ALGORYTMY I MODELE OBLICZENIOWE W ANALIZIE CHEMICZNEJ

AUTOREFERAT

Grzegorz Skoraczyński

Początki chemometrii sięgają lat 60. XX wieku, kiedy następował rozwój języków programowania wysokiego poziomu, takich jak Fortran. Głównymi obszarami badań chemometrycznych były między innymi przetwarzanie zbiorów danych uzyskanych z analizy instrumentalnej [1] lub wspomagane komputerowo planowanie syntezy [2, 3]. Równolegle z rozwojem chemometrii badacze byli zainteresowani zautomatyzowanym przetwarzaniem informacji chemicznych [4, 5], jak na przykład kodowaniem struktury chemicznej [6], czy efektywnym przechowywaniem danych w bazach danych [7]. Wraz z rozwojem publicznych i komercyjnych baz danych oraz rosnącą ilością przetwarzanych informacji, w latach 80. cheminformatyka stwała się coraz bardziej popularna [8], a w latach 90. została uznana za odrębną dyscyplinę [9].

W niniejszej pracy podejmujemy dwa niżej opisane problemy chemometrii i cheminformatyki:

- problem uliniowienia czasu retencji (RT) w zautomatyzowanej analizie danych z zakresu chromatografii cieczowej ze spektrometrią mas (LC-MS),
- problem predykcji syntezowalności cząsteczek jako heurystyki retrosyntezy.

1 PROBLEM ULINIOWIENIA CZASU RETENCJI

Jednym z wyzwań analizy zbiorów danych LC-MS jest korekcja błędów spowodowanych dryfem czasu retencji. Dryf czasu retencji stał się istotny wraz z pojawieniem się wysokoprzepustowych kolumn, dla których separacja zajmuje stosunkowo dużo czasu, zwykle do kilku godzin, a czas elucji peptydów może wahać się nawet o 10 minut [10]. Może on być skorygowany przez protokół eksperymentalny tylko w ograniczonym zakresie [11]. Może on wpływać na cały gradient lub tylko na pojedyncze piki, zamieniając kolejność ich elucji. Zmiany te mogą mieć różne przyczyny, takie jak zmiana lub starzenie się kolumny, niestabilność chemiczna próbki lub niedokładne przygotowanie eksperymentu [12].

Dryf czasu retencji wymaga korekty, zwykle nazywanej uliniowieniem. W efekcie uzyskujemy odpowiedniość sygnałów między powtarzanymi eksperymentami LC-MS. Zdecydowana większość podejść do

uliniowienia czasu retencji ma ograniczoną stosowalność, ponieważ są projektowane przy założeniu, że jony eluują monotonicznie w funkcji czasu retencji. W związku z tym nie są w stanie poradzić sobie ze zamianą kolejności elucji.

Jednym ze stosowanych podejść do uliniowienia są algorytmy, które znajdują odpowiedniość wcześniej wykrytych cech dwóch lub więcej chromatogramów, np. OpenMS [13] lub Quandenser [14]. Cechy to wypukłe zbiory pików reprezentujące sygnał pojedynczego analitu, natomiast cechy które odpowiadają sobie pomiędzy eksperymentami nazywamy cechami konsensusowymi. Zgodnie z najlepszą wiedzą autorów, wszystkie algorytmy dopasowujące cechy redukują je do pojedynczego punktu zawierającego pik monoizotopowy i średni czas retencji, ignorując informacje na przykład o obwiedni izotopowej. Bez charakterystyki przestrzennej cech i informacji o koeluujących jonach, zamiany kolejności elucji są praktycznie niewykrywalne [15]. Głównym powodem tego uproszczenia jest trudność w znalezieniu odpowiednich miar do porównywania cech. Zwykle stosuje się dystans euklidesowy między punktami lub jednowymiarowe oceny podobieństwa np. kosinus [16].

W tej pracy przedstawiamy Alignsteina [17], algorytm do uliniowienia czasu retencji w chromatografii cieczowej ze spektrometrią mas. Znajduje on dopasowanie sygnałów nawet o zamienionej kolejności elucji. W tym celu zaimplementowaliśmy uogólnienie dystansu Wassersteina jako miarę podobieństwa cech, która pozwala na uniknięcie redukcji informacji lub wymiaru analizowanych danych (patrz Sekcja 1.1). Co więcej, Alignstein nie wymaga ani próbki referencyjnej, ani wcześniejszej identyfikacji sygnału. Algorytm jest zweryfikowany na publicznie dostępnych danych porównawczych, uzyskując konkurencyjne wyniki.

1.1 *Porównywanie widm masowych*

Zazwyczaj widma masowe są porównywane za pomocą kosinusowej miary podobieństwa [16]. Pomimo swojej popularności, miara ta nie ma zastosowania do porównywania cech, ponieważ nie są one skalowalne z wymiarem [18]. Z tego powodu, proponujemy dystans Wassersteina [19] z dodatkowymi uogólnieniami [20, 21] jako miarę podobieństwa cech.

Odległość Wassersteina jest metryką opartą na teorii optymalnego transportu. Opisuje, jak najtaniej przekształcić jedną cechę w drugą. Przekształcenia te mogą obejmować przesuwanie sygnału, ale też dzielenie lub łączenie sygnału między pikami. Formalnie załóżmy, że mamy dwie cechy dyskretne μ i ν, gdzie $\mu(x)$ jest intensywnością μ dla wartości M/Z x. Następnie definiujemy plan transportu T tak, aby $T(x, y)$ odpowiadał ilości sygnału, który jest transportowany z piku x cechy μ do piku y cechy ν. Koszt transportu to suma ilości przetransportowanego sygnału między wszystkimi parami pików pomnożonych przez odległość między pikami:

$$\sum_{x,y} T(x, y) \cdot d(x, y), \tag{1}$$

gdzie $d(x, y)$ to odległość ℓ_1 między pikami x i y. Dystans Wassersteina W jest równy kosztowi optymalnego planu transportu:

$$W(\mu, \nu) = \min_T \sum_{x,y} T(x, y) \cdot d(x, y). \tag{2}$$

Zauważyliśmy jednak, że dystans Wassersteina nieporawnie znajduje odległość między zaszumionymi widmami. W tym celu, używamy uogólnienia dystansu Wassersteina (GWD) zgodnie z propozycją Chizata i in. [21]. GWD różni się od dystansu Wassersteina głównie możliwością pominięcia transportu części sygnału. Dokładniej, GWD pozwala na pominięcie transportu sygnału na odległość większą niż zdefiniowany przez użytkownika parametr λ ze stałą karą proporcjonalną do λ i ilości nieprzetransportowanego sygnału:

$$W(\mu, \nu) = \min_T \sum_{x,y} \Big(T(x, y) \cdot d(x, y) + \lambda \cdot F(T_\mu, \mu) + \lambda \cdot F(T_\nu, \nu) \Big), \tag{3}$$

gdzie T_μ i T_ν są rozkładami brzegowymi planu transportu a F dywergencją dobraną tak, aby możliwe było przybliżenie planu transportu T do cech μ i ν. Aby obliczyć GWD, regularyzujemy ten problem karą entropijną, co pozwala na szybkie i stabilne numerycznie obliczenia, przy użyciu algorytmu Sinkhorna-Knoppa [22].

Przykład zastosowania dystansu Wassersteina jako miary podobieństwa widm

Jednym ze skutecznych przykładów zastosowania dystansu Wassersteina jako miary podobieństwa widm jest algorytm masserstein [23]. Jest to algorytm do regresji liniowej widm, nazywanej również dekonwolucją. Pod pojęciem dekonwolucji rozumiemy tutaj problem oszacowania proporcji zidentyfikowanych widm referencyjnych w eksperymentalnym widmie pewnej mieszaniny. Załóżmy, że mamy widmo eksperymentalne μ mieszaniny, dla której chcemy policzyć proporcje składników i n widm referencyjnych $\nu_1, \ldots, \nu_n$ substancji. Chcemy znaleźć kombinację wypukłą proporcji widm referencyjnych $\boldsymbol{p} = (p_0, p_1, \ldots, p_n)^\intercal$, takie że $p_0 + p_1 + \ldots + p_n = 1$, która najlepiej modeluje widmo eksperymentalne. Zdefiniujmy także widmo ν_m, które będzie opisywać widmo eksperymentalne za pomocą widm referencyjnych:

$$\nu_m = p_1 \cdot \nu_1 + \ldots + p_n \cdot \nu_n.$$

Aby uwzględnić pewną ilość sygnału, której nie wyjaśniają widma referencyjne (szum), wprowadzamy dodatkowe widmo pomocnicze ω. Najlepsze dopasowanie widma eksperymentalnego za pomocą kombinacji widm referencyjnych wyrażamy jako problem minimalizacji dystansu Wassersteina:

$$\boldsymbol{p}^* = \operatorname*{argmin}_{\boldsymbol{p}} W(\mu, p_0 \cdot \omega + \nu_m).$$

ω interpretujemy tak, że koszt transportu sygnału do tego widma jest stały i równy κ.

1.2 *Alignstein, schemat algorytmu*

Alignstein na początku wykonuje preprocessing cech, który polega na między innymi ich normalizowaniu i skalowaniu w wymiarze czasu

retencji, tak aby jego zmienność była podobnej wielkości jak zmienność w wymiarze M/Z. Następnie uliniowienie przeprowadzane jest w dwóch fazach: najpierw środki ciężkości są klasteryzowane, a następnie cechy są dopasowywane. Środki ciężkości cech ze wszystkich chromatogramów są klasteryzowane w celu stworzenia kandydatów na cechy konsensusowe, którzy są weryfikowani w fazie dopasowywania cech. Podczas tej fazy algorytm wyszukuje pary najbardziej podobnych cech we wszystkich chromatogramach. Odbywa się to poprzez znalezienie dopasowania cech o minimalnym koszcie, gdzie koszt jest równy sumie GWD między dopasowanymi cechami. Interpretujemy ten problem jako znalezienie maksymalnego przepływu o minimalnym koszcie w odpowiednio zaprojektowanej sieci przepływowej, odpowiednio iteracyjnie powtarzanego dla każdego chromatogramu. Problem ten ma odpowiednio sformułowane ograniczenia, dzięki czemu wynik jest zgodny z wymaganiami, np. algorytm traktuje jednakowo (symetrycznie) wszystkie chromatogramy wejściowe i nie wymaga próby referencyjnej.

1.3 *Walidacja algorytmu na danych porównawczych*

Oceniliśmy poprawność Alignsteina, porównując go z innymi algorytmami na publicznie dostępnych danych porównawczych. Odtworzyliśmy protokół oceny z pracy Lange i in. [24] (dalej nazywane badaniem CAAP). Przeanalizowaliśmy dwa zbiory danych proteomicznych z CAAP: P1 i P2 oraz jeden metabolomiczny: M1. Aby ocenić poprawność algorytmów uliniowienia, autorzy badania zaproponowali miary precyzji i czułości ulioniowienia. Ponadto, zgodnie z propozycją autorów algorytmu SIMA [25], obliczyliśmy statystykę F, będącą średnią harmoniczną precyzji i czułości uliniowienia. W porównaniu uwzględniliśmy publicznie dostępne wyniki algorytmów opublikowanych niedawno: MZMine 2 [26], SIMA [25], MassUntagler [27] (tylko zbiór P1) oraz Wandy et al. [28].

Alignstein uzyskał wysoce konkurencyjne wyniki na danych CAAP. W przypadku zbioru P1 idealnie dopasował do prawie wszystkie cechy, jego precyzja i czułość uliowienia wyniosły średnio 0,94, podobnie jak MZmine 2 i OpenMS.

W przypadku zbioru P2 osiągnęliśmy najwyższą średnią wartość czułości uliowienia (średnio 0,82), tj. nasze podejście miało najmniejszą liczbę niedopasowanych cech. Miał niższą średnią precyzję uliowienia równą 0,73 i ustępował jedynie OpenMS. Ogólnie uzyskaliśmy najlepszą średnią wartość statystyki F, równą 0,77. Dla zbioru danych M1 Alignstein osiągnął konkurencyjne wyniki: precyzję uliowienia równą 0,88, czułość uliowienia równą 0,91, i statystykę F równą 0,89. Potwierdza to, że Alignstein skutecznie skaluje się z liczbą chromatogramów.

1.4 *Zastosowanie Alignsteina do detekcji biomarkerów*

Alignstein jest w stanie wykrywać pewne biomarkery. Aby to zweryfikować, przeanalizowaliśmy dane z pracy Barrangera i in. [29]. Zawierała ona chromatogramy LC-MS/MS białka jelitowego z małży morskich zanieczyszczanych in vivo benzo[a]pirenem (BaP) w różnych stężeniach: 0, 5, 50, 100 $\mu g/L$.

Sprawdziliśmy, czy Alignstein rozpoznaje informacje z tandemowych widm masowych (MS/MS) na podstawie przestrzennych właściwości cech LC-MS. W tym celu, wykryliśmy cechy LC-MS i oznaczyliśmy przy pomocy identyfikacji peptydów uzyskanych z danych tandemowych. Poprawność uliowienia została obliczona ilościowo przy użyciu zaproponowanej przez nas czułości identyfikacji (IR) zdefiniowanego jak poniżej. Wybraliśmy wszystkie powtarzające się identyfikacje, które opisywały pewne cechy i obliczyliśmy ułamek z nich, których cechy zostały poprawnie dopasowane. Dla każdego ze stężenia BaP obliczyliśmy IR dla wszystkich uliniowień pomiędzy powtórzeniami technicznymi próbki. Uzyskaliśmy zadowalające wyniki, IR było równe 81 %, 78 %, 85 %, 86 % dla odpowiednio stężeń BaP 0, 5, 50 i 100 $\mu g/L$.

Ponadto sprawdziliśmy, czy Alignstein może wykryć odpowiednie biomarkery dla próbek w różnych warunkach eksperymentalnych. W tym celu powtórzyliśmy powyższą analizę, uliniawiając chromatogramy próbek o różnych stężeniach BaP uzyskując zadowalające wyniki.

1.5 *Uliniawianie sygnałów o zamienionej kolejności elucji*

Sprawdziliśmy, czy Alignstein prawidłowo dopasowuje cechy o zamienionej kolejności elucji. W tym celu zebraliśmy ponad 580 zidentyfikowanych cech z chromatogramów uzyskanych z pracy Barrangera i in. [29]. Symulowaliśmy dryf RT przez losowo przesuwając cechy w zakresie (-150 s, 150 s) w wymiarze RT oraz w zakresie (-0,3 Da, 0,3 Da) w wymiarze M/Z. Dla takich dwóch zbiorów cech: jednego z oryginalnymi cechami i drugiego z cechami poprzesuwanymi, około 2 % (ok. 3400) wszystkich par cech miało zamienioną kolejność elucji. Uliniowiliśmy te dwa zestawy i zmierzyliśmy liczbę prawidłowo dopasowanych cech oraz ułamek prawidłowo rozwiązanych zamienionych par cech. Nasze narzędzie dopasowało praktycznie wszystkie przesunięte cechy (96 %) i zdecydowaną większość zamienionych par cech (91 %).

2 Problem predykcji syntezowalności cząsteczek jako heurystyki retrosyntezy

W komputerowo wspomaganym planowaniu syntezy (ang. computer-assissted synthesis planning, CASP) można wyróżnić dwa główne zagadnienia: planowanie kolejnych reakcji w przód i retrosyntezę. Retrosynteza jest metodą planowania schematu syntezy związków chemicznych począwszy od prostych prekursorów dostępnych w sprzedaży, przez syntetyzowalne półprodukty do cząsteczki docelowej. Planowanie syntezy było pracochłonnym zadaniem wykonywanym ręcznie do lat 60., kiedy Corey [2] sformalizował ideę CASP, a następnie zaimplementował ją w oprogramowaniu LHASA [3]. Z biegiem lat opracowano nowe rozwiązania, które automatyzowały kolejne elementy planowania, wymagały mniejszej ingerencji człowieka, czy zwiększały szybkość i dokładność algorytmów [30, 31, 32]. W ciągu ostatniej dekady niezależnie opracowano kilka nowoczesnych, opartych na uczeniu maszynowym narzędzi do CASP [33, 34, 35].

Głównym ograniczeniem do narzędzi do CASP jest ich złożoność obliczeniowa. W czasie planowania retrosyntetycznego należy przejść przez potencjalnie wykładniczą przestrzeń częściowych wyników. To sprawia, że narzędzia CASP nie są odpowiednie, gdy należy szybko przetestować syntezowalność wielu cząsteczek. Przykładem mogą metody skriningu wirtualnego, podczas którego duża liczba struktur chemicznych jest jednocześnie oceniana pod kątem pewnych właściwości. Poszukiwanie pełnej syntezy dla każdej struktury byłoby trudne obliczeniowo. Jednym z ostatnio popularnych rozwiązań jest predykcja syntezowalności. Modeli do oceny syntezowalności stały sie szczególnie popularne wraz z rozwojem metod uczenia maszynowego, na przykład SAscore [36], SYBA [37], SCScore [38], RAscore [39].

W tej pracy podejmujemy wyzwanie, czy modele do oceny syntezowalności mogą przyspieszyć planowanie retrosyntetyczne poprzez lepszą priorytetyzację częściowych ścieżek syntezy. Proponujemy trzy różne modele oceny syntezowalności. Różnią się one sposobem kodowania struktury molekularnej i zastosowanym modelem uczenia maszynowego (patrz Sekcje 2.1-2.3). Ponadto proponujemy platformę do testowania i porównywania tych narzędzi (patrz Sekcje 2.4.

2.1 *Model 1: podejście poprzez opisywanie współistnienia różnych wzorców strukturalnych*

Pierwszy model wymagał tłumaczenia wzoru strukturalnego cząsteczki na zestaw deskryptorów o stałej długości. W celu stworzenia tych deskryptorów wykorzystaliśmy kolekcję ponad 250 wzorców motywów strukturalnych i bazę danych zawierającą ponad 6 000 000 cząsteczek [40]. Aby znaleźć rozkład liczności motywów w cząsteczkach, oznaczmy M jako liczbę motywów, K jako liczbę obserwacji (cząsteczek w bazie danych), a obecność motywów pojedynczej cząsteczki jako $\boldsymbol{m} = (m_1, \ldots, m_M)^\mathrm{T}$. Analizując deskryptory motywów cząsteczek z bazy stwierdziliśmy, że ich rozkłady brzegowe można przybliżyć rozkładem dzeta, tj.

$$\mathbb{P}(X_i = m_i) = \frac{(m_i + 1)^{-\alpha_i}}{\zeta(\alpha_i)}$$

gdzie $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_i, \ldots, \alpha_M)^\mathrm{T}$ jest wektorem współczynników, ζ to funkcja Riemanna, a X_i to zmienna losowa opisująca rozkład i-tego motywu. Motywy nie są rozmieszczone niezależnie, dlatego podczas szukania rozkładu całkowitego uwzględniliśmy ich zmienną koincydencję (interakcje motywów). Interakcje motywów opisujemy jako $\boldsymbol{\theta} \in \mathbb{R}^{M,M}$, gdzie θ_{ij} opisuje interakcję między i-tym a j-tym motywem. Wówczas całkowity rozkład liczności motywów jest iloczynem rozkładów brzegowych z uwzględnionymi interakcjami, tj.:

$$\mathbb{P}_{\boldsymbol{\alpha}\boldsymbol{\theta}}(\boldsymbol{X} = \boldsymbol{m}) \propto \exp\left(\sum_i \alpha_i \log(m_i + 1) + \sum_{i<j} \theta_{ij} \mathbb{1}(m_i \neq 0)\mathbb{1}(m_j \neq 0)\right). \tag{4}$$

Takie sformułowanie jest markowowskim polem losowym [41]. Parametry α można znaleźć za pomocą estymatora największej wiarygodności dla rozkładu dzeta [42]. Nie możemy jednak wyprowadzić analitycznie zwartego wzoru (4), gdyż oszacowanie współczynników

$\boldsymbol{\theta}$ stanowiło wyzwanie. Zmienne $\boldsymbol{\alpha}$ i $\boldsymbol{\theta}$ oszacowaliśmy poprzez maksymalizację prawdopodobieństwa $\mathbb{P}_{\boldsymbol{\alpha\theta}}(\boldsymbol{X} = \boldsymbol{m})$. Ze względu na kwadratową liczbę nieznanych parametrów chcieliśmy uwzględnić tylko te najistotniejsze, dlatego regularyzowaliśmy ten problem karą LASSO ℓ_1 [43]:

$$(\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\theta}}) = \underset{\boldsymbol{\alpha\theta}}{\operatorname{argmin}} \{ -\log \mathbb{P}_{\boldsymbol{\alpha\theta}}(\boldsymbol{X} = \boldsymbol{m}) + \lambda ||\boldsymbol{\theta}||_1 \}, \tag{5}$$

gdzie λ jest parametrem kary ℓ_1. Rozwiązaliśmy ten problem (5) przy użyciu metody gradientu proksymalnego [44] jako zaproponowane przez Miasojedowa i Rejchela [45] dla modelu Isinga [46]. Ponieważ nie mogliśmy obliczyć gradientu funkcji $\log \mathbb{P}_{\boldsymbol{\alpha\theta}}(\boldsymbol{X} = \boldsymbol{m})$ we wzorze (5), oszacowaliśmy to za pomocą stochastycznej wersji metody proksymalnego gradientu. W tym celu użyliśmy próbnika Gibbsa [47] do aproksymacji łącznego rozkładu poprzez losowanie sekwencji obserwacji. Użyliśmy algorytmu Metropolisa-Hastingsa w próbniku Gibbsa [48], tj. dla każdej współrzędnej wykonaliśmy krok algorytmu Metropolisa-Hastingsa zamiast próbkowania z rozkładu warunkowego. Dla każdego kroku algorytmu spadku wzdłuż gradientu, próbnik Gibbsa był uruchamiany ponad 1000 razy, a współczynniki $\boldsymbol{\alpha}$ i $\boldsymbol{\theta}$ były aktualizowane. Jedna tura algorytmu Metropolisa w próbniku Gibbsa polegała na iteracji po motywach w naturalnym porządku i ponownym próbkowaniu obecności motywu w pewnej abstrakcyjnej cząsteczce. Ta cząsteczka była przekazywana pomiędzy etapami, a dla pierwszego etapu jej wartości były losowe.

Zauważyliśmy, że głównym ograniczeniem tego modelu jest przeuczenie się do znanych i często współwystępujących par motywów, na przykład podobnych lub nakładających się struktur różnych rodzajów alkoholi. Konstrukcja modelu nie była odporna na tego rodzaju interakcje, w wyniku czego nietypowe interakcje pojawiające się tylko w niewielkiej ilości związków zostały pominięte.

2.2 *Model 2: podejście za pomocą uczenia nadzorowanego*

Aby ominąć ograniczenia poprzedniego modelu, zaprojektowaliśmy nowy, nazwany Motif Feasibility Score (MF-Score). Do wcześniej opisanych deskryptorów, dodaliśmy dwa rodzaje deskryptorów: deskryptor masy cząsteczki i deskryptory przestrzennych interakcji. Masa cząsteczkowa opisuje rozmiar cząsteczki oraz jak motywy są upakowane w cząsteczce. Deskryptory przestrzennych interkacji motywów kodują, wzajemnie oddziaływania między fragmentami cząsteczki, które mogą zakłócić jej stabilność. Jako reprezentatywny zbiór danych o istniejących związkach wykorzystaliśmy bazę danych ZINC15 [49]. Dla każdej cząsteczki z bazy danych obliczyliśmy deskryptory motywów i statystyki interakcji. Stworzyliśmy 14 grup motywów wysoce oddziałujących (współwystępujących) i 41 pojedynczych motywów niewchodzących w interakcje. Deskryptory przestrzenne zostały przedstawione jako macierz maksymalnych i minimalnych odległości między wszystkimi parami motywów. Tutaj odległość między dwoma motywami została zdefiniowana jako długość najkrótszej ścieżki między instancjami motywów na grafie odpowiadającym strukturze cząsteczki. Ze względu na dużą rzadkość tej macierzy zredukowaliśmy jej wy-

miar za pomocą analizy głównych składowych (PCA), rzutowaliśmy na 5 pierwszych składowych głównych.

Deskryptory były predyktorami syntezowalności cząsteczki. Zmienna objaśniana była binarna, gdzie 0 odpowiadało nieistniejącym (niesyntezowalnym) cząsteczkom, a 1 odpowiadało istniejącej, łatwo syntezowalnej cząsteczce. Aby przedstawić cząsteczki nieistniejące, stworzyliśmy zestaw deskryptorów, używając metody bootstrap [50].

Na tym zestawie danych został wytranowaliśmy model Gradient Boosting Machines (GBM) [51]. W celu oszczędniejszego wykorzystania pamięci przy jednoczesnym utrzymaniu stabilnych wyników, stworzyliśmy 40 modeli GBM, każdy wytrenowany na ponad 100 000 cząsteczek losowo wybranych z bazy danych. MF-Score był medianą predykcji wszystkich modeli.

Porównaliśmy wyniki MF-Score z SAscore i SCScore na 40 cząsteczkach uzyskanych z pracy o SAscore [36]. Obliczyliśmy współczynnik korelacji rangowej Spearmana [52] między wynikami MF-Score a SAscore i SCScore. Wszystkie wyniki są ze sobą skorelowane. Analiza pojedynczego modelu GBM ujawniła jednak ryzyko przeuczenia do oczywistych wzorców strukturalnych. Tutaj pokazaliśmy, że model może być nadmiernie dopasowany do grup aromatycznych lub zawierających azot. Może to być spowodowane trudnością w generowaniu nietrywialnych deskryptorów, które poprawnie opisywałyby nieistniejące cząsteczki.

2.3 *Model 3: podejście przy użyciu uczenia częściowo nadzorowanego*

Aby uwzględnić ograniczenia obu poprzednich modeli, zaprojektowaliśmy OC-MF-Score, który oparliśmy na technice klasyfikacji jednej klasy. Pozwoliło nam to ominąć trudność w stworzeniu odpowiednich deskryptorów nieistniejących cząsteczek. Tutaj użyliśmy modelu jednoklasowego SVM (OCSVM) [53] z jądrem radialnym (RBF). Co więcej, jako deskryptorów użyliśmy metody ECFP4 [54], która ma przewagę nad poprzednio użytymi motywami, ponieważ pozwala zakodować wszystkie możliwe fragmety strutury cząsteczki.

ECFP4 jest metodą tworzenia numerycznej reprezentacji struktury chemicznej cząsteczki poprzez enumerację podgrafów drzewa przeszukiwania wszerz a następnie ich haszowanie. OCSVM to rozszerzenie SVM przeznaczone do wykrywania wartości odstających. Zaprojektowany został jako model SVM, który rozdziela obserwacje jednej klasy na dwa zbiory: obserwacje pochodzące z tej klasy i wartości odstające.

OC-MF-Score został wytrenowany jako model OCSVM na reprezentatywnym podzbiorze bazy danych ZINC15 składającym się z 100 000 cząsteczek zakodowanych za pomocą ECFP4 o długości 128.

Zweryfikowaliśmy wyniki podobnie do MF-Score, porównując z SAscore, SCScore na 40 cząsteczkach uzyskanych z oryginalnej pracy SAscore. OC-MF-Score silnie koreluje z wynikami SAscore i SCSCore, co sugeruje, że wyniki OC-MF-Score są zgodne z oczekiwaniami. Porównywanie z pojedynczym wynikiem może jednak skutkować przeuczeniem do zbioru weryfikacyjnego. Dlatego predykcje OC-MF-Score mogą być częściowo niewiarygodne. Z tego powodu stworzyliśmy zbiór porównawczy do weryfikowania poprawności modeli do predykcji syntezowalności.

W celu poprawy weryfikacji modeli do oceny syntezowalności, stworzyliśmy platformę testową do ich porównywania [55], w której przeanalizowaliśmy OC-MF-Score, SAScore, SCSCore, RAscore i SYBA. W tym celu przygotowaliśmy bazę 49 związków, z których 44 mają udokumentowaną syntezę. Nasza baza danych zawiera szerokie spektrum związków, począwszy od łatwo syntetyzowalnych, przez takie o złożonej syntezie i niskiej wydajności, aż po niesyntetyzowalne. Cząsteczki z tej bazy danych stanowiły zbiór danych wejściowych do narzędzia AiZynthFinder.

Oceniliśmy, czy modele do predykcji syntezowalności mogą przewidywać wyniki retrosyntetycznego planowania. Dla wszystkich związków z bazy przeprowadziliśmy planowanie retrosyntetyczne przy użyciu narzędzia AiZynthFinder z domyślnymi parametrami. Ścieżki syntezy zostały znalezione dla 22 związków. Aby znaleźć optymalne wartości predykcji modeli, które odróżniają cząsteczki syntezowalne od tych, których nie można zsyntetyzować, przeanalizowaliśmy ich krzywe ROC. Zmierzyliśmy również jakość wyników, obliczając dokładność predykcji oraz pole powierzchni pod krzywą ROC (AUC). Zarówno dla AUC, jak i dokładności SAscore i RAscore osiągnęły wysokie wyniki. Natomiast, zarówno dla SCScore, jak i SYBA wyniki były gorsze o około 20 punktów procentowych. Niestety najgorsze wyniki osiągnął OC-MF-Score.

Sprawdziliśmy również, czy modele do predykcji syntezowalności mogą modelować złożoność planowania retrosyntetycznego. Z tego powodu przeanalizowaliśmy drzewa wyszukiwania algorytmu AiZynthFinder dla cząsteczek z bazy danych. Obliczyliśmy korelację rangową Spearmana [52] między predykcjami modeli a parametrami złożoności drzew przeszukiwań, takimi jak szerokość drzewa czy liczba węzłów. Wszystkie RAscore, SAscore, OC-MF-Score i SCScore korelują ujemnie ze wszystkimi parametrami złożoności. SYBA natomiast nie koreluje z żadnym z parametrów złożoności. Analogicznie jak wcześniej najlepiej wypadły RAscore i SAscore, najsilniejszą ujemną korelację zaobserwowano dla liczby węzłów.

Ponadto sprawdziliśmy, czy modele do predykcji syntezowalności nadają się jako heurystyka do priorytetyzowania kolejnych kroków przeszukiwania. W tym celu podzieliliśmy wszystkie węzły na trzy grupy: rozwiązane, nierozwiązane i wewnętrzne. Węzły rozwiązane odpowiadają kompletnym rozwiązaniom, tj. ich wszystkie cząsteczki dostępne są w sprzedaży. Nierozwiązane węzły odpowiadają częściowym rozwiązaniom niewykonalnej ścieżki syntezy. Pozostałe węzły są wewnętrzne, tj. są to węzły posiadające ścieżkę do rozwiązanego liścia. Aby sprawdzić, czy modele do predykcji syntezowalności odpowiednio nadają priorytety węzłom, przeanalizowaliśmy, czy modele te odróżniają węzły wewnętrzne od nierozwiązanych. Rozważaliśmy pary węzłów związane z pojedynczym etapem reakcji. Przeanalizowaliśmy dwie konfiguracje: i) węzły rodzeństwa wewnętrzne i nierozwiązane z wewnętrznym rodzicem oraz ii) wewnętrzny rodzic od nierozwiązanego dziecka. Użyliśmy testu t dla jednej próby [56] dla różnic predykcji par węzłów. Zaobserwowaliśmy, że praktycznie wszystkie

modele poprawnie rozróżniają węzły wewnętrzne od nierozwiązanych oraz węzły rozwiązane od nierozwiązanych.

Na koniec sprawdziliśmy, czy zmodyfikowany wybór węzłów uwzględniający modele do predykcji syntezowalności węzłów, może przyspieszyć planowanie retrosyntetyczne. W tym celu zastąpiliśmy ułamek we wzorze priorytetyzującym kolejne częściowe wyniki AizynthFindera wartością predykcji każdego z modeli. Zmiana ta jednak nie poprawiła znacząco żadnego parametru złożoności drzewa poszukiwań.

Podsumowując, przeanalizowaliśmy, czy modele do predykcji syntezowalności mogą skutecznie przyspieszyć proces retrosyntezy i potwierdziliśmy, że mogą one w większości przypadków dobrze odróżnić cząsteczki syntezowalne od niesyntezowalnych. Jednak projektowanie algorytmów retrosyntetycznego planowania jest wymagające i potrzebuje ciągłego doskonalenia w celu uzyskania szybszego czasu działania i dokładniejszych wyników. Naszym zdaniem modele hybrydowe oparte na uczeniu maszynowym i ludzkiej intuicji ze starannie opracowanymi algorytmami planowania mogą nadal skutecznie zwiększać swoją skuteczność.

BIBLIOGRAFIA

[1] Kim Esbensen i Paul Geladi. "The Start and Early History of Chemometrics: Selected Interviews. Part 2". W: *Journal of Chemometrics* 4.6 (1990), s. 389–412.

[2] E. J. Corey i W. Todd Wipke. "Computer-Assisted Design of Complex Organic Syntheses". W: *Science* (paź. 1969).

[3] E. J. Corey, Richard D. Cramer i W. Jeffrey Howe. "Computer-Assisted Synthetic Analysis for Complex Molecules. Methods and Procedures for Machine Generation of Synthetic Intermediates". W: *Journal of the American Chemical Society* 94.2 (sty. 1972), s. 440–459.

[4] "Chemoinformatics Theory". W: *Chemoinformatics: Theory, Practice, & Products*. Red. B. A. Bunin i in. Dordrecht: Springer Netherlands, 2007, s. 1–49. ISBN: 978-1-4020-5001-5.

[5] Thomas Engel. "Basic Overview of Chemoinformatics". W: *Journal of Chemical Information and Modeling* 46.6 (list. 2006), s. 2267–2277.

[6] William J. Wiswesser. "How the WLN Began in 1949 and How It Might Be in 1999". W: *Journal of Chemical Information and Computer Sciences* 22.2 (maj 1982), s. 88–93.

[7] R. L. DesJarlais i in. "Structure-Based Design of Nonpeptide Inhibitors Specific for the Human Immunodeficiency Virus 1 Protease". W: *Proceedings of the National Academy of Sciences of the United States of America* 87.17 (wrz. 1990), s. 6644–6648.

[8] Richard G. Brereton. "A Short History of Chemometrics: A Personal View". W: *Journal of Chemometrics* 28.10 (2014), s. 749–760.

[9] Peter Willett. "Chemoinformatics: A History". W: *WIREs Computational Molecular Science* 1.1 (2011), s. 46–56.

[10] Kristin B. Runkle i in. “Inhibition of DHHC20-Mediated EGFR Palmitoylation Creates a Dependence on EGFR Signaling”. W: *Molecular Cell* 62.3 (maj 2016), s. 385–396.

[11] Bin Zhou i in. “LC-MS-based Metabolomics”. W: 8.2 (2012), s. 470–481.

[12] Lloyd R. Snyder, Joseph J. Kirkland i John W. Dolan. *Introduction to Modern Liquid Chromatography*. John Wiley & Sons, Inc., list. 2009.

[13] Eva Lange i in. “A Geometric Approach for the Alignment of Liquid Chromatography—Mass Spectrometry Data”. W: *Bioinformatics (Oxford, England)* 23.13 (lip. 2007), s. i273–i281.

[14] Matthew The i Lukas Käll. “Focus on the Spectra That Matter by Clustering of Quantification Data in Shotgun Proteomics”. W: *Nature Communications* 11.1 (czer. 2020), s. 3234.

[15] R. Smith, D. Ventura i J. T. Prince. “LC-MS Alignment in Theory and Practice: A Comprehensive Algorithmic Review”. W: *Briefings in Bioinformatics* 16.1 (list. 2013), s. 104–117.

[16] Seongho Kim i Xiang Zhang. “Comparative Analysis of Mass Spectral Similarity Measures on Peak Alignment for Comprehensive Two-Dimensional Gas Chromatography Mass Spectrometry”. W: *Computational and Mathematical Methods in Medicine* 2013 (2013), s. 1–12.

[17] Grzegorz Skoraczyński, Anna Gambin i Błażej Miasojedow. “Alignstein: Optimal transport for improved LC-MS retention time alignment”. W: *GigaScience* 11.giac101 (list. 2022).

[18] Florian Huber i in. “Spec2Vec: Improved Mass Spectral Similarity Scoring through Learning of Structural Relationships”. W: *PLOS Computational Biology* 17.2 (lut. 2021). Red. Lars Juhl Jensen, e1008724.

[19] L. V. Kantorovich. “Mathematical Methods of Organizing and Planning Production”. W: *Management Science* 6.4 (lip. 1960), s. 366–422.

[20] Gabriel Peyré i Marco Cuturi. “Computational Optimal Transport: With Applications to Data Science”. W: *Foundations and Trends® in Machine Learning* 11.5-6 (2019), s. 355–607.

[21] Lénaïc Chizat i in. “Scaling Algorithms for Unbalanced Optimal Transport Problems”. W: *Mathematics of Computation* 87.314 (lut. 2018), s. 2563–2609.

[22] Paul Knopp i Richard Sinkhorn. “Concerning Nonnegative Matrices and Doubly Stochastic Matrices.” W: *Pacific Journal of Mathematics* 21.2 (1967), s. 343–348.

[23] Michał Aleksander Ciach i in. “Masserstein: Linear Regression of Mass Spectra by Optimal Transport”. W: *Rapid Communications in Mass Spectrometry* (wrz. 2020), e8956.

[24] Eva Lange i in. “Critical Assessment of Alignment Procedures for LC-MS Proteomics and Metabolomics Measurements”. W: *BMC Bioinformatics* 9.1 (wrz. 2008).

[25] Björn Voss i in. “SIMA: Simultaneous Multiple Alignment of LC/MS Peak Lists”. W: *Bioinformatics (Oxford, England)* 27.7 (lut. 2011), s. 987–993.

[26] Tomáš Pluskal i in. “MZmine 2: Modular Framework for Processing, Visualizing, and Analyzing Mass Spectrometry-Based Molecular Profile Data”. W: *BMC Bioinformatics* 11.1 (lip. 2010).

[27] R. Ballardini i in. “MassUntangler: A Novel Alignment Tool for Label-Free Liquid Chromatography–Mass Spectrometry Proteomic Data”. W: *Journal of Chromatography A* 1218.49 (grud. 2011), s. 8859–8868.

[28] Joe Wandy i in. “Incorporating Peak Grouping Information for Alignment of Multiple Liquid Chromatography-Mass Spectrometry Datasets”. W: *Bioinformatics (Oxford, England)* 31.12 (lut. 2015), s. 1999–2006.

[29] Audrey Barranger i in. “Antagonistic Interactions between Benzo[a]Pyrene and Fullerene (C60) in Toxicological Response of Marine Mussels”. W: *Nanomaterials* 9.7 (lip. 2019), s. 987.

[30] S. Hanessian, Jonathan Franco i Benoit Larouche. “The Psychobiological Basis of Heuristic Synthesis Planning - Man, Machine and the Chiron Approach”. W: *Pure and Applied Chemistry* 62.10 (sty. 1990), s. 1887–1910.

[31] Wolf-Dietrich Ihlenfeldt i Johann Gasteiger. “Computer-Assisted Planning of Organic Syntheses: The Second Generation of Programs”. W: *Angewandte Chemie International Edition in English* 34.23-24 (1996), s. 2613–2633.

[32] Ivar Ugi i in. “Computer-Assisted Solution of Chemical Problems—The Historical Development and the Present State of the Art of a New Discipline of Chemistry”. W: *Angewandte Chemie International Edition in English* 32.2 (1993), s. 201–227.

[33] Sara Szymkuć i in. “Computer-Assisted Synthetic Planning: The End of the Beginning”. W: *Angewandte Chemie International Edition* 55.20 (2016), s. 5904–5937.

[34] Philippe Schwaller i in. “Predicting Retrosynthetic Pathways Using Transformer-Based Models and a Hyper-Graph Exploration Strategy”. W: *Chemical Science* 11.12 (mar. 2020), s. 3316–3325.

[35] Samuel Genheden i in. “AiZynthFinder: A Fast, Robust and Flexible Open-Source Software for Retrosynthetic Planning”. W: *Journal of Cheminformatics* 12.1 (list. 2020), s. 70.

[36] Peter Ertl i Ansgar Schuffenhauer. “Estimation of Synthetic Accessibility Score of Drug-like Molecules Based on Molecular Complexity and Fragment Contributions”. W: *Journal of Cheminformatics* 1.1 (grud. 2009), s. 8.

[37] Milan Voršilák i in. “SYBA: Bayesian Estimation of Synthetic Accessibility of Organic Compounds”. W: *Journal of Cheminformatics* 12.1 (maj 2020), s. 35.

[38] Connor W. Coley i in. “SCScore: Synthetic Complexity Learned from a Reaction Corpus”. W: *Journal of Chemical Information and Modeling* 58.2 (lut. 2018), s. 252–261.

[39] Amol Thakkar i in. “Retrosynthetic Accessibility Score (RAscore) – Rapid Machine Learned Synthesizability Classification from AI Driven Retrosynthetic Planning”. W: *Chemical Science* 12.9 (mar. 2021), s. 3339–3349.

[40] David Weininger. “SMILES, a Chemical Language and Information System. 1. Introduction to Methodology and Encoding Rules”. W: *Journal of Chemical Information and Modeling* 28.1 (lut. 1988), s. 31–36.

[41] David Sherrington i Scott Kirkpatrick. “Solvable Model of a Spin-Glass”. W: *Physical Review Letters* 35.26 (grud. 1975), s. 1792–1796.

[42] Norman L. Johnson, Adrienne W. Kemp i Samuel Kotz. *Univariate Discrete Distributions*. John Wiley & Sons, Inc., sty. 2005.

[43] Robert Tibshirani. “Regression Shrinkage and Selection via the Lasso”. W: *Journal of the Royal Statistical Society. Series B (Methodological)* 58.1 (1996), s. 267–288.

[44] Neal Parikh. “Proximal Algorithms”. W: *Foundations and Trends® in Optimization* 1.3 (2014), s. 127–239.

[45] Blazej Miasojedow i Wojciech Rejchel. “Sparse Estimation in Ising Model via Penalized Monte Carlo Methods”. W: *Journal of Machine Learning Research* 19.75 (2018), s. 1–26.

[46] E. Ising. “Beitrag Zur Theorie Des Ferromagnetismus”. W: *Zeitschrift fur Physik* 31 (lut. 1925), s. 253–258.

[47] Stuart Geman i Donald Geman. “Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images”. W: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-6.6 (list. 1984), s. 721–741.

[48] Nicholas Metropolis i in. “Equation of State Calculations by Fast Computing Machines”. W: *The Journal of Chemical Physics* 21.6 (czer. 1953), s. 1087–1092.

[49] Teague Sterling i John J. Irwin. “ZINC 15 – Ligand Discovery for Everyone”. W: *Journal of Chemical Information and Modeling* 55.11 (list. 2015), s. 2324–2337.

[50] B. Efron. “Bootstrap Methods: Another Look at the Jackknife”. W: *The Annals of Statistics* 7.1 (sty. 1979), s. 1–26.

[51] Jerome H. Friedman. “Greedy Function Approximation: A Gradient Boosting Machine.” W: *The Annals of Statistics* 29.5 (paź. 2001), s. 1189–1232.

[52] C. Spearman. “The Proof and Measurement of Association between Two Things”. W: *The American Journal of Psychology* 15.1 (1904), s. 72–101.

[53] Bernhard Schölkopf i in. “Support Vector Method for Novelty Detection”. W: *Advances in Neural Information Processing Systems*. T. 12. MIT Press, 1999.

[54] David Rogers i Mathew Hahn. “Extended-Connectivity Fingerprints”. W: *Journal of Chemical Information and Modeling* 50.5 (maj 2010), s. 742–754.

[55] Grzegorz Skoraczyński i in. “Critical Assessment of Synthetic Accessibility Scores in Computer-Assisted Synthesis Planning.” W: (list. 2022).

[56] Student. “The Probable Error of a Mean”. W: *Biometrika* 6.1 (1908), s. 1–25.