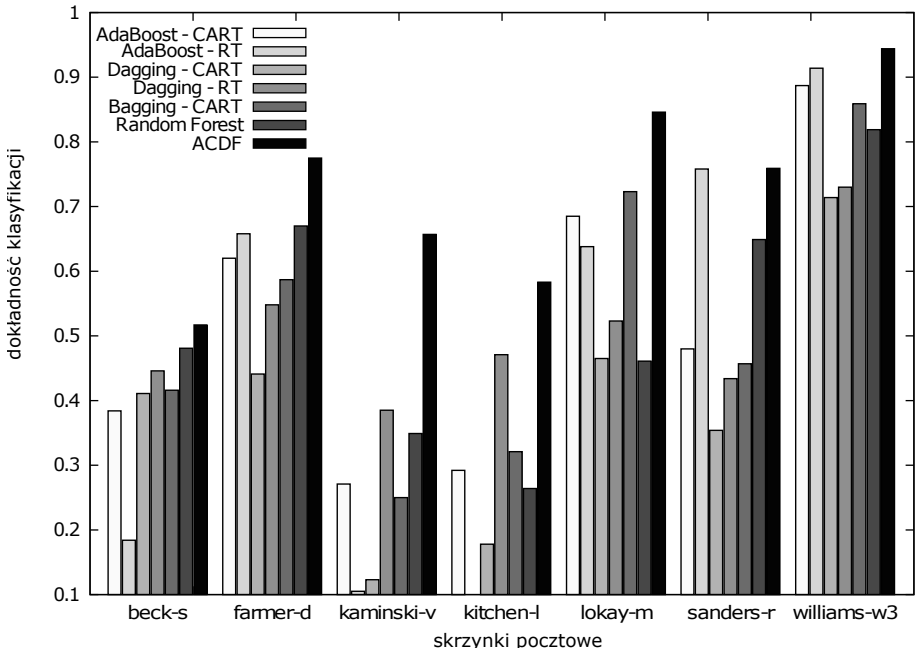
Skrzynka pocztowa	AdaBoost		Dagging		Bagging	Random	
	CART	Random Tree	CART	Random Tree	CART	Forest	ACDF
beck-s	0,384	0,184	0,411	0,446	0,416	0,481	**0,517**
farmer-d	0,620	0,658	0,441	0,548	0,587	0,670	**0,775**
kaminski-v	0,271	0,105	0,123	0,385	0,250	0,349	**0,657**
kitchen-l	0,292	–	0,178	0,471	0,321	0,264	**0,583**
lokay-m	0,685	0,638	0,465	0,523	0,723	0,461	**0,846**
sanders-r	0,480	0,758	0,354	0,434	0,457	0,649	**0,759**
williams-w3	0,887	0,914	0,714	0,730	0,859	0,819	**0,944**

W tab. 6.7 i 6.8 przedstawiono wyniki dokładności klasyfikacji wiadomości e-mail do folderów dla algorytmu ACDF oraz wyniki pojedynczego najlepszego drzewa dla każdego zbioru danych.

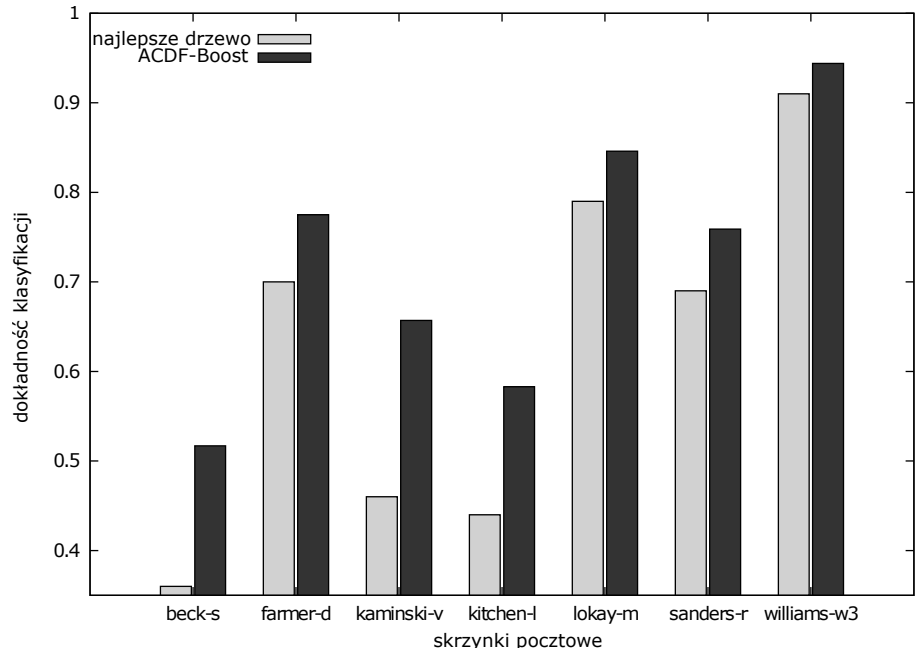
Tabela 6.8: ACDF-Boost i pojedyncze drzewo z tego lasu

data set	Best Tree	ACDF-Boost
beck-s	0,389	**0,517**
farmer-d	0,707	**0,775**
kaminski-v	0,498	**0,657**
kitchen-l	0,478	**0,583**
lokay-m	0,818	**0,846**
sanders-r	0,721	**0,759**
williams-w3	0,937	**0,944**

Na rys. 6.3 przedstawiona została dokładność klasyfikacji uzyskana przy zastosowaniu algorytmu ACDF, natomiast na rys. 6.4 przedstawiono najlepsze po-



Rysunek 6.3: Dokładność klasyfikacji uzyskana przy zastosowaniu algorytmu ACDF oraz algorytmów z systemu WEKA



Rysunek 6.4: Najlepsze pojedyncze drzewa decyzyjne dla algorytmu ACDF

jedyncze drzewa decyzyjne znajdujące się w danym zestawie danych. Jak można zauważyć, pomimo słabych pojedynczych klasyfikatorów, algorytm ACDF uzyskuje bardzo dobre wyniki klasyfikacji, co jest znamienne dla zespołów klasyfikatorów.

Przeprowadzone doświadczenia potwierdzają, że zastosowanie adaptacyjnego zespołu klasyfikatorów (ACDF) pozwala na kategoryzację wiadomości e-mail z wysoką trafnością przypisania do folderów. Klasyczne zespoły klasyfikatorów dla zaproponowanej metody nie prowadzą do otrzymania tak dobrych wyników i często klasyfikują wiadomości ze znacznie mniejszą precyzją (nawet 30-40%). Jednak po porównaniu otrzymanych wyników z poprzednimi badaniami, okazało się, że zastosowanie zespołu klasyfikatorów daje gorsze rezultaty niż pojedynczy klasyfikator z analizą odbiorców.

Statystyczna analiza wyników potwierdza opisane rezultaty. Parametry testu Friedmana zostały przedstawione w tab. 6.9. Dodatkowo przeanalizowano średnie wartości rankingu analizowanych podejść. W tabeli 6.10 przedstawiono statystyczne porównanie omówionych metod jako różnice w rankingu między porównywanymi algorytmami. Czcionki pogrubione wskazują wartości, które spełniają kryterium 5% krytycznej różnicy dla przeprowadzonego testu. Wartości poniżej zera wskazują, że analizowany algorytm (wiersz w tabeli) jest gorszy niż algorytm porównany (kolumna w tabeli).

Tabela 6.9: Wyniki testu Friedmana i średnie rangi (pogrubioną czcionką zaznaczono najlepszą metodę) dla danych z tab. 6.7

	Wartość
N	7
Chi-Kwadrat	21,979592
Liczba stopni swobody	6
Wartość p jest mniejsza niż	0,0012
5% krytyczna różnica	1,746315
Średnie rangi	
Ada Boost CART	4,0
Ada Boost Random Tree	4,57142857143
Dagging CART	6,28571428571
Dagging Random Tree	4,28571428571
Bagging CART	4,0
Random Forest	3,85714285714
ACDF	**1,0**

Tabela 6.10: Statystyczne różnice pomiędzy algorytmami (pogrubioną czcionką zaznaczono krytyczne różnice) dla danych z tab. 6.7

	AdaBoost		Dagging		Bagging	Random	ACDF
	CART	Random Tree	CART	Random Tree	CART	Forest	
AdaBoost CART	–	0,571429	**2,285714**	0,285714	0,0	-0,142857	**-3,0**
AdaBoost Random Tree	-0,571429	–	1,714286	-0,285714	-0,571429	-0,714286	**-3,571429**
Dagging CART	**-2,285714**	-1,714286	–	**-2,0**	**-2,285714**	**-2,428571**	**-5,285714**
Dagging Random Tree	-0,285714	0,285714	**2,0**	–	-0,285714	-0,428571	**-3,285714**
Bagging CART	0,0	0,571429	**2,285714**	0,285714	–	-0,142857	**-3,0**
Random Forest	0,142857	0,714286	**2,428571**	0,428571	0,142857	–	**-2,857143**
ACDF	**3,0**	**3,571429**	**5,285714**	**3,285714**	**3,0**	**2,857143**	–

7

Tworzenie mapy kontaktów oraz analiza sieci społecznych

Kolejnym celem rozprawy jest stworzenie sieci społecznej opartej na kontaktach pomiędzy nadawcą a odbiorcami wiadomości e-mail, a także na podstawie analizy i obserwacji sieci społecznej wyodrębnienie grupy użytkowników posiadających podobną strukturę społeczną. Aby zrealizować ten cel w pierwszej kolejności należy zbudować sieć społeczną dla całego zbioru Enron E-mail, a następnie przeprowadzić analizę całej sieci oraz odpowiednich fragmentów, co w efekcie pozwoli doprowadzić do wyodrębnienia grup użytkowników.

Analiza kontaktów pomiędzy poszczególnymi pracownikami korporacji przeprowadzona jest w celu wyznaczenia liderów z punktu widzenia rozprzestrzeniania się informacji lub wpływania na osoby będące w bezpośrednim sąsiedztwie. Analiza ta w dalszych pracach pozwoli na stworzenie algorytmu, którego zastosowanie posłuży do poprawienia dokładności klasyfikacji wiadomości e-mail do poszczególnych folderów w skrzynkach pocztowych pracowników firmy Enron. Odtworzenie mapy kontaktów w postaci sieci powiazań społecznych ma kluczowe znaczenie dla procesów przepływu informacji w korporacji.

Sieci społeczne związane są również z budowaniem sieci komunikacji [94], czyli procesem wymiany informacji, zasobów i możliwości, prowadzonym przy pomocy wzajemnie korzystnych kontaktów. Skupiając się na stworzeniu i analizie mapy powiązań społecznych na podstawie zbioru Enron E-mail, badania zostały podzielone na trzy etapy:

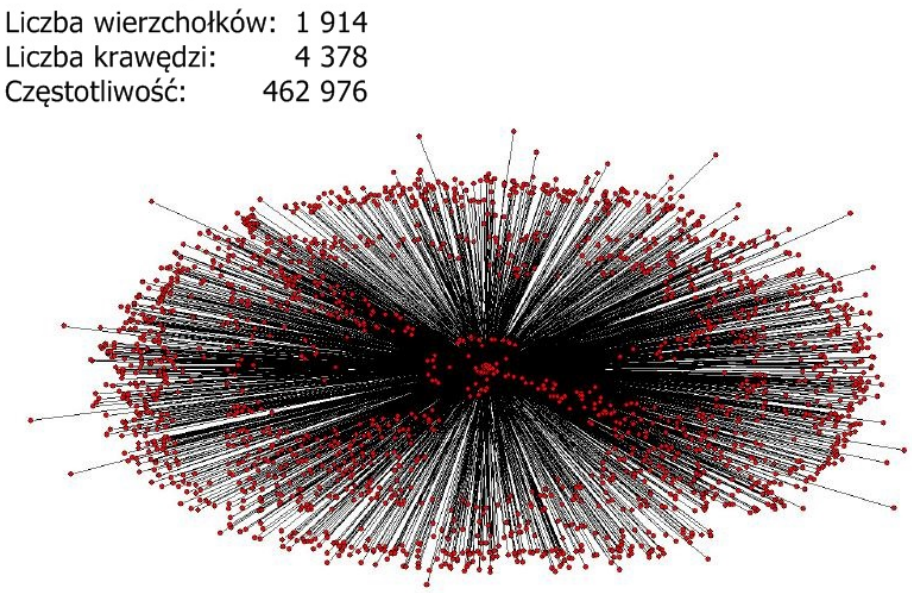
- analiza całej sieci (analiza makro),
- analiza części sieci (analiza meso),
- analiza poszczególnych pracowników (analiza mikro).

Dzięki takiemu podejściu do badań można otrzymać nie tylko ogólny obraz komunikacji w organizacji, ale przede wszystkim pozwala to na uzyskanie nieformalnej struktury firmy i mapy przepływu informacji w przedsiębiorstwie [84].

7.1 Sieć jako całość – analiza makro

Analiza makro to spojrzenie na organizację jako całość, dzięki czemu można określić charakter firmy pod względem komunikacji i współpracy wszystkich pracowników, a niekiedy także klientów. Poprzez mapowanie procesów komunikacji, czy analizy poziomu i struktury znajomości pracowników w danym przedsiębiorstwie, powstaje swoista, nieformalna struktura organizacyjna przedsiębiorstwa.

W pierwszym etapie przeprowadzonych badań dotyczących tworzenia i analizy sieci powiązań na podstawie zbioru Enron E-mail pod uwagę wzięte zostały wszystkie wysłane i odebrane wiadomości e-mail. Wierzchołkami w sieci zostały wszystkie adresy poczty elektronicznej występujące choć raz w całym zbiorze. Natomiast powiązania między tymi wierzchołkami to informacja o wysłaniu bądź odebraniu przynajmniej jednej wiadomości e-mail.



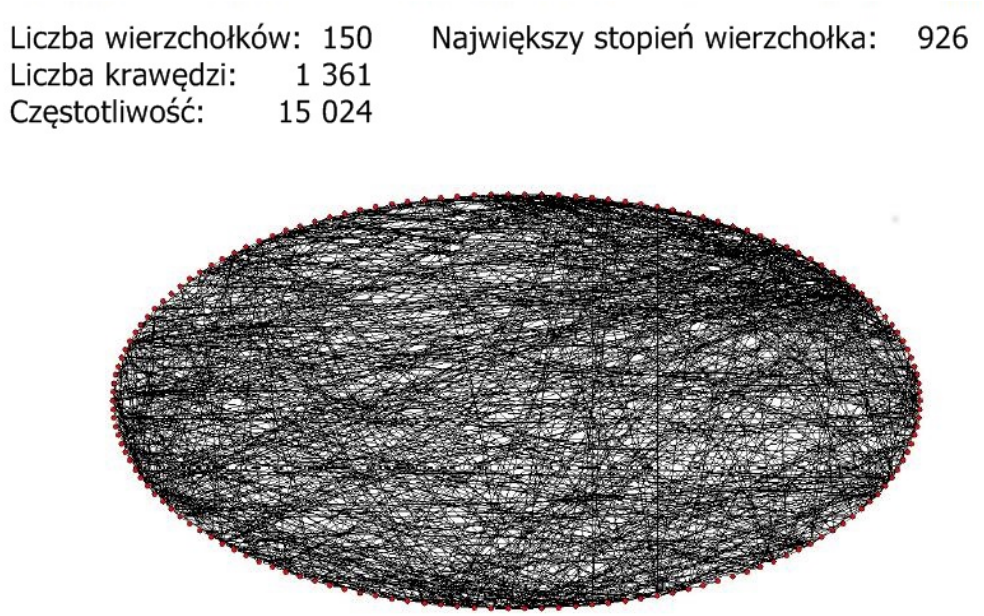
Rysunek 7.1: Wizualizacja sieci społecznej dla zbioru Enron E-mail

Na rys. 7.1 przedstawiona została wizualizacja sieci dla całego zbioru wiadomości firmy Enron. Sieć ta jest zbudowana z 1 914 obiektów, w skład których wchodzą nie tylko pracownicy korporacji Enron, ale także klienci i inne osoby zewnętrze, którzy kontaktowali się ze sobą za pomocą poczty elektronicznej. Wszystkie obiekty sieci połączone są ze sobą 4 378 krawędziami. Liczba ta wskazuje powiązania pomiędzy poszczególnymi osobami. Dla takiego zbioru obiektów i

krawędzi częstotliwość przepływu informacji, czyli liczba przesłanych wiadomości e-mail wynosi 462 976.

7.2 Analiza podsieci – analiza meso

W kolejnym etapie przeprowadzonych badań skupiono się analizie grupy pracowników będących na stanowiskach menadżerskich firmy Enron. Analiza sieci ograniczona do pewnej grupy społecznej w ramach danego przedsiębiorstwa nosi nazwę analizy meso. Analiza ta skupia się na wewnętrznych relacjach danej grupy obiektów, wyodrębnionych ze względu na formalne kryteria podziału, tj. przynależność do odpowiednich działów, staż pracy lub stanowisko. Stosując tą metodę można określić nieformalne grupy pracowników, którzy w szczególny sposób ze sobą współpracują bądź komunikują się ze sobą, dzięki posiadanej wiedzy, bądź uczestniczą w tym samym procesie dotyczącym np. danego projektu.



Rysunek 7.2: Wizualizacja sieci społecznej dla 150 skrzynek pocztowych

Skrzynki pocztowe tych osób wraz z utworzonymi folderami i przypisanymi do nich wiadomościami e-mail dostępne są w postaci zbioru Enron e-mail. Skrzynek tych jest 150, znajduje się w nich ok. 600 tys. wiadomości e-mail. Na potrzeby stworzenia sieci społecznej z badanego zbioru danych zostały usunięte wszystkie wiadomości e-mail wysłane przez pracowników firmy na własną skrzynkę pocztową. Ze zbioru wyeliminowane zostały także foldery utworzone przez programy pocztowe w sposób automatyczny, które w nazwie zawierają hasła *sent* lub *inbox*.

Dodatkowo usunięto wiadomości, których nadawca lub odbiorca występował tylko jeden raz w całym zbiorze Enron E-mail.

Po takiej modyfikacji w zbiorze zostało 150 skrzynek pocztowych zawierających wiadomości e-mail przypisane do różnych folderów. W tej części przeprowadzonych badań stworzono sieć społeczną zawierającą 150 obiektów, które powiązane były ze sobą 1 361 krawędziami. Pomiędzy tymi osobami zostało przesłanych 15 024 wiadomości e-mail co jest określane jako częstotliwość przepływu informacji. Wizualizacja opisanej sieci społecznej została przedstawiona na rys. 7.2, gdzie najwyższy stopień wierzchołka, czyli liczba krawędzi wchodzących i wychodzących z danego obiektu wynosi 926.

7.3 Analiza sieci dla najważniejszego obiektu – analiza mikro

W kolejnym kroku przeprowadzonych badań skupiono się na stworzeniu oraz analizie sieci społecznej dotyczącej przepływu wiadomości e-mail pomiędzy jednym pracownikiem a pozostałymi osobami. Pracownik ten został wybrany na podstawie kryterium przepływu wiedzy i informacji, w związku z czym został określony jako najważniejszy obiekt w sieci, ponieważ z jego skrzynki pocztowej zostało przesłanych najwięcej wiadomości e-mail.

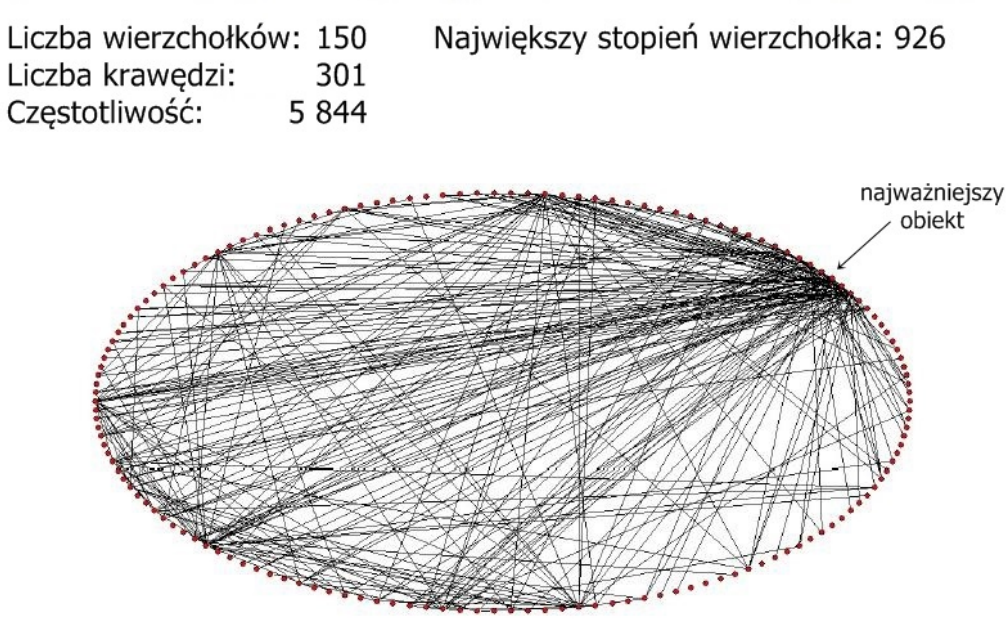
Metody badań SNA pozwalają na analizowanie małego wycinka sieci jakim jest sieć relacji poszczególnego pracownika. Taka analiza nosi nazwę analizy mikro. Dzięki tej metodzie istnieje możliwość zidentyfikowania pracowników tworzących tzw. wąskie gardła w ramach procesu przepływu informacji, ale także pracowników będących liderami w swojej dziedzinie.

Na rys. 7.3 przedstawiona jest wizualizacja sieci społecznej dla najważniejszego obiektu, wybranego ze względu na największy stopień wierzchołka. Przedstawiona sieć składa się ze 150 obiektów, połączonych ze sobą 301 krawędziami. Częstotliwość przepływu informacji wynosi 5 844 przesłanych wiadomości e-mail.

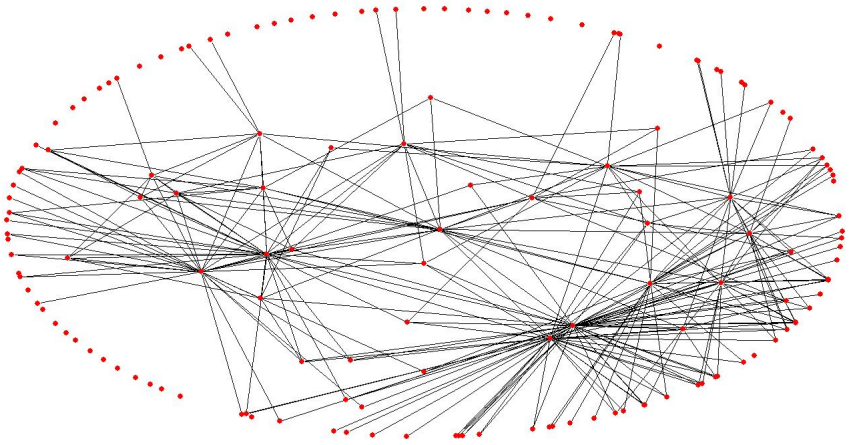
Ze względu na brak kontaktów niektórych pracowników z najważniejszym obiektem w sieci wyodrębnionych zostało 45 obiektów niepowiązanych ze sobą, co zostało przedstawione na rys. 7.4. Jednak należy pamiętać, że przedstawiona sieć dotyczy tylko relacji najważniejszego obiektu z pozostałymi. W przypadku wybrania innego obiektu jako najważniejszego, relacje w nowej sieci społecznej są zupełnie inne niż dotychczas przedstawione, a obiekty niepowiązane w tej sieci posiadają połączenia z innymi obiektami nowej sieci.

7.4 Wyodrębnienie grup

Badania dotyczące wyodrębnienia grup użytkowników ze zbioru pracowników Enron rozpoczęto od przeanalizowania zbioru danych Enron e-mail pod względem



Rysunek 7.3: Wizualizacja sieci społecznej dla najważniejszego obiektu



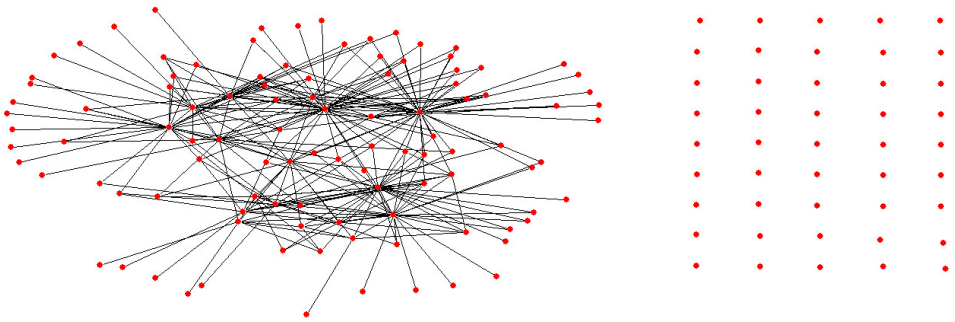
Rysunek 7.4: Analiza sieci społecznej dla najważniejszego obiektu

kontaktów pomiędzy nadawcą a odbiorcami oraz liczby przesyłania wiadomości. Jednak ze względu na brak dostępu do skrzynek pocztowych osób, które nie były pracownikami firmy Enron, wyodrębniono tylko te wiadomości, które były przesyłane pomiędzy pracownikami Enron. Wiadomości przesyłane pomiędzy pracownikami a osobami z zewnątrz zostały usunięte z analizowanego zbioru. Następnie stworzono macierz powiązań między nadawcą a odbiorcami, której fragment jest przedstawiony na rys. 7.5.

Nadawca \ Odbiorca	1	2	3	4	5	6	7	8	…	…	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	62	63	64	65	66	67	68	69	70	71	72	…
1		4																					9				3	6		9									
2								8														2		1				7									8	
3																						4																
4					3																							1										
…													5								76																	
…																																						
35																											3											
36											18	58						2															1	3				
37														8		15				2				1	1													
38															6											5		4										
39																																			2			
40		5														2								1				1								7		
41	11	4		2												4	30								5		12	9		14				5			
42																																6						
43																2																						
44				12																			84									3						
45																		1	5														24	3				
46											10							27			2		1										3	2				
47					7											6					133																	
48																																						
…																																						

Rysunek 7.5: Macierz powiązań pomiędzy nadawcą a odbiorcami

Na podstawie stworzonej macierzy powiązań utworzono sieć społeczną, którą przedstawiono na rys. 7.6. Wierzchołkami sieci są wszyscy użytkownicy skrzynek pocztowych ze zbioru Enron e-mail, natomiast krawędzie to połączenia, dla których częstotliwość interakcji, czyli liczba przesłanych wiadomości pomiędzy poszczególnymi osobami, wynosiła więcej niż 10. Wierzchołki sieci, które nie posiadają żadnych powiązań z innymi wierzchołkami to osoby, które miały mniej niż założona liczba przesłanych wiadomości e-mail.



Rysunek 7.6: Wizualizacja sieci do wyodrębnienia grup użytkowników

W stworzonej sieci społecznej obliczono stopnie wierzchołków w sieci zgodnie ze wzorem (3.7), a następnie wybrano węzły sieci o najwyższym stopniu wierzchołka, które razem z najbliższymi sąsiadami utworzyły poszczególne grupy użytkowników w sieci. Wszystkie stworzone grupy użytkowników zostały przedstawione w tab. 7.1.

Tabela 7.1: Wszystkie grupy użytkowników.

Nazwa grupy	Użytkownicy
GRUPA1	cash-m; haedicke-m; sager-e; sanders-r; taylor-m
GRUPA2	blair-l; corman-s; dasovich-j; horton-s; kean-s; scott-s
GRUPA3	badeer-r; corman-s; dasovich-j; kean-s; sanders-r; scott-s; shapiro-r; steffes-j
GRUPA4	germany-c; hodge-j; neal-s; parks-j; ruscitti-k
GRUPA5	cuilla-m; ermis-f; holst-k; lenhart-m; scott-s; shively-h; smith-m; wolfe-j
GRUPA6	causholli-m; semperger-c; slinger-r; solberg-g; symes-k; williams-w3
GRUPA7	cash-m; haedicke-m; kitchen-l; sager-e; sanders-r; taylor-m
GRUPA8	hyatt-k; lokay-m; scott-s; watson-k
GRUPA9	corman-s; dasovich-j; kean-s; lay-k; sanders-r; shapiro-r
GRUPA10	buy-r; haedicke-m; jones-t; kaminski-v; kitchen-l; lavorato-j; neal-s; presto-k; shively-h; storey-g; taylor-m; whalley-g; whalley-l; zipper-a
GRUPA11	arnold-j; kitchen-l; lavorato-j; neal-s; presto-k; shively-h; whalley-g; whalley-l
GRUPA12	hyatt-k; lokay-m; mcconnell-m; schoolcraft-d; scott-s; watson-k
GRUPA13	arnold-j; griffith-j; may-l; mclaughlin-e; quigley-d
GRUPA14	hodge-j; nemec-g; perlingiere-d; ward-k; whitt-m
GRUPA15	jones-t; phanis-s; shackleton-s; taylor-m; williams-j
GRUPA16	hyvl-d; mann-k; mims-thurston-p; nemec-g; perlingiere-d; ward-k
GRUPA17	cash-m; dasovich-j; haedicke-m; jones-t; mann-k; sager-e; sanders-r; stclair-c
GRUPA18	cash-m; dasovich-j; haedicke-m; kean-s; sager-e; sanders-r; steffes-j
GRUPA19	corman-s; dasovich-j; hernandez-j; hyatt-k; lokay-m; quigley-d; scott-s
GRUPA20	linder-e; merriss-s; semperger-c; symes-k; williams-w3
GRUPA21	jones-t; mann-k; shackleton-s; stclair-c; taylor-m; ward-k; williams-j
GRUPA22	jones-t; sager-e; shackleton-s; stclair-c; taylor-m
GRUPA23	dasovich-j; gilbertsmith-d; presto-k; sanders-r; shapiro-r; steffes-j
GRUPA24	scholtes-d; semperger-c; symes-k; williams-w3
GRUPA25	hyvl-d; nemec-g; shackleton-s; ward-k; arnold-j; perlingiere-d; williams-j
GRUPA26	blair-l; geaccone-t; hyatt-k; kaminski-v; lokay-m; mcconnell-m; rapp-b; schoolcraft-d; watson-k; ybarbo-p
GRUPA27	delainey-d; kitchen-l; lavorato-j; whalley-g; whalley-l
GRUPA28	nemec-g; staab-t; whitt-m
GRUPA29	mann-k; semperger-c; solberg-g; symes-k; williams-w3
GRUPA30	hyvl-d; jones-t; kitchen-l; sager-e; shackleton-s; stclair-c; taylor-m
GRUPA31	cash-m; haedicke-m; jones-t; kitchen-l; shackleton-s; stclair-c; taylor-m; zipper-a
GRUPA32	delainey-d; kaminski-v; kitchen-l; lavorato-j; whalley-g; whalley-l; zipper-a
GRUPA33	bass-e; beck-s; farmer-d; griffith-j; nemec-g; perlingiere-d; smith-m
GRUPA34	baughman-d; davis-d; griffith-j; kitchen-l; lay-k; rogers-b
GRUPA35	beck-s; buy-r; delainey-d; hayslett-r; kaminski-v; kitchen-l; may-l; mcconnell-m; shankman-j; white-s

8

Algorytm mrowiskowy z zastosowaniem sieci społecznej

Prace nad udoskonaleniem algorytmu poprawiającego trafność przypisywania wiadomości e-mail do folderów doprowadziły do zaprojektowania nowego algorytmu do problemu automatycznego kategoryzowania wiadomości e-mail. Proponowany algorytm oparty jest na metodyce algorytmów mrowiskowych, eksploracji danych oraz sieci społecznych. Inspiracją do jego powstania był algorytm mrowiskowy do konstruowania drzew decyzyjnych (rozdział 2), struktura tabel decyzyjnych (rozdział 4) oraz sieci społeczne (rozdział 3).

Wykonywanie algorytmu polega na budowie sieci społecznej opartej na kontaktach pomiędzy użytkownikami, dokładnej analizie tej sieci oraz wyborze grup użytkowników, a następnie na zastosowaniu algorytmu mrowiskowego do budowy klasyfikatora. Szczegółowe rozwiązania związane z algorytmem do automatycznego kategoryzowania wiadomości e-mail do folderów oraz wyniki eksperymentów zostały opisane w niniejszym rozdziale. Nowy algorytm został zaproponowany również w celu umożliwienia użytkownikom sugerowania tworzenia nowych folderów i umieszczania w nich wiadomości, na podstawie struktury folderów innych użytkowników z grupy kontaktowej.

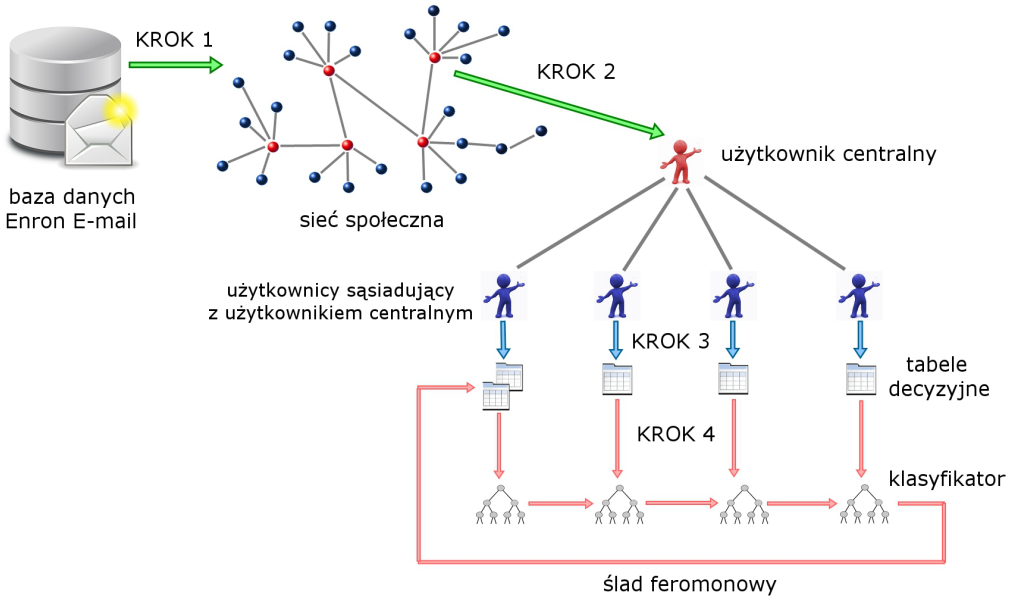
8.1 Idea algorytmu

Pierwszym krokiem w proponowanej metodzie poprawiającej trafność przypisania wiadomości e-mail do folderów jest stworzenie sieci społecznej na podstawie kontaktów pomiędzy nadawcą a odbiorcami wiadomości e-mail ze zbioru Enron (rys. 8.1 – krok 1). Wierzchołkami sieci są wszyscy użytkownicy skrzynek pocztowych ze zbioru Enron e-mail, natomiast krawędzie to połączenia przedstawiające przesyłanie wiadomości e-mail pomiędzy poszczególnymi pracownikami Enron

Corporation.

Następnym krokiem algorytmu jest przeprowadzenie analizy utworzonej sieci społecznej, obliczenie stopni wierzchołków w sieci, zgodnie ze wzorem (3.7) oraz wybranie węzłów sieci o najwyższym stopniu wierzchołka (3.8) (tzw. węzły centralne), które określają kluczowych użytkowników w firmie (rys. 8.1 – krok 2). Dla określonych węzłów centralnych, poprzez zastosowanie współczynnika gronowania opisanego wzorem (3.13), należy dokonać procesu klasteryzacji poprzez wybranie węzłów sąsiadujących z węzłami centralnymi. Tym sposobem tworzone są grupy użytkowników często kontaktujących się ze sobą i posiadające podobną strukturę folderów (alg. 5 – linia 2).

Przeprowadzając analizę struktury folderów w skrzynkach pocztowych użytkowników z danej grupy zastosowano algorytm Levenshteina [54] służący do obliczania podobieństwa łańcuchów tekstowych. Otrzymana w wyniku działania algorytmu liczba (tzw. odległość Levenshteina) określa ile działań należy wykonać, aby dokonać zamiany jednego łańcucha znaków na drugi. Nazwy folderów, dla których odległość Levenshteina była stosunkowo niewielka zostały zmapowane na identyczną nazwę.



Rysunek 8.1: Schemat działania proponowanego algorytmu.

Kolejnym krokiem proponowanego algorytmu było przetworzenie zbioru danych z bazy Enron e-mail do odpowiedniej struktury danych. Tabele decyzyjne powstały osobno dla każdego użytkownika w danej grupie (rys. 8.1 – krok 3, alg. 5 – linia 6). Na podstawie przeprowadzonych badań opisanych w pracy [19], w której zaobserwowano, że informacje wpływające na trafność przypisania wia-

domości do folderów pochodzą z pierwszych pięciu linii wiadomości, przygotowano tabele decyzyjne składające się z siedmiu atrybutów wybranych w taki sposób, aby określały najważniejsze informacje o każdej wiadomości. Opis budowy tabeli decyzyjnej oraz sposób przekształcenia zbioru wiadomości e-mail do tabeli decyzyjnej został szczegółowo przedstawiony w rozdziale 4.5.

Algorytm 5: Pseudokod proponowanego algorytmu

```
1  zbiór_użytkowników_centralnych = analizuj_sieć_społeczną(zbiór_enron_e-mail);
   //wzór(3.8)
2  grupa_użytkowników = SNA_grupy(zbiór_użytkowników_centralnych); //dla danego
   użytkownika //wzór(3.13)
3  //Użytkownik centralny jest pierwszy
4  for osoba=1 to liczba_użytkowników_w_grupie do
5    zbiór_danych[osoba]=przygotuj_tabelę_decyzyjną(osoba)
6  endFor
7  feromon = inicjalizuj_ślad_feromonowy(); //wspólne dla wszystkich użytkowników
8  //Pierwsza i ostatnia iteracja jest dla użytkownika centralnego
9  for osoba=1 to (liczba_użytkowników_w_grupie+1) do
10   for i=1 to (liczba_iteracji / (liczba użytkowników +1)) do
11     najlepszy_klasyfikator = NULL;
12     for j=1 to liczba_mrówek do
13       nowy_klasyfikator = konstruuj_klasyfikator_aACDT(feromon, zbiór_danych[osoba]));
14       //wzór(8.1)
15       oceń_jakość_klasyfikatora(nowy_klasyfikator);
16       if nowy_klasyfikator jest_lepszej_jakości_od najlepszy_klasyfikator then
17         najlepszy_klasyfikator = nowy_klasyfikator;
18       endIf
19     endFor
20     aktualizuj_ślad_feromonowy(najlepszy_klasyfikator, feromon); //wzór(2.16)
21     //Tylko w ostatniej iteracji - dla użytkownika centralnego
22     if osoba == (liczba_użytkowników_w_grupie+1) then
23      if najlepszy_klasyfikator jest_lepszej_jakości_od najlepiej_zbudowany_klasyfik then
24        najlepiej_zbudowany_klasyfik = najlepszy_klasyfikator;
25       endIf
26     endIf
27   endFor
28 endFor
29 wynik = najlepiej_zbudowany_klasyfik;
```

Po uruchomieniu algorytmu mrowiskowego (rys. 8.1 – krok 4) wielokrotnie budowany jest klasyfikator na podstawie danych treningowych (alg. 5 – linia 13, wzór 8.1). Każdy klasyfikator jest testowany (alg. 5 – linia 15) i w zależności od otrzymanych wyników odkładany jest feromon (alg. 5 – linia 20). Klasyfikator dla każdego użytkownika budowany jest wraz z analizą sieci komunikacji przypisanej do niego grupy (alg. 5 – linia 13).

Poprzez algorytm budowany jest klasyfikator dla wybranego użytkownika, następnie z zastosowaniem tej samej macierzy śladu feromonowego budowany jest klasyfikator dla każdej kolejnej osoby z grupy. Po zbudowaniu klasyfikatorów i

jednocześnie ustabilizowaniu się macierzy śladu feromonowego następuje budowa ostatecznego klasyfikatora dla wybranego użytkownika, zgodnie z topologią pierścienia (alg. 5 – linie 21-24). Takie zastosowanie pozwala na zachowanie (poprzez ślad feromonowy) informacji związanych z decyzjami pozostałych osób w grupie, co wpływa na budowanie klasyfikatora dla właściwego użytkownika.

W związku z analizą sieci społecznej modyfikacji ulega także funkcja heurystyczna algorytmu, a jej wartość, wyznaczana na podstawie wzoru (2.10), jest określona jako:

$$\underset{a_j \leqslant a_j^R, j=1,\ldots,M}{\arg\max} \left(\sum_{g=1}^{G} \left(\frac{P_l P_r}{4} \left[\sum_{k=1}^{K} |p(k|m_l) - p(k|m_r)| \right]^2 \right) \right), \tag{8.1}$$

gdzie:

g to numer osoby z listy adresatów,

G to liczba adresatów z grupy użytkowników.

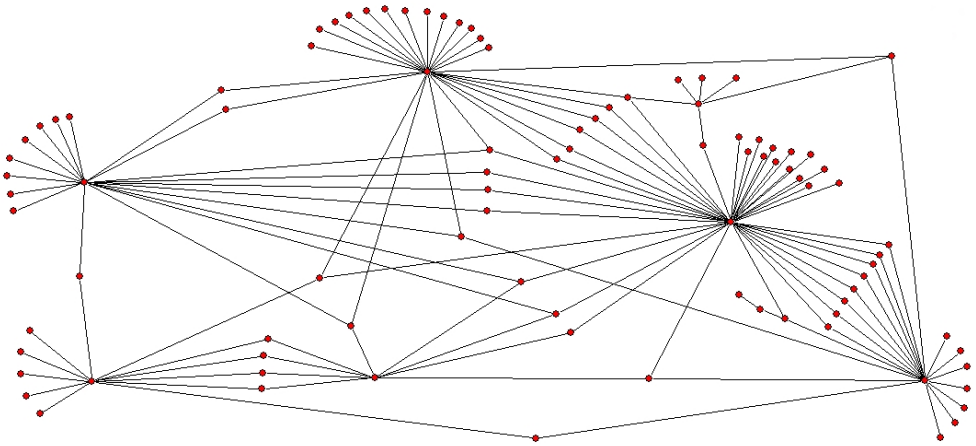
Informacja o odłożonym śladzie feromonowym pełni rolę sprzężenia zwrotnego podczas budowania klasyfikatora dla danego użytkownika, co ma wpływ na podejmowaną decyzję dotyczącą wyboru folderu. Po skończeniu pracy algorytmu otrzymywany jest najlepszy klasyfikator (alg. 5 – linia 29), a następnie jest weryfikowany na podstawie nowych danych, nieużywanych podczas pracy algorytmu. Sposób działania proponowanego algorytmu został przedstawiony w postaci pseudokodu (alg. 5).

8.2 Przeprowadzone badania

W celu przeprowadzenia eksperymentów z zastosowaniem proponowanej metody dokonano analizy całego zbioru Enron e-mail. Na tej podstawie stworzono sieć społeczną pracowników Enron Corporation, która została przedstawiona na rys. 8.2.

Wierzchołkami sieci są wszyscy użytkownicy skrzynek pocztowych ze zbioru Enron e-mail, natomiast krawędzie to połączenia, dla których częstotliwość interakcji, czyli liczba przesłanych wiadomości pomiędzy poszczególnymi osobami, wynosiła więcej niż 10. Wierzchołki sieci, które nie są powiązane z żadnymi innymi wierzchołkami to osoby, które mają mniej niż założona liczba przesłanych wiadomości e-mail.

Następnie podczas analizy otrzymanej sieci społecznej wybrano węzły sieci o najwyższym stopniu wierzchołka, które razem z najbliższymi sąsiadami utworzyły poszczególne grupy użytkowników w sieci. W celu sprawdzenia działania proponowanego algorytmu oraz porównania otrzymanych wyników z poprzednimi badaniami wybrano dziewięć grup użytkowników, które zostały przedstawione w tab. 8.1. Natomiast parametry wybranych grup użytkowników oraz parametry użytkownika kluczowego w tych grupach są widoczne w tab. 8.2 i w tab. 8.3.



Rysunek 8.2: Wizualizacja sieci dla proponowanego algorytmu.

Tabela 8.1: Wybrane grupy użytkowników.

Nazwa grupy	Kluczowy użytkownik	Najbliżsi sąsiedzi kluczowego użytkownika
Grupa 1	lokay-m	hyatt-k, mcconnell-m, schoolcraft-d, scott-s, watson-k
Grupa 2	sanders-r	cash-m, dasovich-j, haedicke-m, kean-s, sager-e, steffes-j
Grupa 3	shackleton-s	jones-t, mann-k, stclair-c, taylor-m, ward-k, williams-j
Grupa 4	steffes-j	dasovich-j, gilbertsmith-d, presto-k, sanders-r, shapiro-r
Grupa 5	symes-k	scholtes-d, semperger-c, williams-w3
Grupa 6	williams-w3	mann-k, semperger-c, solberg-g, symes-k
Grupa 7	farmer-d	bass-e, beck-s, griffith-j, nemec-g, perlingiere-d, smith-m
Grupa 8	beck-s	buy-r, delainey-d, hayslett-r, kaminski-v, kitchen-l, may-l, mcconnell-m, shankman-j, white-s
Grupa 9	rogers-b	baughman-d, davis-d, griffith-j, kitchen-l, lay-k

Proponowany algorytm został zaimplementowany w języku C++. Wszystkie obliczenia wykonano na komputerze z procesorem Intel Core i5 2.27 GHz z 2.9 GB RAM. Komputer działał pod kontrolą systemu operacyjnego Debian GNU/Linux. Budowa sieci społecznej oraz wybór grup użytkowników na podstawie analizy sieci społecznej zostały przeprowadzone w sposób deterministyczny, co opisano w rozdziale 8.1. Natomiast dla algorytmu mrowiskowego wykonano doświadczenia, które zostały powtórzone 30 razy dla każdego ze zbiorów danych. Każde doświadczenie wykonane zostało przy 250 pokoleniach liczących 25 agentów-mrówek. Parametry pracy algorytmu były następujące: $q_0 = 0,3$; $\alpha = 3$; $\gamma = 0,1$.

Tabela 8.2: Parametry wybranych grup użytkowników.

Nazwa grupy	Kluczowy użytkownik w grupie	Liczba klas (folderów) w grupie	Liczba obiektów (e-mail) w grupie
Grupa 1	lokay-m	87	4963
Grupa 2	sanders-r	230	8024
Grupa 3	shackleton-s	126	4131
Grupa 4	steffes-j	154	4247
Grupa 5	symes-k	35	3598
Grupa 6	williams-w3	58	5138
Grupa 7	farmer-d	156	5998
Grupa 8	beck-s	185	6522
Grupa 9	rogers-b	81	2963

Tabela 8.3: Parametry użytkownika kluczowego wybranych grup użytkowników.

Nazwa grupy	Kluczowy użytkownik w grupie	Liczba klas dla użytkownika kluczowego	Liczba obiektów dla użytkownika kluczowego
Grupa 1	lokay-m	11	2493
Grupa 2	sanders-r	29	1181
Grupa 3	shackleton-s	39	886
Grupa 4	steffes-j	21	617
Grupa 5	symes-k	11	767
Grupa 6	williams-w3	18	2767
Grupa 7	farmer-d	24	3538
Grupa 8	beck-s	84	1703
Grupa 9	rogers-b	14	1395

Kolejnym krokiem autorskiego algorytmu jest przetworzenie zbioru danych z bazy Enron e-mail do postaci tabeli decyzyjnej osobno dla każdej skrzynki pocztowej w danej grupie. Przygotowana tabela decyzyjna składa się z sześciu atrybutów warunkowych oraz jednego atrybutu decyzyjnego *category*, który określa do jakiego folderu przypisana zostaje wiadomość.

Atrybuty warunkowe wybrano w taki sposób, aby określały najważniejsze informacje o każdej wiadomości. Składają się z nadawcy, trzech pierwszych słów z tematu wiadomości e-mail, informacji w postaci wartości boolowskiej, czy osoba, która otrzymała wiadomość była dodana do kopii wiadomości e-mail (jeśli nie, to znaczy, że była adresatem) oraz długości wiadomości e-mail. Ponadto z tematu maila pominięto podstawowe zwroty oraz łączniki, natomiast dodatkowo wspierano słowa, która należały do zbioru klas decyzyjnych.

Po utworzeniu grup użytkowników budowane są kolejno klasyfikatory dla osoby, do której przypisana jest grupa (czyli dla użytkownika kluczowego w grupie),

następnie dla kolejnych osób z danej grupy i na końcu znowu dla użytkownika kluczowego. Ślad feromonowy jest ten sam przez cały czas trwania algorytmu, co pozwala na zachowanie informacji o podjętych decyzjach poprzednich osób w grupie, a także po zmodyfikowaniu algorytmu umożliwia sugerowanie zakładania nowych folderów.

Uzyskane wyniki przedstawione w tab. 8.4 wskazują na znaczną poprawę w przypadku zastosowania autorskiego algorytmu. Wyniki dla pozostałych algorytmów opisane są w rozdziale 6.

Tabela 8.4: Porównanie podejść opartych na algorytmach mrowiskowych z autorskim algorytmem.

Skrzynka pocztowa	aADCT	ACDF	proponowany algorytm
beck-s	0,583	0,517	**0,600**
farmer-d	0,811	0,775	**0,834**
lokay-m	0,888	0,846	**0,891**
sanders-r	0,829	0,759	**0,871**
williams-w3	**0,962**	0,944	0,960
shackleton-s	0,709	0,698	**0,751**
steffes-j	0,841	0,824	**0,863**
symes-k	0,930	0,916	**0,937**
rogers-b	**0,911**	0,898	0,900

Dla dziewięciu zbiorów danych stworzonych na podstawie dziewięciu grup użytkowników proponowany algorytm za każdym razem uzyskuje lepsze rezultaty w porównaniu do algorytmów klasycznych. Dla niektórych zbiorów (sanders-r, symes-k) trafność przypisania wiadomości do folderów wzrosła nawet o 15%.

Dla dwóch zbiorów (rogers-b, williams-w3) trafność przypisania wiadomości do folderów jest na bardzo wysokim poziomie (93-96%), jednak zastosowanie analizy sieci społecznej nie wpłynęło na poprawę otrzymanych rezultatów. Dodatkowo porównując wyniki proponowanego algorytmu z algorytmem opisanym w rozdziale 6 można zauważyć, że poprawa dokładności przypisania folderu do e-mail wynosi 1-3% (dla beck-s, farmer-d, steffes-j, symes-k) natomiast w przypadku zbiorów sanders-r, shackleton-s trafność przypisania wiadomości do folderów jest lepsza nawet o 5%, co potwierdza słuszność zastosowania sieci społecznych i analizy skrzynek pocztowych dla grupy użytkowników, zamiast pojedynczych osób.

W tab. 8.4 przedstawiono porównanie najlepszych wyników ze wszystkich przeprowadzonych eksperymentów dla problemu automatycznej kategoryzacji wiadomości e-mail do folderów. Dla dwóch zbiorów (rogers-b, williams-w3) trafność przypisania wiadomości do folderów jest na bardzo wysokim poziomie (93-96%), jednak zastosowanie analizy sieci społecznej nie wpłynęło na poprawę otrzymanych rezultatów. Natomiast dla pozostałych siedmiu zbiorów danych proponowana metoda za każdym razem uzyskuje lepsze rezultaty, a trafność przypisania

wiadomości do folderów wynosi nawet 5% w porównaniu do algorytmu mrowiskowego, co potwierdza słuszność zastosowania sieci społecznych i analizy skrzynek pocztowych dla grupy użytkowników, zamiast pojedynczych osób. Dodatkowo zastosowanie macierzy śladu feromonowego z zachowaniem folderów wszystkich osób z grupy umożliwia użytkownikom sugerowanie zakładania nowych folderów, a wyniki badań wskazują na pozytywne efekty działania takiego podejścia.

Tabela 8.5: Wyniki testu Friedmana i średnie rangi dla danych z tab. 8.4

	Wartość
N	9
Chi-Kwadrat	14,88889
Liczba stopni swobody	2
Wartość p jest mniejsza niż	0,0006
5% krytyczna różnica	0,440684
Średnie rangi	
aACDT	1,77778
ACDF	3
proponowany algorytm	**1,22222**

Tabela 8.6: Statystyczne różnice pomiędzy algorytmami dla danych z tab. 8.4

	aACDT	ACDF	Proponowany algorytm
aACDT	–	**1,22222**	**-0,55556**
ACDF	**-1,22222**	–	**-1,77778**
Proponowany algorytm	**0,55556**	**1,77778**	–

Parametry Testu Friedmana zostały przedstawione w tab. 8.5 i są następujące: $Chi - Kwadrat = 14,88889$, *Liczba stopni swobody* $= 2$, *Wartość p jest mniejsza niż* $0,0001$, 5% *krytyczna różnica* $= 0,440684$. Na podstawie testu uzyskano ranking, który wykazał, że najlepsza predykcja osiągana jest w przypadku autorskiego algorytmu (średnia wartość rangi = 1,22222), następnie jest algorytm aACDT (1,77778), a na końcu algorytm ACDF (3,0). W tabeli 8.6 przedstawiono statystyczne porównanie omówionych metod jako różnice w rankingu między porównywanymi algorytmami. Czcionki pogrubione wskazują wartości, które spełniają kryterium 5% krytycznej różnicy dla przeprowadzonego testu. Wartości poniżej zera wskazują, że analizowany algorytm (wiersz w tabeli) jest gorszy niż algorytm porównany (kolumna w tabeli). Jak można jednocześnie zauważyć, dla najlepszej metody każdorazowo występuje krytyczna różnica w stosunku do pozostałych podejść, co wskazuje na to, że jest ona znacznie lepsza w przypadku przeprowadzenia predykcji od innych analizowanych algorytmów.

9

Mechanizm sugerowania zakładania nowych folderów

Ostatnim celem w niniejszej rozprawie jest przedstawienie mechanizmu, który umożliwia sugerowanie zakładania nowych folderów dla użytkowników, na podstawie struktury folderów innych użytkowników wyznaczonych przez stworzoną sieć społeczną. Proponowana metoda oparta jest na analizie macierzy śladu feromonowego tworzonej podczas klasyfikowania wiadomości do folderów.

Same mechanizmy sugerowania przypisywania wiadomości do nowych folderów nie są obecnie nowością, gdyż są w praktyce stosowane w niektórych systemach pocztowych. Jednak należy zwrócić uwagę przede wszystkim na ich zawężoną tematykę i sposób działania. Nowe foldery dotyczą w szczególności wiadomości generowanych automatycznie lub rozpoznawanych za pomocą programu pocztowego, jako wiadomości związanych z forami dyskusyjnymi, ofertami handlowymi czy serwisami społecznościowymi. Nie sposób jednak znaleźć algorytmy, przy pomocy których możliwe byłoby sugerowanie bardziej nietypowych folderów dla wiadomości, które nie są generowane automatycznie. Autorska metoda nie tylko związana jest z sugestią nowych folderów, ale dodatkowo bazuje na możliwościach, jakie dają algorytmy mrowiskowe oraz sieci społeczne.

Zgodnie z zaproponowaną metodą opisaną w rozdziale 8, algorytm do automatycznego przypisywania wiadomości do folderów wraz z mechanizmem sugerującym użytkownikom tworzenie nowych folderów w swoich skrzynkach pocztowych polega na:

- przeprowadzeniu analizy dotychczas odebranych wiadomości e-mail pod względem kontaktów użytkowników;
- stworzeniu sieci społecznej opartej na kontaktach pomiędzy nadawcą a odbiorcami wiadomości;

- wyodrębnieniu grupy użytkowników posiadających podobną strukturę społeczną, na podstawie analizy i obserwacji sieci społecznej;
- przetworzeniu zbioru danych do postaci tabeli decyzyjnej w obrębie danej grupy;
- zastosowaniu algorytmu opartego na rozwiązaniach znanych z algorytmów mrowiskowych;
- przedstawieniu mechanizmu predykcji folderów dla użytkowników, na podstawie analizy macierzy klasyfikacji wiadomości do folderów.

Zasadniczym aspektem jest w tym przypadku wyodrębnienie grupy kontaktów dla użytkownika, któremu mają zostać zasugerowane nowe foldery. W tym celu, zgodnie z utworzoną siecią kontaktów, należy ustalić najbliższych sąsiadów tego użytkownika (traktowanego jako użytkownika centralnego), a następnie na podstawie preferencji tych użytkowników dokonać sugestii stworzenia nowych folderów.

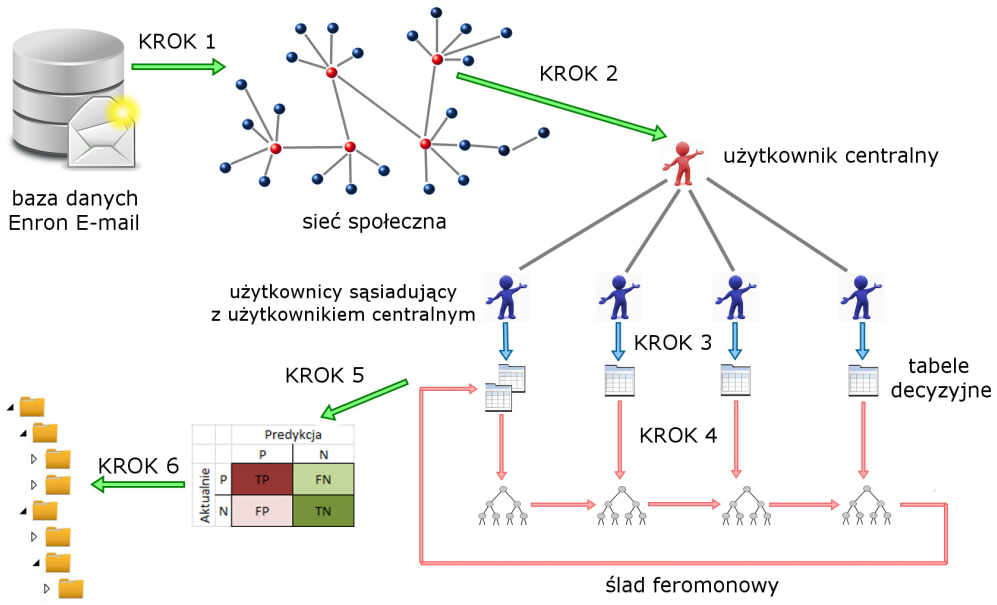
Główna idea rozwiązania bazuje na analizie wspólnej macierzy śladu feromonowego dla wszystkich użytkowników w grupie. W klasycznej wersji proponowanego algorytmu (opisanej w rozdziale 8), pomimo zastosowania grupy użytkowników, jako dostępne wartości atrybutu decyzyjnego dopuszczalne są jedynie te, które pierwotnie występują u użytkownika, dla którego wykonywana jest predykcja. Wiąże się to m.in. z tym, że wszystkie wiadomości, które pozostali użytkownicy przechowują we własnych, unikalnych względem użytkownika centralnego, folderach zostają pominięte. W tym przypadku dopuszczalne wartości atrybutu decyzyjnego są sumą wartości atrybutów decyzyjnych wszystkich użytkowników w grupie (nie tylko centralnego), zgodnie ze wzorem:

$$D = D_1 \cup D_2 \cup \ldots \cup D_n, \tag{9.1}$$

gdzie:
D_i to zbiór wartości atrybutów decyzyjnych $i - tego$ użytkownika,
n to liczba użytkowników w grupie.

W znacznym uproszczeniu można stwierdzić, że jeśli wiadomość o zbliżonych cechach (atrybutach) pozostali użytkownicy w grupie będą przechowywali w folderze, którego użytkownik centralny nie ma, to zostanie mu zasugerowane utworzenie nowego folderu. Jak można zauważyć, w tym przypadku duże znaczenie ma wstępne przetworzenie danych i dostosowanie nazw folderów do zbliżonych, aby różnice wynikające np. z zapisu nazwy folderu nie sugerowały różnicy pomiędzy folderami.

Schemat działania proponowanego algorytmu z mechanizmem predykcji folderów został przedstawiony na rys. 9.1. Na podstawie opisanego w rozdziale 8.1 autorskiego algorytmu, po przejściu przez przez kroki 1–4 otrzymywany jest najlepiej zbudowany klasyfikator, którego działanie weryfikowane jest na podstawie danych testowych. Podczas pracy algorytmu tworzona jest macierz śladu feromonowego (rys. 9.1 – krok 5), której analiza pozwala na zasugerowanie użytkownikowi utworzenie nowych folderów (rys. 9.1 – krok 6).



Rysunek 9.1: Schemat działania proponowanego algorytmu z mechanizmem predykcji folderów

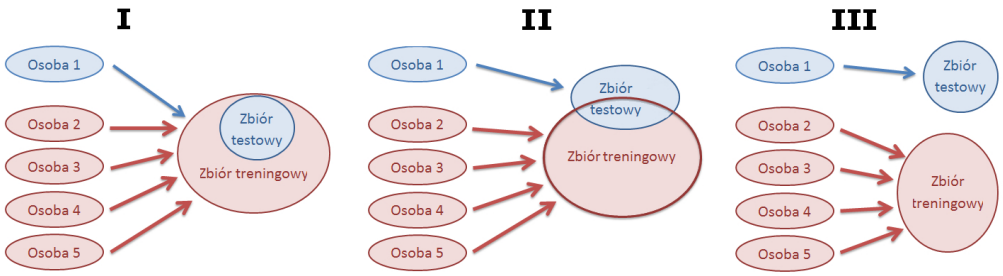
W wyniku pracy tak zaprojektowanego algorytmu, a także pewnej formy pamięci (decyzji innych użytkowników) poprzez ślad feromonowy możliwe jest określenie wagi nowej sugestii. W tym celu zaproponowana została macierz śladu feromonowego będąca analogią do klasycznej macierzy błędu, która w tym przypadku ma na celu zobrazowanie powstałego rozwiązania, a nie określenia błędów klasyfikacji.

Macierz błędów (ang. Confusion matrix) to narzędzie stosowane do oceny jakości modeli klasyfikacyjnych, które przedstawia zależność dokładności klasyfikacji każdej z klas oraz błędów wskazujących obiekty zaklasyfikowane do innej klasy. Wiersze w takiej macierzy odpowiadają poprawnym klasom decyzyjnym, a kolumny decyzjom przewidywanym przez klasyfikator. Dokładność klasyfikacji poszczególnych klas odczytujemy na podstawie przecięcia wierszy z kolumnami. Definicja macierzy błędów przedstawiona jest w tab. 9.1.

Tabela 9.1: Definicja macierzy błędów

		Klasa przewidywana	
		Pozytywna	Negatywna
Klasa rzeczywista	Pozytywna	Prawdziwie pozytywna, TP	Fałszywie negatywna, FN
	Negatywna	Fałszywie pozytywna, FP	Prawdziwie negatywna, TN

Przeprowadzone badania zostały podzielone na trzy oddzielne etapy ze względu na podział danych na zbiór treningowy i testowy. W każdym z trzech etapów zbiór treningowy stanowią skrzynki pocztowe osób sąsiadujących z użytkownikiem centralnym (kluczowym), natomiast zbiór testowy to skrzynka pocztowa użytkownika centralnego. W zależności od etapu badań zbiór testowy może w całości zawierać się w zbiorze treningowym (etap I), jest w 50% zawarty w zbiorze treningowym (etap II) lub stanowi zupełnie nowe dane dla klasyfikatora, gdyż nie ma wspólnych elementów w stosunku do zbioru treningowego (etap III). Charakterystyka podziału zbiorów została także przedstawiona na rys. 9.2.

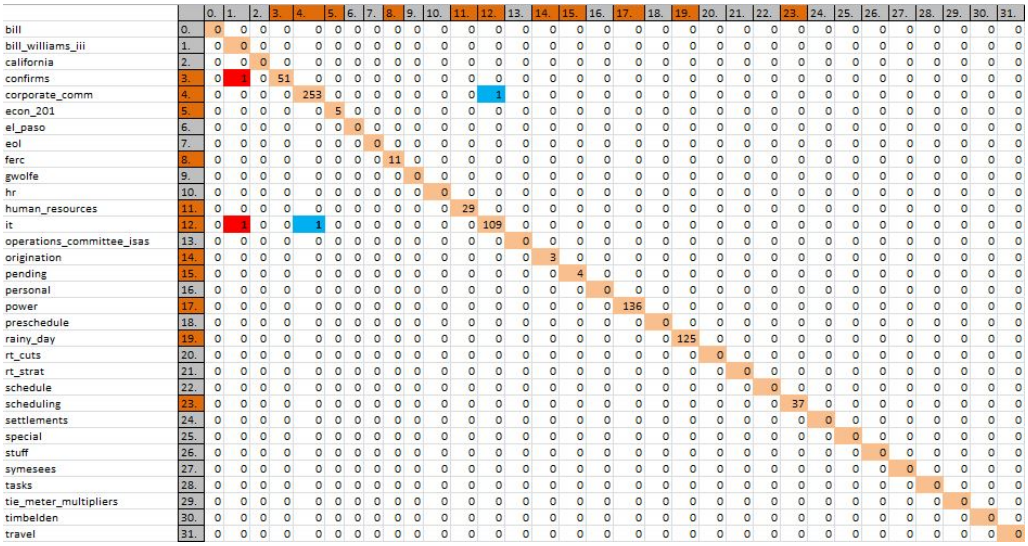


Rysunek 9.2: Rodzaje podziału danych na zbiory treningowe i testowe

Dla dziewięciu użytkowników kluczowych grup z tab. 8.1, utworzonych na podstawie sieci społecznej opartej na kontaktach pracowników firmy Enron Corporation stworzono i przeanalizowano macierze śladu feromonowego odkładanego podczas klasyfikowania wiadomości do folderów.

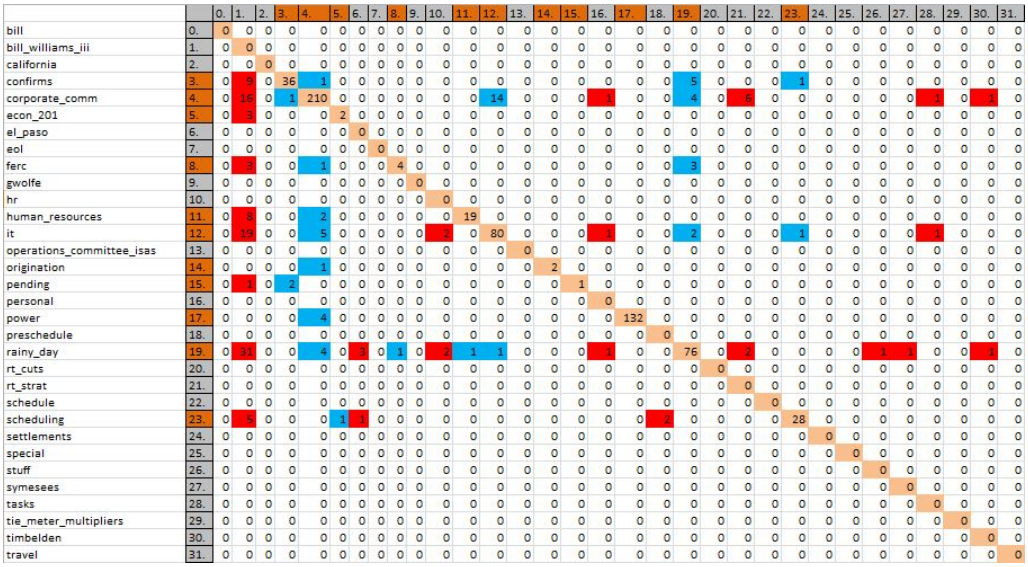
Dla każdego użytkownika stworzono trzy macierze śladu feromonowego, zgodnie z trzema etapami badań. Na rys. 9.3 przedstawiono macierz dla I etapu dla użytkownika $symes - k$. Natomiast rysunki 9.4 i 9.5 przedstawiają macierze śladu feromonowego odpowiednio dla II i III etapu dla tego samego użytkownika.

W przedstawionych macierzach (na rysunkach 9.3 – 9.5), na przekątnej wykazana jest liczba wiadomości prawidłowo przypisanych do folderów. Na niebiesko zaznaczone są wiadomości błędnie przypisane do folderów, jednak są to foldery, które występują w skrzynce pocztowej użytkownika. Natomiast na czerwono



	0.	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	13.	14.	15.	16.	17.	18.	19.	20.	21.	22.	23.	24.	25.	26.	27.	28.	29.	30.	31.
bill	0.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
bill_williams_iii	1.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
california	2.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
confirms	3.	0	1	0	51	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
corporate_comm	4.	0	0	0	0	253	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
econ_201	5.	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
el_paso	6.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
eol	7.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ferc	8.	0	0	0	0	0	0	0	11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
gwolfe	9.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
hr	10.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
human_resources	11.	0	0	0	0	0	0	0	0	0	0	29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
it	12.	0	1	0	0	1	0	0	0	0	0	0	109	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
operations_committee_isas	13.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
origination	14.	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
pending	15.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
personal	16.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
power	17.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	136	0	0	0	0	0	0	0	0	0	0	0	0	0	0
preschedule	18.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rainy_day	19.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	125	0	0	0	0	0	0	0	0	0	0	0	0
rt_cuts	20.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rt_strat	21.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
schedule	22.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
scheduling	23.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
settlements	24.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
special	25.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
stuff	26.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
symesees	27.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
tasks	28.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
tie_meter_multipliers	29.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
timbelden	30.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
travel	31.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Rysunek 9.3: Macierz śladu feromonowego dla symes-k - etap I

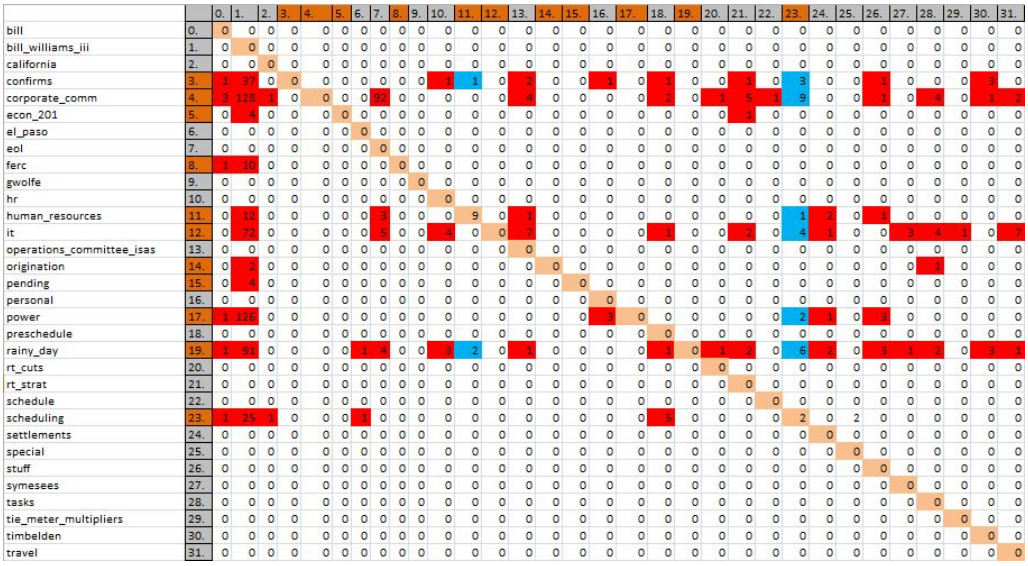


	0.	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	13.	14.	15.	16.	17.	18.	19.	20.	21.	22.	23.	24.	25.	26.	27.	28.	29.	30.	31.
bill	0.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
bill_williams_iii	1.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
california	2.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
confirms	3.	0	9	0	36	1	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	1	0	0	0	0	0	0	0	0
corporate_comm	4.	0	16	0	1	210	0	0	0	0	0	0	14	0	0	0	1	0	0	4	0	6	0	0	0	0	0	0	1	0	1	0
econ_201	5.	0	3	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
el_paso	6.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
eol	7.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ferc	8.	0	3	0	0	1	0	0	0	4	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0
gwolfe	9.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
hr	10.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
human_resources	11.	0	8	0	0	2	0	0	0	0	0	19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
it	12.	0	19	0	0	5	0	0	0	0	0	2	0	80	0	0	0	1	0	0	2	0	0	0	1	0	0	0	0	1	0	0
operations_committee_isas	13.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
origination	14.	0	0	0	0	1	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
pending	15.	0	1	0	2	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
personal	16.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
power	17.	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	132	0	0	0	0	0	0	0	0	0	0	0	0	0	0
preschedule	18.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rainy_day	19.	0	31	0	0	4	0	3	0	1	0	2	1	1	0	0	0	1	0	0	76	0	2	0	0	0	0	1	1	0	0	1	0
rt_cuts	20.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rt_strat	21.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
schedule	22.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
scheduling	23.	0	5	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	28	0	0	0	0	0	0	0	0
settlements	24.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
special	25.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
stuff	26.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
symesees	27.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
tasks	28.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
tie_meter_multipliers	29.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
timbelden	30.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
travel	31.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Rysunek 9.4: Macierz śladu feromonowego dla symes-k - etap II

zaznaczono wiadomości przypisane do folderów, które wcześniej nie występowały w skrzynce pocztowej użytkownika kluczowego. Informacje o tych folderach są pobierane ze struktur folderów w skrzynkach użytkowników będących najbliższymi sąsiadami kluczowego użytkownika, którzy razem tworzą grupy użytkowników.

Na podstawie przeprowadzonej analizy macierzy śladu feromonowego odkładanego podczas pracy klasyfikatora, otrzymano wyniki przypisania wiadomości



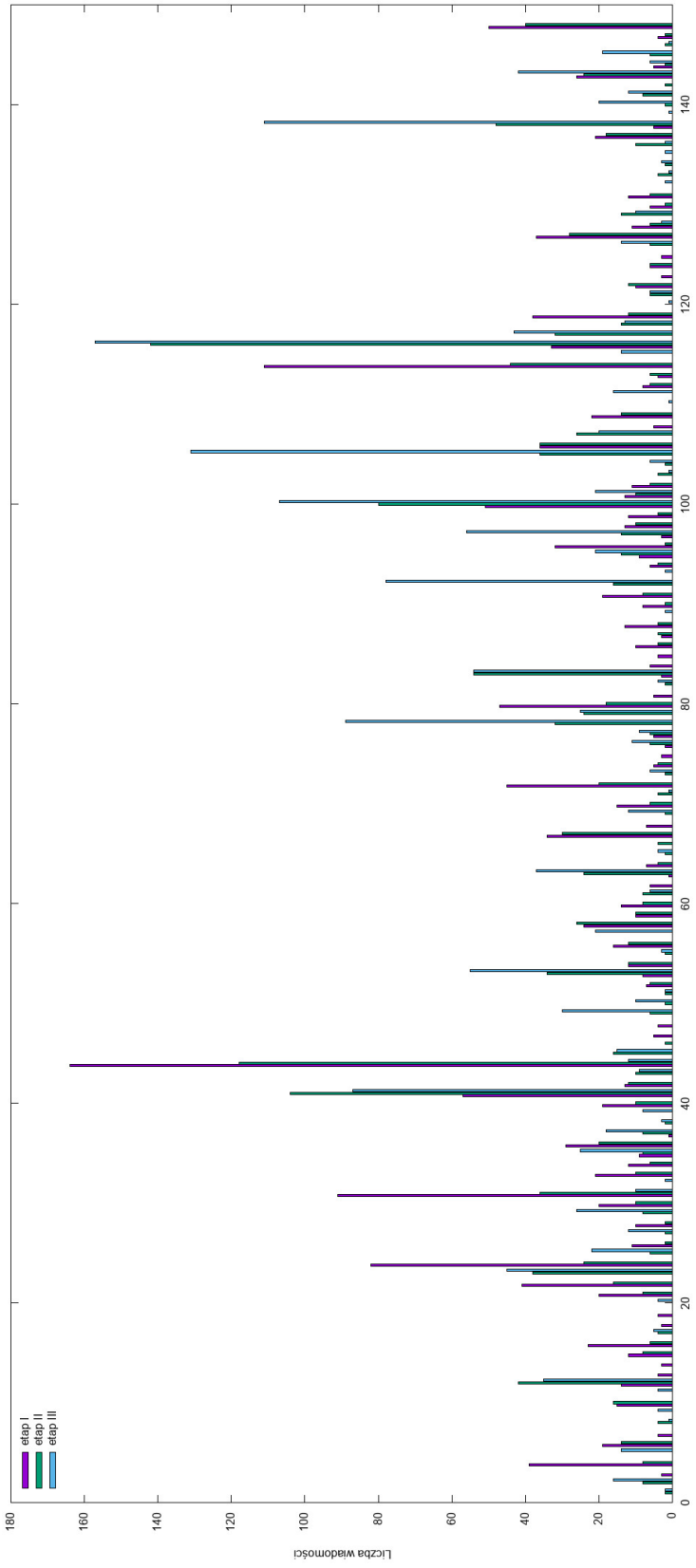
		0.	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	13.	14.	15.	16.	17.	18.	19.	20.	21.	22.	23.	24.	25.	26.	27.	28.	29.	30.	31.
bill	0.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
bill_williams_iii	1.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
california	2.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
confirms	3.	1	37	0	0	0	0	0	0	0	0	1	1	0	3	0	0	1	0	1	0	0	1	0	3	0	0	1	0	0	0	3	0
corporate_comm	4.	3	128	1	0	0	0	0	32	0	0	0	0	0	4	0	0	0	0	2	0	1	5	1	9	0	0	1	0	1	0	1	2
econ_201	5.	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
el_paso	6.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
eol	7.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ferc	8.	1	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
gwolfe	9.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
hr	10.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
human_resources	11.	0	12	0	0	0	0	0	3	0	0	0	9	0	1	0	0	0	0	0	0	0	0	0	1	2	0	1	0	0	0	0	0
it	12.	0	72	0	0	0	0	0	5	0	0	4	0	0	7	0	0	0	0	1	0	0	2	0	4	1	0	0	3	4	1	0	7
operations_committee_isas	13.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
origination	14.	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
pending	15.	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
personal	16.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
power	17.	1	126	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	2	1	0	3	0	0	0	0	0
preschedule	18.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rainy_day	19.	1	91	0	0	0	0	1	4	0	0	1	2	0	1	0	0	0	0	1	0	1	2	0	6	2	0	3	1	2	0	3	1
rt_cuts	20.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rt_strat	21.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
schedule	22.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
scheduling	23.	1	25	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	2	0	2	0	0	0	0	0	0
settlements	24.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
special	25.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
stuff	26.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
symesees	27.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
tasks	28.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
tie_meter_multipliers	29.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
timbelden	30.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
travel	31.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Rysunek 9.5: Macierz śladu feromonowego dla symes-k - etap III

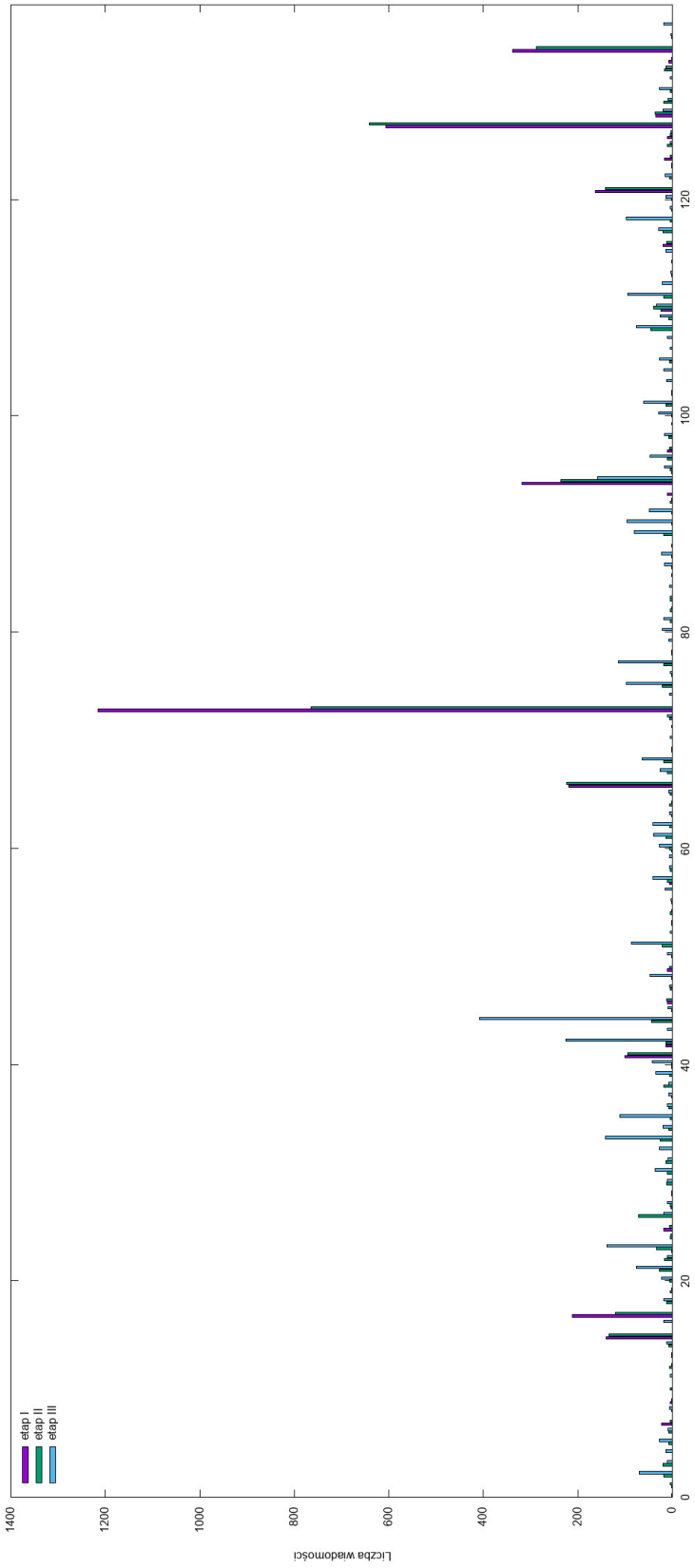
do folderów dla wszystkich użytkowników kluczowych z tab. 8.1 wybranych zgodnie z algorytmem opisanym w rozdziale 8.1. Foldery, do których zostały przypisane wiadomości ze skrzynki użytkownika kluczowego, to wszystkie foldery, które zawierały się w skrzynkach pocztowych wszystkich użytkowników w grupie zgodnie ze wzorem 9.1, bez względu na występowanie tych folderów w skrzynce pocztowej użytkownika kluczowego.

Z otrzymanych wyników odrzucone zostały foldery, do których nie została przypisana żadna wiadomość, a same wyniki w postaci wykresów zostały przedstawione na rysunkach 9.6 – 9.13 odpowiednio dla każdego użytkownika kluczowego. Sugerowane foldery, które powinny zostać utworzone dla danego użytkownika, to te, do których w II lub III etapie zostało sklasyfikowanych wiele wiadomości, natomiast nie były one przypisane do tych folderów podczas etapu I.

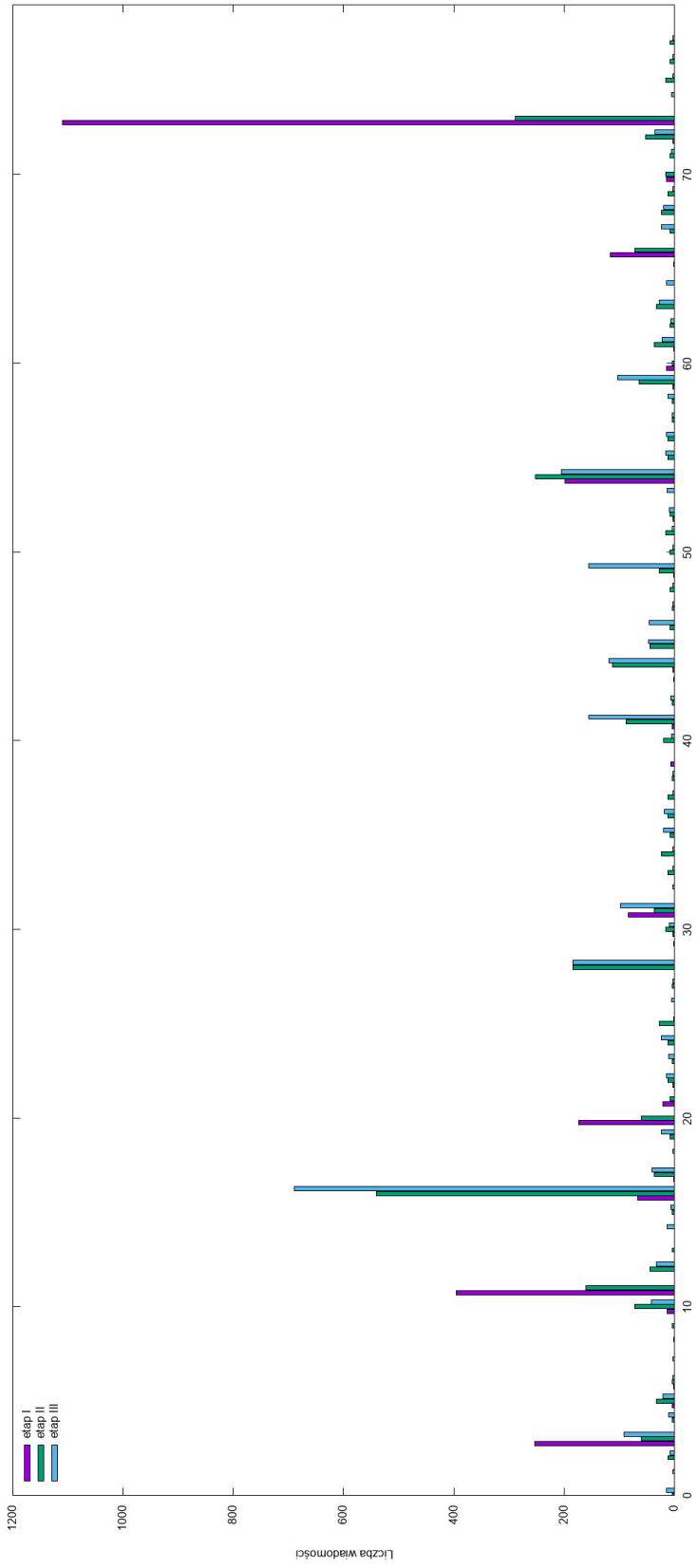
Otrzymane wyniki zależą nie tylko od dużej częstotliwości kontaktów pomiędzy poszczególnymi osobami, ale przede wszystkim od subiektywnie utworzonych struktur folderów innych osób. Jednocześnie stworzone macierze pozwalają na zaobserwowanie rzeczywistego rozwiązania - często bardzo duża liczba wiadomości przypisanych do folderów utworzonych przez innych użytkowników (dla przypadków zaznaczonych na czerwono w macierzy) w stosunku do liczby pozostałych wiadomości oznacza, że proponowana sugestia utworzenia folderu posiada duże wsparcie w przypadku grupy użytkowników. Natomiast im mniejsza wartość, tym słabsze wsparcie sugestii.



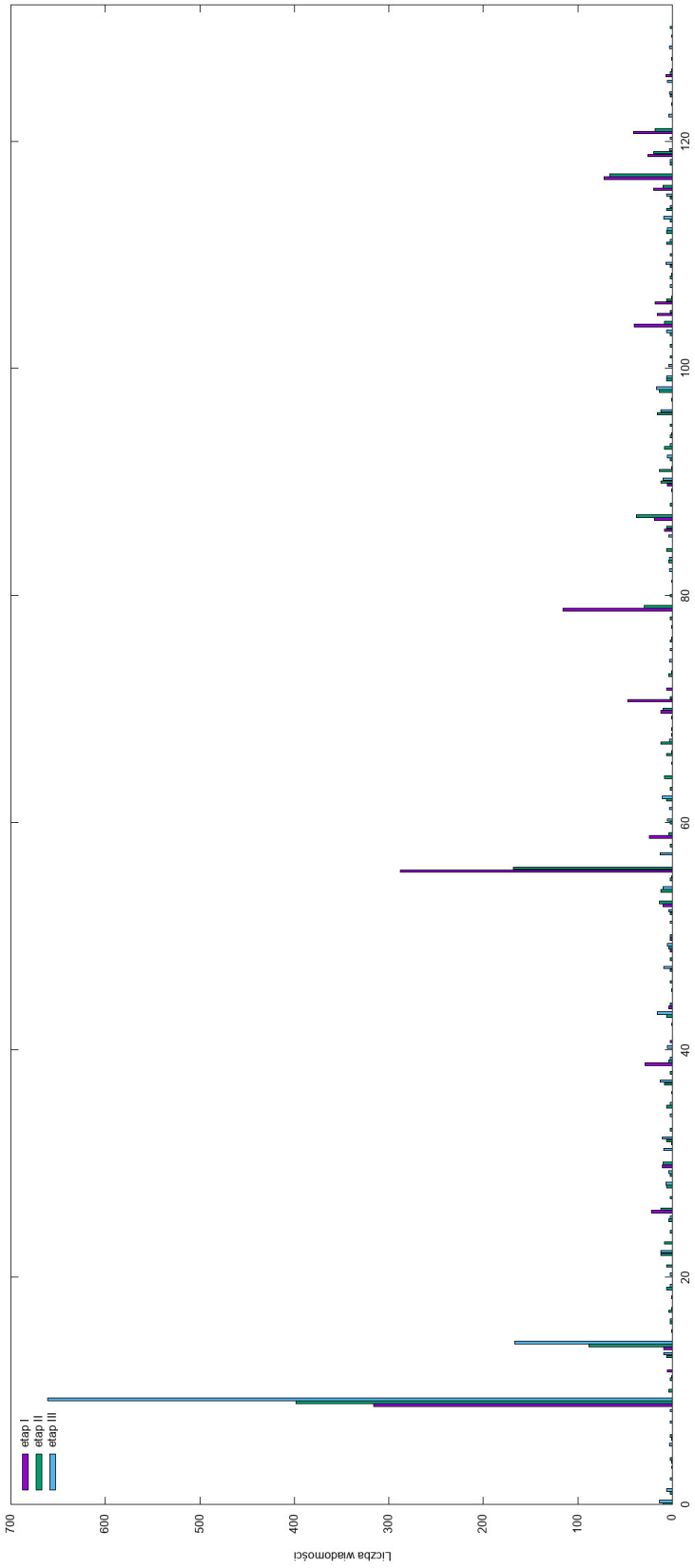
Rysunek 9.6: Przypisanie wiadomości do folderów dla skrzynki beck-s



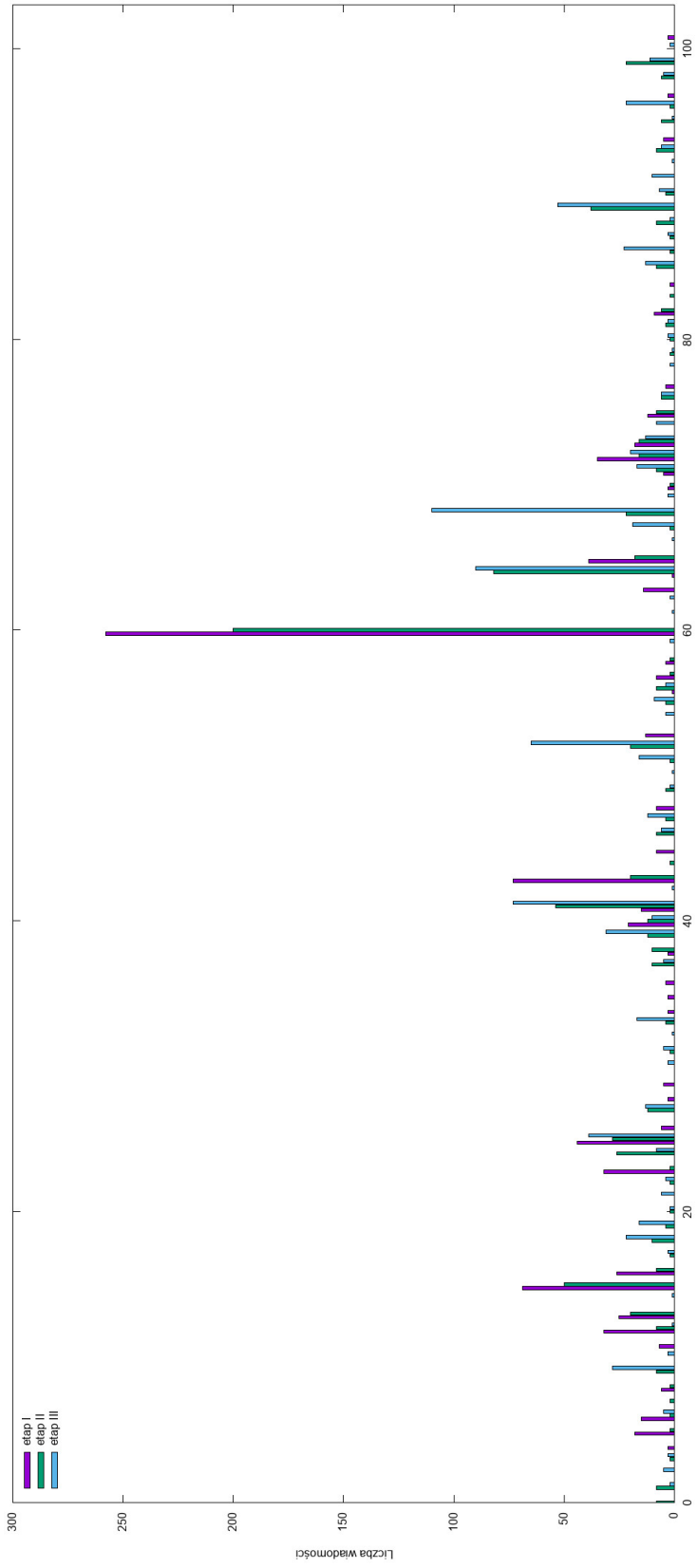
Rysunek 9.7: Przypisanie wiadomości do folderów dla skrzynki farmer-d



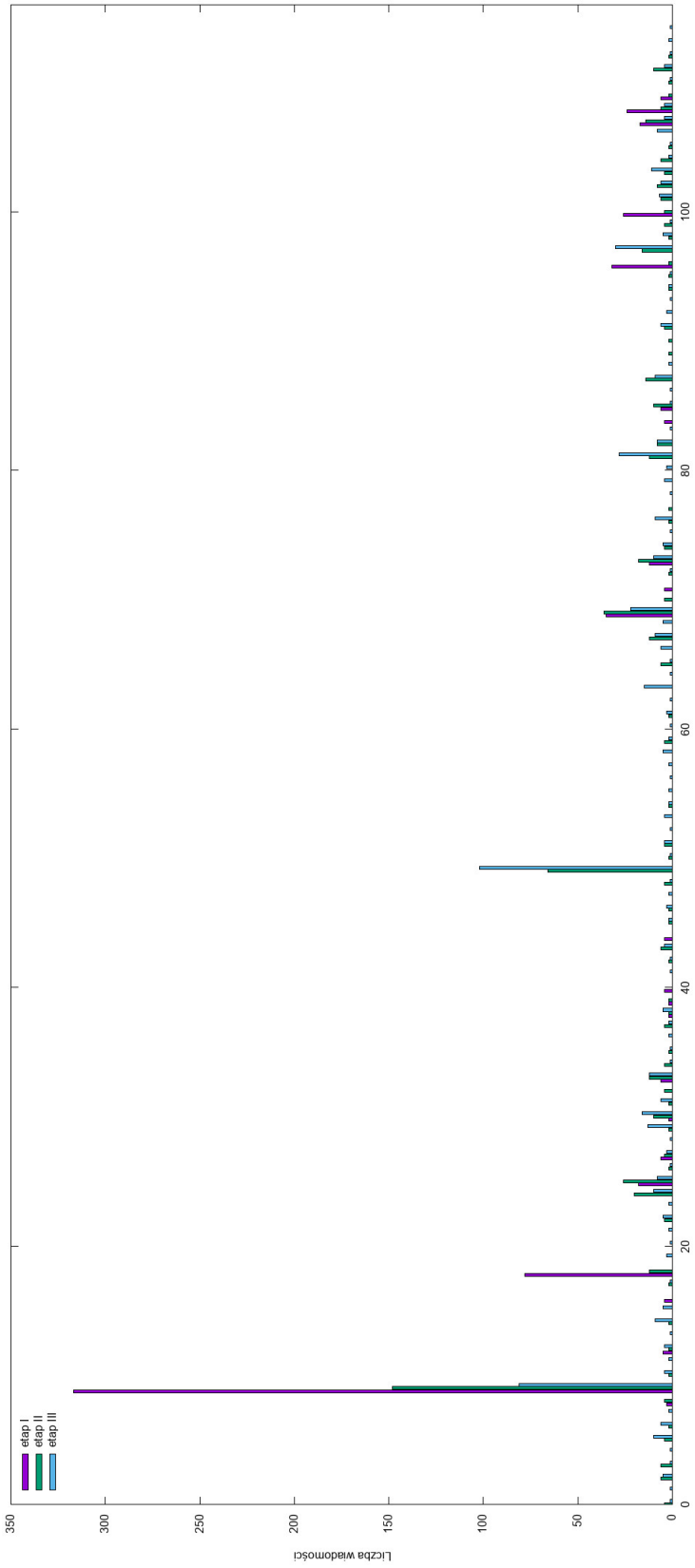
Rysunek 9.8: Przypisanie wiadomości do folderów dla skrzynki lokay-m



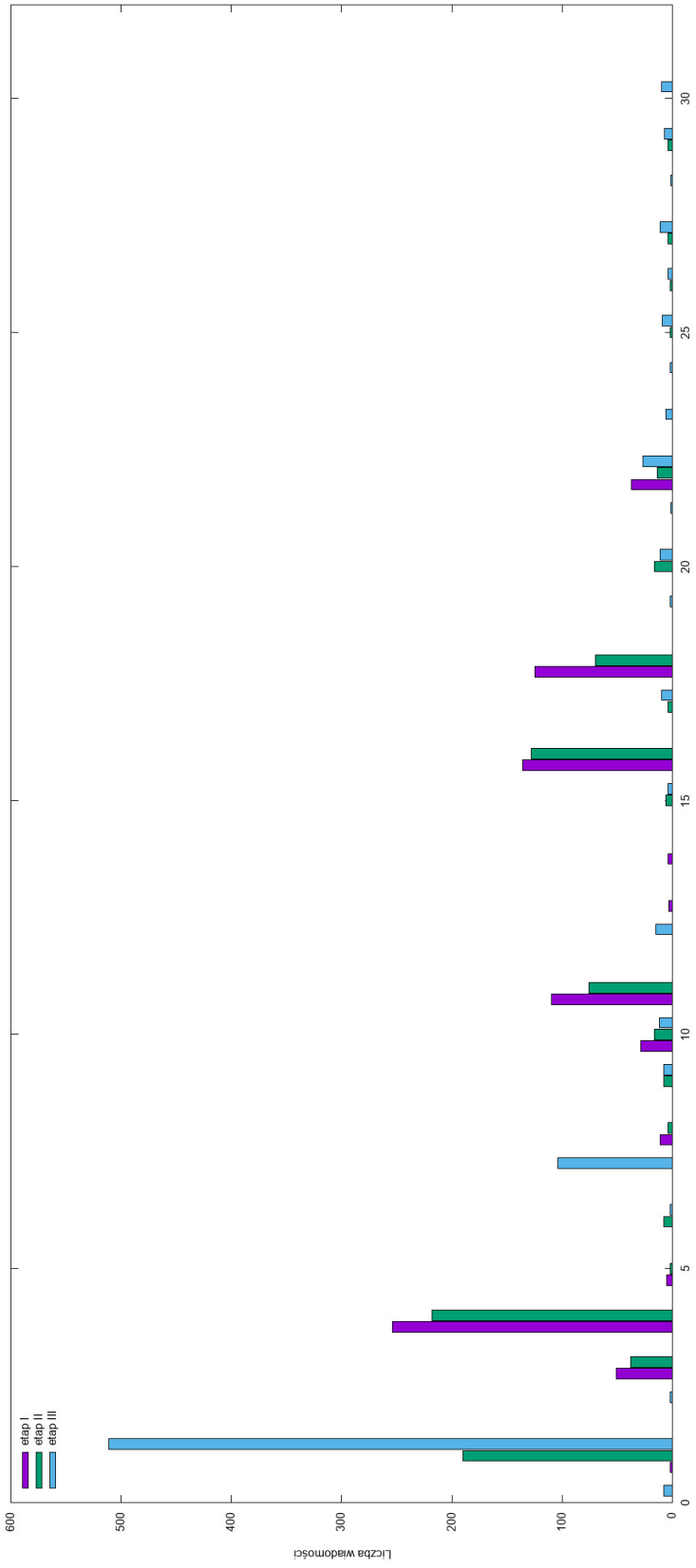
Rysunek 9.9: Przypisanie wiadomości do folderów dla skrzynki sanders-r



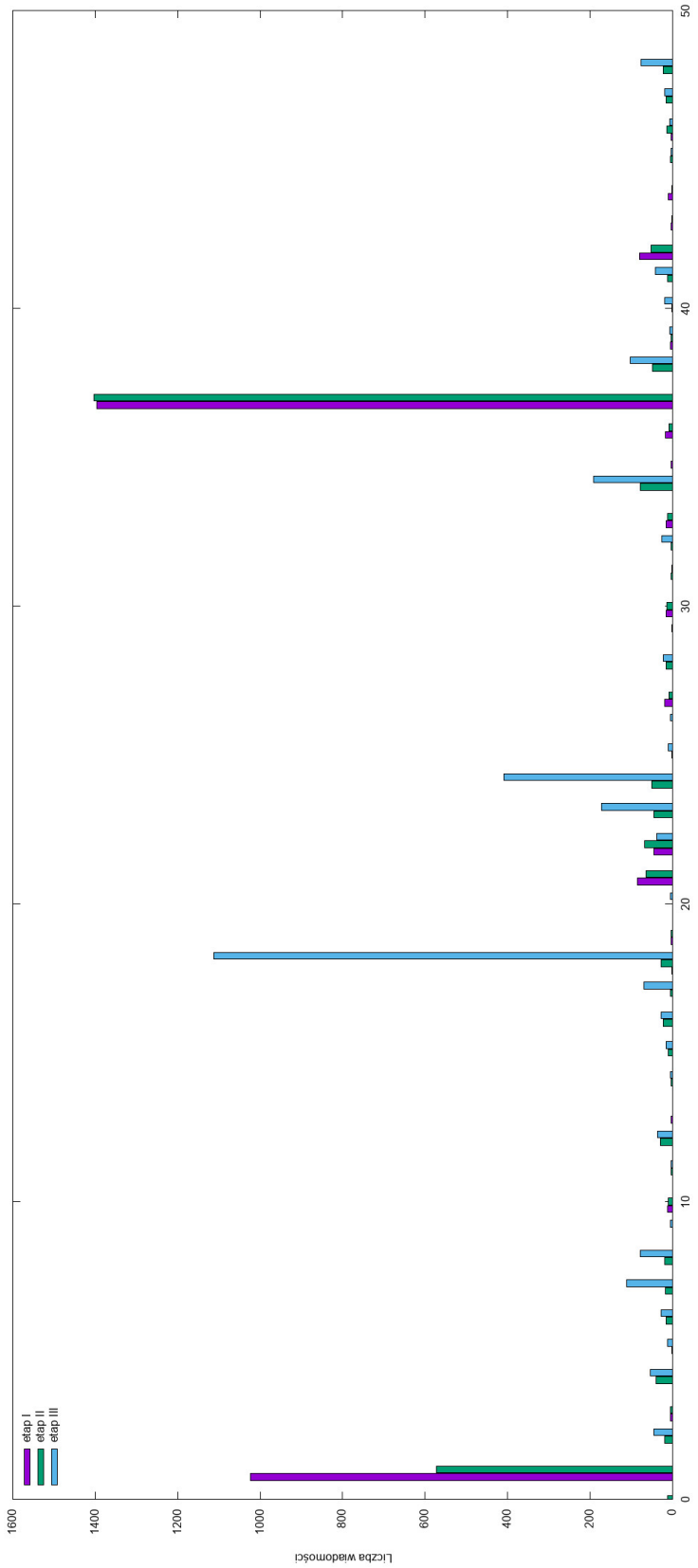
Rysunek 9.10: Przypisanie wiadomości do folderów dla skrzynki shacleton-s



Rysunek 9.11: Przypisanie wiadomości do folderów dla skrzynkisteffes-j



Rysunek 9.12: Przypisanie wiadomości do folderów dla skrzynki rogers-b



Rysunek 9.13: Przypisanie wiadomości do folderów dla skrzynki wiliams-w3

9

Podsumowanie

Niniejsza rozprawa, której tematem jest automatyczna kategoryzacja wiadomości elektronicznych z zastosowaniem sieci społecznych oraz algorytmów mrowiskowych składa się z dwóch części. Pierwsza, to w dużym stopniu część poznawcza, która dotyczy m.in. zastosowania sieci społecznych czy algorytmów mrowiskowych, natomiast druga poświęcona jest nowemu algorytmowi do automatycznego przypisania wiadomości elektronicznych do folderów. W przypadku obydwu problemów omówione zostały niezbędne zagadnienia teoretyczne oraz wykonano praktyczną realizację związaną z implementacją algorytmów oraz analizą wyników przeprowadzonych eksperymentów.

Wszystkie cele postawione na początku rozprawy zostały zrealizowane, a w ich wyniku potwierdzona została postawiona teza rozprawy, a więc zastosowanie algorytmów mrowiskowych i sieci społecznych w problemie automatycznego kategoryzowania wiadomości e-mail pozwala na poprawę trafności przypisywania wiadomości do folderów oraz umożliwia sugerowanie zakładania nowych folderów dla użytkowników. W trakcie realizacji celów dokładnie przeanalizowano zbiór danych Enron E-mail (rozdział 4), oczyszczono go (rozdział 4.4) dostosowując do analizowanego problemu oraz przekształcono do odpowiedniej struktury (rozdział 4.5).

Stworzono sieć społeczną opartą na kontaktach pomiędzy nadawcami a odbiorcami wiadomości e-mail (rozdział 7), a także na podstawie przeprowadzonej analizy i obserwacji sieci społecznej wyodrębniono grupy użytkowników posiadających podobną strukturę społeczną (rozdział 7.4). Następnie dla wszystkich użytkowników wybranych grup przekształcono ich skrzynki pocztowe do postaci tabel decyzyjnych (rozdział 4.5). Na ich podstawie zbudowano klasyfikator stosując algorytmy mrowiskowe, dzięki którym możliwe jest przeszukiwanie większej przestrzeni rozwiązań i znajdowanie alternatywnych metod rozwiązań (rozdział 8).

Przeprowadzono badania stosując klasyczne klasyfikatory (rozdział 5), zespo-

ły klasyfikatorów (rozdział 6), a także algorytmy mrowiskowe do konstruowania drzew decyzyjnych (rozdział 6.1) oraz lasów decyzyjnych (rozdział 6.2), a następnie porównano otrzymane rezultaty. Analiza otrzymanych wyników przyczyniła się do zaprojektowania autorskiego algorytmu do automatycznego kategoryzowania wiadomości e-mail do folderów (rozdział 8), którego zastosowanie umożliwia sugerowanie zakładania nowych folderów dla użytkowników (rozdział 9).

Na podstawie przeprowadzonych eksperymentów potwierdzone zostało poprawienie trafności automatycznej kategoryzacji wiadomości e-mail do folderów przy zastosowaniu algorytmów mrowiskowych oraz analizy sieci społecznych. Już sama analiza specjalnie przygotowanej tabeli decyzyjnej przy zastosowaniu zaadaptowanego algorytmu ACDT pozwoliła na otrzymanie satysfakcjonujących wyników. Stworzenie sieci społecznej, na podstawie której dokonano analizy komunikacji pomiędzy użytkownikami pozwoliło na znaczne poprawienie wyników.

Wszystkie wyniki eksperymentów, które mogły zostać zweryfikowane za pomocą testu statystycznego zostały poddane analizie mającej na celu określenie istotności różnic pomiędzy rezultatami uzyskiwanymi przez klasyfikatory. Dla celów porównania zastosowano nieparametryczny test Manna-Whitneya-Wilcoxona (test sumy rang Wilcoxona dla dwu próbek). Dla wszystkich analiz hipoteza mówiąca o braku różnic pomiędzy dwoma próbkami została odrzucona, w związku z czym potwierdzono, że zaproponowany algorytm uzyskuje lepsze rezultaty niż porównywane z nim algorytmy. Dodatkowo, dla porównania ze sobą niezależnych wykonań zaproponowanego algorytmu, w wyniku testu potwierdzona została hipoteza o braku różnic pomiędzy dwoma próbkami. Potwierdza to powtarzalność wyników uzyskiwanych przez proponowany algorytm.

Analiza algorytmu do automatycznego kategoryzowania wiadomości e-mail do folderów zaproponowanego w niniejszej rozprawie, przedstawione wyniki badań eksperymentalnych oraz opisane wnioski mogą stanowić podstawę do dalszej pracy związanej z omawianym problemem. Badania zamieszczone w rozprawie wykonane zostały na zbiorach danych o różnych wielkościach, jednak zasadniczą kwestią, która powinna zostać rozwiązana jest dostosowanie algorytmu do pracy ze zbiorami danych, w których obiekty opisane są m.in. przez atrybuty o wartościach ciągłych.

Proponowany algorytm został także zastosowany do opracowania mechanizmu sugerowania zakładania nowych folderów i umieszczenia w tych folderach wiadomości, na podstawie struktury folderów innych użytkowników z grupy. W tym przypadku działanie algorytmu było takie samo jak metoda opisana w rozdziale 4.5, z tą różnicą, że analizowane były wszystkie foldery z całej grupy, a nie tylko jednego użytkownika.

Bibliografia

[1] Enron E-mail Dataset. Dostępne w Internecie: `https://www.cs.cmu.edu/~./enron/` [dostęp: 2017-02-25 18:30].

[2] Gottfried Achenwall. Abriss der neuesten staatswissenschaft der heutigen vornehmsten europäischen reiche und republiken. *Göttingen: Schmidt*, 1749.

[3] Manu Aery, Sharma Chakravarthy. emailsift: Email classification based on structure and content. *Data Mining, Fifth IEEE International Conference on*, strony 8–pp. IEEE, 2005.

[4] Tiago A Almeida, Akebo Yamakami. Content-based spam filtering. *Neural Networks (IJCNN), The 2010 International Joint Conference on*, strony 1–7. IEEE, 2010.

[5] Nikolaos Ampazis, Helen Iakovaki, Georgios Dounias. Author identification of e-mail messages with olmam trained feedforward neural networks. *Tools with Artificial Intelligence, 2007. ICTAI 2007. 19th IEEE International Conference on*, wolumen 2, strony 413–417. IEEE, 2007.

[6] S. Aral, M. Van Alstyne. Network structure & information advantage, 2007.

[7] Albert-László Barabási, Réka Albert. Emergence of scaling in random networks. *science*, 286(5439):509–512, 1999.

[8] Albert-Laszlo Barabâsi, Hawoong Jeong, Zoltan Néda, Erzsebet Ravasz, Andras Schubert, Tamas Vicsek. Evolution of the social network of scientific collaborations. *Physica A: Statistical mechanics and its applications*, 311(3):590–614, 2002.

[9] R. Beckers, S. Goss, J.L. Deneubourg, Pasteels J.M. Colony size, communication and ant foraging strategy. *Psyche*, 96:239–256, 1989.

[10] R. Bekkerman, A. McCallum, G. Huang. Automatic categorization of email into folders: Benchmark experiments on enron and sri corpora. *Center for Intelligent Information Retrieval, Technical Report IR*, 2004.

[11] U. Boryczka, J. Kozak. Ant Colony Decision Trees – a new method for constructing decision trees based on Ant Colony Optimization. *Computational Collective Intelligence. Technologies and Applications*, LNCS, strony 373–382. Springer, 2010.

[12] Urszula Boryczka, Jan Kozak. An adaptive discretization in the ACDT algorithm for continuous attributes. *Computational Collective Intelligence. Technologies and Applications*, wolumen 6923 serii *LNCS*, strony 475–484. Springer Berlin, 2011.

[13] Urszula Boryczka, Jan Kozak. Ant Colony Decision Forest Meta-ensemble. *ICCCI (2)*, strony 473–482, 2012.

[14] Urszula Boryczka, Jan Kozak. On-the-go adaptability in the new ant colony decision forest approach. *Intelligent Information and Database Systems, ACIIDS*, strony 157–166, 2014.

[15] L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.

[16] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[17] L. Breiman, J. H. Friedman, R. A. Olshen, C. J. Stone. *Classification and Regression Trees*. Chapman & Hall, New York, 1984.

[18] José M Carmona-Cejudo, Gladys Castillo, Manuel Baena-García, Rafael Morales-Bueno. A comparative study on feature selection and adaptive strategies for email foldering. *Intelligent Systems Design and Applications (ISDA), 2011 11th International Conference on*, strony 1294–1299. IEEE, 2011.

[19] José M Carmona-Cejudo, Gladys Castillo, Manuel Baena-García, Rafael Morales-Bueno. A comparative study on feature selection and adaptive strategies for email foldering using the abc-dynf framework. *Knowledge-Based Systems*, 46:81–94, 2013.

[20] Anurat Chapanond, Mukkai S Krishnamoorthy, Bülent Yener. Graph theoretic and spectral analysis of enron email data. *Computational & Mathematical Organization Theory*, 11(3):265–281, 2005.

[21] Priyanka Chhabra, Rajesh Wadhvani, Sanyam Shukla. Spam filtering using support vector machine. *Special Issue IJCCT*, 1(2):3, 2010.

[22] Paweł Cichosz. *Systemy uczące się*. Wydawnictwa Naukowo-Techniczne, 2000.

[23] Peter Clark, Tim Niblett. The cn2 induction algorithm. *Machine Learning*, strony 261–283, 1989.

[24] William W Cohen, i in. Learning rules that classify e-mail. *AAAI spring symposium on machine learning in information access*, wolumen 18, strona 25. California, 1996.

[25] A. Colorni, M. Dorigo, V. Maniezzo, M. Trubian. Ant system for job–shop scheduling. *Belgian Journal of Operations Research, Statistics and Computer Science (JORBEL)*, 34:39–53, 1994.

[26] Corinna Cortes, Vladimir Vapnik. Support-vector networks. *Machine Learning*, 1995.

[27] Bin Cui, Anirban Mondal, Jialie Shen, Gao Cong, Kian-Lee Tan. On effective e-mail classification via neural networks. *International Conference on Database and Expert Systems Applications*, strony 85–94. Springer, 2005.

[28] Jonathon N. Cummings, Rob Cross. Structural properties of work groups and their consequences for performance. *Social Networks*, 25:197–210, 2003.

[29] Karl F. Doerner, Daniel Merkle, Thomas Stützle. Special issue on ant colony optimization. *Swarm Intelligence*, 3(1):1–2, 2009.

[30] M. Dorigo, G. Di Caro. *New Ideas in Optimization*. McGraw–Hill, London, UK, 1999.

[31] M. Dorigo, G. Di Caro, L. Gambardella. Ant algorithms for distributed discrete optimization. *Artif. Life*, 5(2):137–172, 1999.

[32] M. Dorigo, V. Maniezzo, A. Colorni. The ant system: an autocatalytic optimization process. Raport instytutowy 91-016, Department of Electronics, Politecnico di Milano, Italy, 1996.

[33] Marco Dorigo, Mauro Birattari, Christian Blum, Maurice Clerc, Thomas Stützle, Alan F. T. Winfield, redaktorzy. *Ant Colony Optimization and Swarm Intelligence, 6th International Conference, ANTS 2008*, wolumen 5217 serii *LNCS*. Springer, 2008.

[34] Marco Dorigo, Mauro Birattari, Thomas Stützle, Université Libre, De Bruxelles, Av F. D. Roosevelt. Ant colony optimization – artificial ants as a computational intelligence technique. *IEEE Comput. Intell. Mag*, 1:28–39, 2006.

[35] B. Efron. Bootstrap methods: Another look at the jackknife. *The Annals of Statistics*, 7(1):1–26, 1979.

[36] P Erdős, Alfréd Rényi. Some further statistical properties of the digits in cantor's series. *Acta Mathematica Hungarica*, 10(1-2):21–29, 1959.

[37] Leonhard Euler. Solutio problematis ad geometriam situs pertinentis. *Commentarii academiae scientiarum Petropolitanae*, 8:128–140, 1741.

[38] Ronald Aylmer Fisher. *Statistical methods for research workers.* Genesis Publishing Pvt Ltd, 1925.

[39] Alex A Freitas, Rafael S Parpinelli, Heitor S Lopes. Ant colony algorithms for data classification. *Encyclopedia of Information Science and Technology, Second Edition*, strony 154–159. IGI Global, 2009.

[40] Y. Freund, R. E. Schapire. Experiments with a new boosting algorithm. *International Conference on Machine Learning*, strony 148–156, 1996.

[41] Y. Freund, R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997.

[42] Eugeniusz Gatnar. *Symboliczne metody klasyfikacji danych.* Wydaw. Naukowe PWN, 1998.

[43] Peter Gloor, Francesca Grippa, Johannes Putzke, Casper Lassenius, Hauke Fuehres, Kai Fischbach, Detlef Schoder. Measuring social capital in creative teams through sociometric sensors. *International Journal of Organisational Design and Engineering*, 2012.

[44] Peter A. Gloor. *Swarm Creativity: Competitive Advantage through Collaborative Innovation Networks.* Oxford University Press, USA, 2006.

[45] Edmond Halley. An estimate of the degrees of the mortality of mankind, drawn from curious tables of the births and funerals at the city of breslaw; with an attempt to ascertain the price of annuities upon lives. 1693.

[46] Smith Padhric Hand David, Mannila Heikki. *Eksploracja danych.* 2005.

[47] John H. Holland. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence.* MIT Press, Cambridge, MA, USA, 1975.

[48] Sergei Izrailev, Dimitris Agrafiotis. A novel method for building regression tree models for qsar based on artificial ant colony systems. *Journal of Chemical Information and Computer Sciences*, 41(1):176–180, 2001.

[49] Shih-Wen Ke, Chris Bowerman, Michael Oakes. Perc: A personal email classifier. *Advances in Information Retrieval*, strony 460–463. Springer, 2006.

[50] M. Kearns. Thoughts on hypothesis boosting. Project for Ron Rivest's machine learning course at MIT. Rękopis. Dostępne w Internecie: `http://www.cis.upenn.edu/~mkearns/papers/boostnote.pdf` [dostęp: 2010-07-20 23:00], 1988.

[51] S. Kiritchenko, S. Matwin. Email classification with co–training. Raport instytutowy, University of Ottawa, 2002.

[52] Bryan Klimt, Yiming Yang. The enron corpus: A new dataset for email classification research. *Machine learning: ECML 2004*, strony 217–226. Springer, 2004.

[53] M. Kowalkiewicz. Odkrywanie wiedzy. www.kie.ae.poznan.pl/~marek/.

[54] Vladimir I Levenshtein. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet physics doklady*, wolumen 10, strony 707–710, 1966.

[55] David D. Lewis. *Representation and Learning in Information Retrieval*. Praca doktorska, Department of Computer Science, University of Massachusetts, 1992.

[56] David Martens, Bart Baesens, Tom Fawcett. Editorial survey: swarm intelligence for data mining. *Machine Learning*, 82(1):1–42, 2011.

[57] Z. Michalewicz. *Algorytmy genetyczne + struktury danych = programy ewolucyjne*. Wydawnictwo Naukowo-Techniczne, Warszawa, 1999.

[58] Zbigniew Michalewicz. *Algorytmy genetyczne+ struktury danych*. Wydawnictwa Naukowo-Techniczne, 2003.

[59] Jacob Levy Moreno, i in. *Who Shall Survive? Foundations of Sociometry, Group Psychotherapy and Sociodrama*. Beacon House, Beacon, NY, 1953,1978.

[60] Jacob Levy Moreno, Helen Hall Jennings, i in. *Who Shall Survive?: A New Approach to the Problem of Human Interrelations*, wolumen 58. Nervous and Mental Disease Publishing Co, 1934.

[61] T. Morzy. Eksploracja danych: problemy i rozwiązania. *Materiały konferencyjne PLOUG*, Zakopane, 1999.

[62] T. Morzy, A. Leceniewska. Eksploracja danych. http://wazniak.mimuw.edu.pl/images/f/f5/ED-4.2-m01-1.0-kolor.pdf, 2006.

[63] Tadeusz Morzy. *Eksploracja danych: metody i algorytmy.* Wydawnictwo Naukowe PWN, 2013.

[64] Jan Jerzy Mulawka. *Systemy ekspertowe.* Wydawnictwa Naukowo-Techniczne, 1997.

[65] Walenty Ostasiewicz. *Statystyczne metody analizy danych.* Wydawnictwo Akademii Ekonomicznej im. Oskara Langego, 1999.

[66] Fernando EB Otero, Alex A Freitas, Colin G Johnson. Inducing decision trees with an ant colony optimization algorithm. *Applied Soft Computing*, 12(11):3615–3626, 2012.

[67] Witold Paleczek. *Metody analizy danych (na przykładach).* Wydaw. PC, 2004.

[68] Rafael S Parpinelli, Heitor S Lopes, Alex Alves Freitas. Data mining with an ant colony optimization algorithm. *IEEE Transactions on evolutionary computation*, 6(4):321–332, 2002.

[69] Zbigniew Pawłowski. *Ekonometria.* Pa/c\anstwowe Wydawnictwo Naukowe, 1964.

[70] Terry R. Payne, Peter Edwards. Interface agents that learn an investigation of learning issues in a mail agent interface. *Applied Artificial Intelligence*, strony 1–32, 1997.

[71] J. R. Quinlan. Induction of decision trees. *Machine Learning*, 1(1):81–106, 1986.

[72] J. R. Quinlan. *C4.5: Programs for Machine Learning.* Morgan Kaufmann, San Mateo, CA, 1993.

[73] J. R. Quinlan. Improved use of continuous attributes in c4.5. *J. Artif. Intell. Res. (JAIR)*, 4:77–90, 1996.

[74] Źródło Internetowe. Wikipedia, wolna encyklopedia. http://pl.wikipedia.org/. Polska wersja encyklopedii.

[75] Jason Rennie. ifile: An application of machine learning to e-mail filtering. *Proc. KDD 2000 Workshop on Text Mining, Boston, MA*, 2000.

[76] Luis EC Rocha, Fredrik Liljeros, Petter Holme. Simulated epidemics in an empirical spatiotemporal network of 50,185 sexual contacts. *PLoS Comput Biol*, 7(3):e1001109, 2011.

[77] RSES Rough Set Exploration System. Dostępne w Internecie: `http://logic.mimuw.edu.pl/~rses/` [dostęp: 2010-01-10 22:30].

[78] C. Rudin, R. E. Schapire. Margin-based ranking and an equivalence between AdaBoost and RankBoost. *J. Mach. Learn. Res.*, 10:2193–2232, 2009.

[79] Ibrahim Farag Sabah Sayed, Samir AbdelRahman. Three-phase tournament-based method for better email classification. *International Journal of Artificial Intelligence & Applications*, 2012.

[80] Mehran Sahami, Susan Dumais, David Heckerman, Eric Horvitz. A bayesian approach to filtering junk e-mail. *Learning for Text Categorization: Papers from the 1998 workshop*, wolumen 62, strony 98–105, 1998.

[81] Neeti Saxena, Bharati Verma, Nitin Shukla. Online email classification using ant clustering algorithm. *Int. Journal of Emerging Technology and Advanced Engineering*, 2, 2012.

[82] R. E. Schapire. The strength of weak learnability. *Machine Learning*, 5:197–227, 1990.

[83] Richard B Segal, Jeffrey O Kephart. Mailcat: An intelligent assistant for organizing e-mail. *Proceedings of the third annual conference on Autonomous Agents*, strony 276–282. ACM, 1999.

[84] Paweł Stępka, Konrad Subda. Wykorzystanie analizy sieci społecznych (sna) do budowy organizacji opartej na wiedzy. *E-mentor*, 1:28, 2009.

[85] Goss S. Gervet J. Deneubourg J.L. Theraulaz, G. Swarm intelligence in wasps colonies: an example of task assignment in multiagents systems. *Proceedings of the 1990 IEEE International Symposium on Intelligent Control*, strony 135–143, 1990.

[86] Shrawan Kumar Trivedi, Shubhamoy Dey, Prabandh Shikhar. Effect of various kernels and feature selection methods on svm performance for detecting email spams. *International Journal of Computer Applications*, 66(21), 2013.

[87] Shrawan Kumar Trivedi, Shubhamoy Dey, Prabandh Shikhar. Effect of various kernels and feature selection methods on svm performance for detecting email spams. *International Journal of Computer Applications*, 66(21), 2014.

[88] J.C. Verhaeghe, J.L. Deneubourg. Experimental study and modelling of food recruitment in the ant tetramorium impurum. *Insectes Sociaux*, 30:347–360, 1983.

[89] Denil Vira, Pradeep Raja, Shidharth Gada. An approach to email classification using bayesian theorem. *Global Journal of Computer Science and Technology*, 2012.

[90] Man Wang, Yifan He, Minghu Jiang. Text categorization of enron email corpus based on information bottleneck and maximal entropy, 2010.

[91] Zhongjian Wang, Zongjie Wang, Yanfeng Gao, Yanfen Lin. Algorithm of e-mail classification based on automatic adapting for user. *Int. Journal of u-and e-Service, Science and Technology*, 8(2):235–242, 2015.

[92] Stanley Wasserman, Katherine Faust. *Social network analysis: Methods and applications*, wolumen 8. Cambridge university press, 1994.

[93] Duncan J Watts, Steven H Strogatz. Collective dynamics of 'small-world'networks. *nature*, 393(6684):440–442, 1998.

[94] Garnett Carl Wilson, Wolfgang Banzhaf. Discovery of email communication networks from the enron corpus with a genetic algorithm using social network analysis., 2009.

[95] Ian H. Witten, Eibe Frank, Mark A. Hall. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Publishers Inc., wydanie 3rd, 2011.

[96] Ke Xu, Cui Wen, Qiong Yuan, Xiangzhu He, Jun Tie. A mapreduce based parallel svm for email classification. *Journal of Networks*, 9(6), 2014.

[97] Seongwook Youn, Dennis McLeod. Spam email classification using an adaptive ontology. *JSW*, 2(3):43–55, 2007.

[98] M. Zakrzewicz. Data mining i odkrywanie wiedzy w bazach danych. *Materiały konferencyjne PLOUG*, Zakopane, 1997.

[99] Peng Zhang, Jinliang Wang, Xiaojia Li, Menghui Li, Zengru Di, Ying Fan. Clustering coefficient and community structure of bipartite networks. *Physica A: Statistical Mechanics and its Applications*, 387(27):6869–6875, 2008.

Spis rysunków

Spis tabel