



Structure-property modeling with advanced machine learning techniques

Dmitry Zankov

► To cite this version:

Dmitry Zankov. Structure-property modeling with advanced machine learning techniques. Cheminformatics. Université de Strasbourg, 2023. English. NNT : 2023STRAF001 . tel-04116204

HAL Id: tel-04116204

<https://theses.hal.science/tel-04116204>

Submitted on 3 Jun 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

ÉCOLE DOCTORALE DES SCIENCES CHIMIQUES

Chimie de la matière complexe – UMR 7140

THÈSE présentée par :

Dmitry ZANKOV

soutenue le : **13 janvier 2023**

pour obtenir le grade de : **Docteur de l'Université de Strasbourg**

Discipline/ Spécialité : **Chimie / Chémoinformatique**

**Modélisation structure-propriété avec
des techniques avancées
d'apprentissage automatique**

THÈSE dirigée par :

M. VARNEK Alexandre

Professeur, Université de Strasbourg

RAPPORTEURS :

Mme. CAMPROUX Anne-Claude

Professeur, Université Paris 7

M. FIORUCCI Sébastien

Docteur, Université Côte d'Azur

AUTRES MEMBRES DU JURY :

M. MADZHIDOV Timur

Docteur, Elsevier Ltd.

M. ROGNAN Didier

Directeur de Recherche au CNRS

Modélisation structure-propriété avec des techniques avancées d'apprentissage automatique

Résumé

Cette thèse est consacrée au développement de techniques avancées d'apprentissage automatique pour la modélisation des propriétés des molécules et des réactions. Le couplage de la méthode d'apprentissage automatique multi-instances (MIL) avec les descripteurs 3D pharmacophoriques a permis de construire des modèles prédictifs prenant en compte l'ensemble des conformations moléculaires. Cette approche 3D ne nécessite pas de sélection et d'alignement de conformères et a été validée dans les études de (i) la bioactivité des composés et (ii) l'énantiosélectivité des catalyseurs organiques chiraux. Dans de nombreux cas, les modèles MIL multi-conformationnelles 3D ont surpassé les approches classiques impliquant des descripteurs 2D populaires. Dans la deuxième partie, un concept d'apprentissage automatique conjugué a été introduit et appliqué à la modélisation des caractéristiques thermodynamiques et cinétiques des réactions chimiques. L'apprentissage automatique conjugué intègre des équations fondamentales avec des algorithmes d'apprentissage automatique, ce qui le distingue de l'apprentissage multitâche traditionnel ne capturant que la relation statistique entre les tâches

Mots-clés : apprentissage multi-instances, modèles conjugués

Résumé en Anglais

This Ph.D. thesis is devoted to the development of advanced machine learning techniques for the modeling of properties of molecules and reactions. Coupling the Multi-Instance machine Learning (MIL) method with the pharmacophoric 3D descriptors enabled the construction of predictive models accounting for an ensemble of molecular conformations. This 3D approach does not require the selection and alignment of conformers and was validated in the case studies of (i) the bioactivity of compounds and (ii) the enantioselectivity of chiral organic catalysts. In many cases, 3D multi-conformation MIL models overperformed classical approaches involving popular 2D descriptors. In the second part, a concept of conjugated machine learning was introduced and applied to the modeling of thermodynamic and kinetic characteristics of reactions. Conjugated machine learning integrates fundamental equations with machine learning algorithms, which distinguishes it from traditional multi-task learning capturing only the statistical relationship between the tasks.

Keywords: multi-instance learning, conjugated machine learning

Acknowledgments

I would like to express my deepest gratitude to my academic advisors Prof. Alexandre Varnek and Dr. Timur Madzhidov for their guidance throughout this project.

Special thanks to Dr. Igor Baskin and Dr. Pavel Polishchuk. Dr. Baskin first proposed the idea of conjugated machine learning. Dr. Polishchuk initiated the project on multi-instance machine learning and proposed the first ideas and implementations.

I am also thankful to my colleagues who contributed to this Ph.D. project: Dr. Mariia Matveieva and Alexandra Nikonenko and to Dr. Olga Klimchuk, Dr. Gilles Marcou, Dr. Dragos Horvath, and Dr. Fanny Bonachera for their help with scientific, technical and administrative issues. Also, thanks to committee members, Prof. Anne-Claude Camproux and Dr. Sébastien Fiorucci.

I also would like to acknowledge the French Embassy in Russia and the University of Strasbourg for the financial support of my research.

List of papers

1. Multi-Instance Learning Approach to Predictive Modeling of Catalysts Enantioselectivity

D. Zankov, P. Polishchuk, T. Madzhidov, A. Varnek

Synlett 32.18 (2021): 1833-1836.

2. Multiple Conformer Descriptors for QSAR Modeling

A. Nikonenko, **D. Zankov**, I. Baskin, T. Madzhidov, P. Polishchuk

Molecular Informatics 40.11 (2021): 2060030.

3. QSAR Modeling Based on Conformation Ensembles Using a Multi-Instance Learning Approach

D. Zankov, M. Matveieva, A. Nikonenko, R. Nugmanov, I. Baskin, A. Varnek, P. Polishchuk, T. Madzhidov

Journal of Chemical Information and Modeling 61.10 (2021): 4913-4923.

4. Multi-instance Learning for Structure-Activity Modeling for Molecular Properties

D. Zankov, P. Polishchuk, A. Nikonenko, M. Shevelev T. Madzhidov

In International Conference on Analysis of Images, Social Networks, and Texts, pp. 62-71. Springer, Cham, 2019.

5. Conjugated Quantitative Structure-Property Relationship Models: Application to Simultaneous Prediction of Tautomeric Equilibrium Constants and Acidity of Molecules

D. Zankov, T. Madzhidov, A. Rakhimbekova, R. Nugmanov, M. Kazymova, I. Baskin, A. Varnek

Journal of Chemical Information and Modeling 59.11 (2019): 4569-4576.

Contents

Résumé en français	6
Introduction	25
Part 1. Multi-instance machine learning in chemoinformatics and bioinformatics.....	27
1.1 Introduction.....	27
1.2 Origins of multi-instance learning	29
1.3 Multi-instance learning algorithms	30
1.4 Multi-instance learning applications.....	38
1.5 Toolkits and software	49
Part 2. 3D structure-property modeling with multi-instance machine learning	50
2.1 Methodological developments.....	50
2.2 Multiple conformer descriptors for QSAR modeling	61
2.3 Modeling of compounds bioactivity with conformation ensembles.....	74
2.4 Modeling of catalysts enantioselectivity with conformation ensembles	87
Part 3. Modeling reaction characteristics with conjugated machine learning	108
3.1 Methodological developments.....	108
3.2 Modeling of tautomeric constant	113
3.3 Modeling of Arrhenius equation parameters	123
3.4 Modeling of selectivity constant of competing reactions	137
Conclusion	142
List of abbreviations	144
References.....	146

Résumé en français

Introduction

Dans une modélisation « structure-propriété » classique, une molécule est encodée par des descripteurs numériques qui sont corrélés avec la propriété cible à l'aide d'algorithmes d'apprentissage automatique. Une fois une corrélation acceptable établie, le modèle obtenu peut être utilisé pour prédire les propriétés de nouvelles molécules qui n'ont pas encore été testées expérimentalement. Il s'agit d'un protocole bien établi hérité de l'apprentissage automatique classique. Le but de cette thèse de doctorat consiste à faire progresser la méthodologie de modélisation en mettant en œuvre de nouvelles techniques d'apprentissage automatique qui capturent mieux la complexité des systèmes chimiques (i) en tenant compte de conformations d'une molécule donnée, (ii) en reliant les modèles statistiques à des équations cinétique et thermodynamique.

Partie I. Modélisation structure-propriété avec apprentissage automatique multi-instance

Par soucis de simplification, la modélisation « structure-propriété » à partir de structures 3D concerne la représentation d'une molécule par un conformère unique (généralement le plus faible énergie) encodé, à son tour, par un ensemble unique de descripteurs 3D. Cette pratique ignore donc la nature dynamique des molécules – l'existence de conformères multiples – qui, par conséquent, n'est pas capturée dans les algorithmes d'apprentissage automatique. Cette approximation n'est pas nécessaire en utilisant l'Apprentissage Automatique Multi-Instance (Multi-Instance Machine Learning (MIL)) [1] (Figure I-1). Dans ce formalisme, un objet (molécule) est composé d'un ensemble d'entités/instances qui le définissent simultanément (conformations, tautomères, stéréoisomères, états de protonation, etc.). Ici, chaque conformère est encodé par un vecteur de descripteurs moléculaire 3D. Ainsi, les algorithmes MIL établissent une corrélation entre ces ensembles et la valeur de la propriété à modéliser.

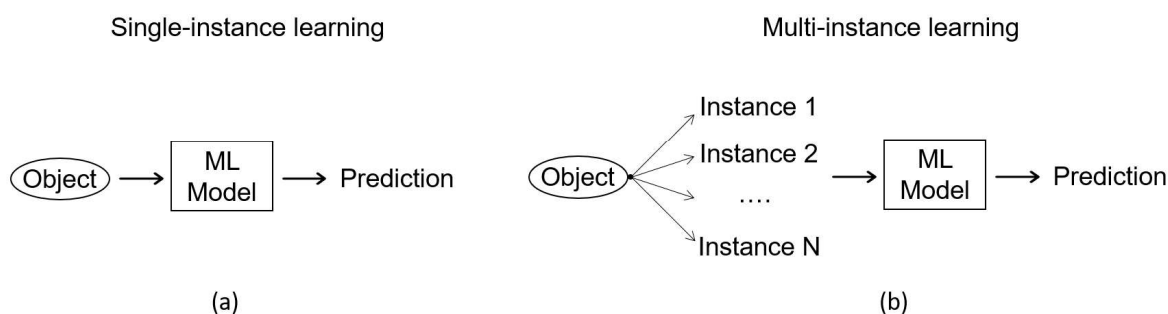


Figure I-1. Approche d'apprentissage par instance unique vs. approche d'apprentissage multi-instances.

Il a été démontré [2] que la prise en compte de plusieurs conformères à faible énergie MIL apporte une solution au problème de la modélisation QSAR en 3D : la sélection de conformères pertinents responsables de l'activité cible. L'approche 3D développée a été appliquée pour construire des modèles prédictifs de la bioactivité des molécules et de l'énantiosélectivité des catalyseurs organiques chiraux en synthèse asymétrique.

I.I Modélisation d'activités biologiques à l'aide d'ensembles conformationnels

Une analyse comparative à grande échelle des approches de modélisation 2D et 3D a été réalisée à l'aide de 175 jeux de données extraits de la base de données ChEMBL-23. Chaque ensemble de données contenait un ensemble de molécules reliées à une constante de liaison expérimentale pKi (bioactivité) mesurée par rapport à une cible particulière. La taille des ensembles de données variait de plusieurs centaines à plusieurs milliers de composés. Chaque ensemble de données a été divisé au hasard en ensembles d'entraînement (pour la construction du modèle) et de test (pour la validation du modèle) dans une proportion de 80/20.

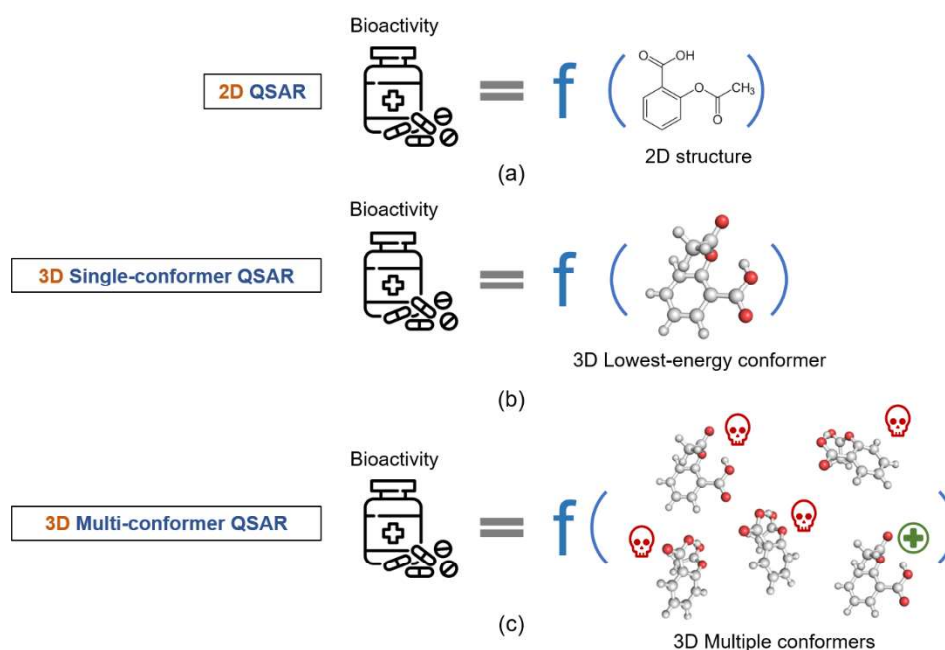


Figure I.I-1. Les trois approches principales considérées dans cette étude pour modéliser la bioactivité : (a) L'approche 2D-QSAR traditionnelle, basée sur des descripteurs 2D, (b) l'approche 3D-QSAR, basée sur un conformère encodé par des descripteurs 3D et (c) la nouvelle approche QSAR multi-instances basée sur de multiples conformères encodés avec des descripteurs 3D et des algorithmes d'apprentissage machine multi-instances.

Trois principales approches de modélisation ont été comparées (Figure I.I-1): une approche classique à instance unique basée sur des descripteurs moléculaires 2D populaires (modèle 2D); une approche 3D mono-instance basée sur les conformères de plus faible énergie (modèle 3D

mono-conformère); et une approche multi-instancs 3D basée sur plusieurs conformères générés (modèle multi-conformères 3D). Des signatures pharmacophores 3D [3] (package *pmapper*) ont été utilisées comme descripteurs 3D. Chaque conformère était représenté par un ensemble de caractéristiques pharmacophoriques (donneur/accepteur de liaison H, centre de la charge positive/négative, hydrophobe et aromatique) déterminées en appliquant les définitions SMARTS correspondantes.

Tableau 1. Performances des modèles 2D et 3D construits sur 139 jeux de test issus de la ChEMBL-23 : valeurs moyennes et médianes du coefficient de détermination (R^2_{Test}).

	R^2_{Test} moyen	R^2_{Test} médian
Modèle 2D	0.39	0.45
Modèle 3D mono-conformère	-0.01	0.04
Modèle 3D multi-conformères	0.47	0.48

Tableau 2. Top-1 représente le nombre de jeux de données pour lesquels le modèle était le meilleur. Top-2 est le nombre de jeux de données où le modèle était en premier ou en second. Top-3 est le nombre de jeux de données où le modèle était en premier, en second ou en 3^e (nombre total de jeux de données où au moins un modèle a obtenu un $R^2_{\text{Test}} > 0.4$)

	Top-1	Top-2	Top-3
Modèle 2D	50	136	139
Modèle 3D mono-conformère	1	8	140
Modèle 3D multi-conformères	88	139	139

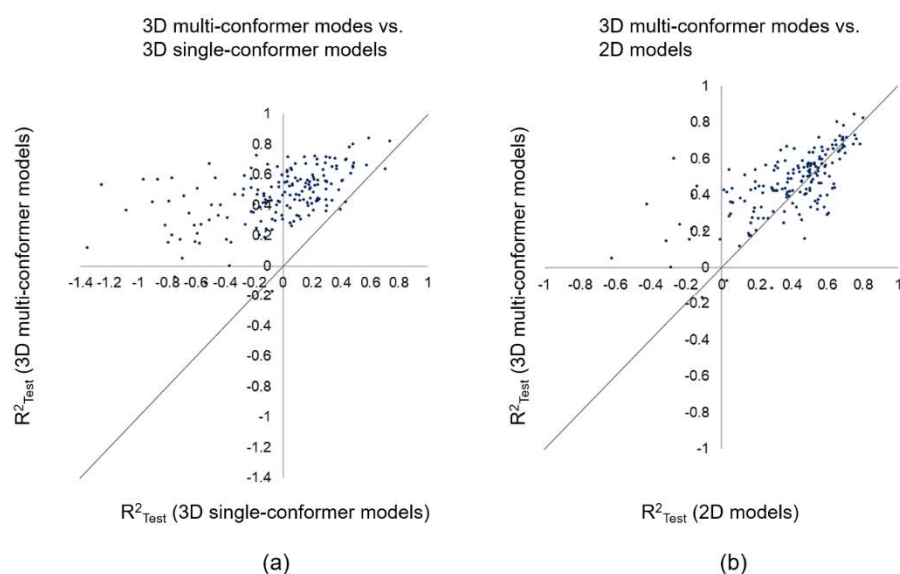


Figure I.I-2. Corrélation entre les valeurs de R^2_{Test} calculées pour les modèles 2D et 3D sur tous les 175 jeux de données.

Tous les quadruplets possibles de caractéristiques d'un conformère particulier ont été énumérés et le nombre de signatures de quadruplets de pharmacophore 3D identiques a été compté pour chaque conformère, ce qui a donné un vecteur de descripteur constitué de nombres entiers. Tous les modèles ont été construits avec un réseau de neurones entièrement connecté avec trois couches cachées de 256, 128 et 64 neurones intégrant une fonction d'activation ReLU.

Par souci de clarté, 36 ensembles de données "non modélisables" pour lesquels aucun des modèles 2D et 3D considérés n'avait un $R^2_{\text{test}} > 0.4$ ont été exclus et l'analyse a été effectuée sur la base des 139 ensembles de données restants. Le Table 3 présente le R^2_{Test} moyen (coefficient de détermination) des modèles 2D et 3D sur 139 ensembles de données filtrés. Le modèle 3D à un seul conformère a montré de mauvaises performances (R^2_{Test} moyen=-0.01), ce qui peut s'expliquer par le fait que le conformère de plus faible énergie pourrait différer considérablement du conformère bioactif responsable de la bioactivité observée. Les performances du modèle 3D augmentent considérablement (R^2_{Test} moyen=0.47) dès que tous les conformères générés disponibles sont inclus dans le modèle multi-conformères 3D, qui surpasse même légèrement les modèles 2D classiques (R^2_{Test} moyen=0.39). Le modèle multi-conformères 3D a démontré avoir le R^2_{Test} le plus élevé dans 63 % des ensembles de données (88 sur 139 ensembles de données) et le modèle 2D était le meilleur dans 36 % des ensembles de données (50 sur 139 ensembles de données) (Figure I.I-2, Tableau 2).

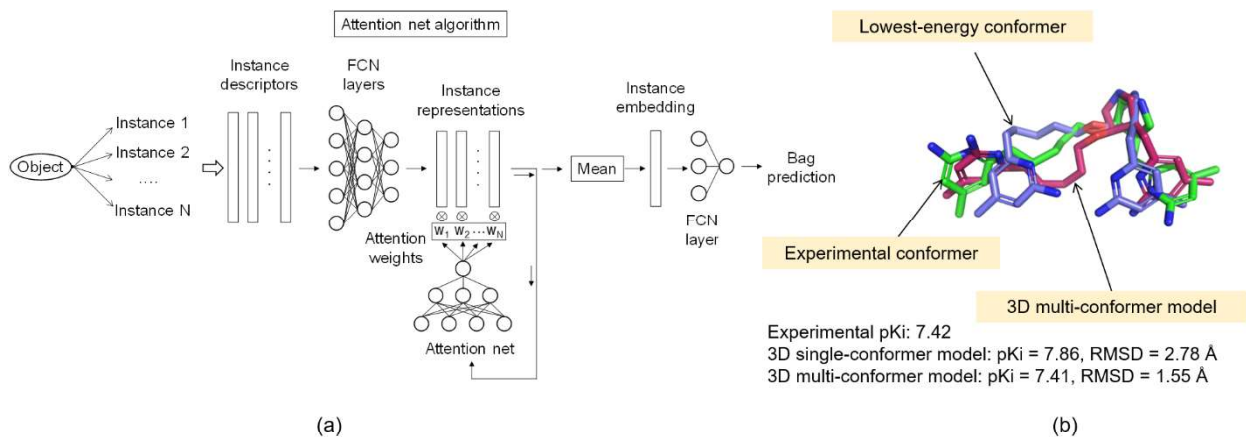


Figure I.I-3. (a) Architecture de réseau de neurones multi-instances avec mécanisme d'attention ; (b) Structures 3D du conformère extraites de la base de données PDB ainsi que la plus basse énergie et conformères prédits par l'algorithme MIL.

Les modèles 3D multi-conformères, construits avec le réseau de neurones avec mécanisme d'attention (Figure I.I-3a), permettent également d'identifier les conformères « bioactifs » les plus pertinents. Pour l'illustrer, des structures 3D de ligands pour la cible CHEMBL2820 ont été ex-

traites de complexes protéine-ligand à partir de la base de données PDB. Ces conformères expérimentaux ont été comparés avec (i) ceux de plus basse énergie (calculés avec les champs de force MMFF94s), (ii) une sélection aléatoire parmi tous les conformères générés par mécanique moléculaire, (iii) ceux prédits par le réseau de neurones utilisant un mécanisme d'attention et (iv) ceux obtenus avec le logiciel d'amarrage moléculaire AutoDock Vina. Les conformères de plus faible énergie et d'amarrage moléculaire s'alignent correctement aux conformères « bioactifs » pour 47% des molécules, ce qui est encore moins bon qu'une sélection aléatoire (60%). Pour sa part, le modèle multi-conformères 3D identifie correctement les conformères bioactifs pour 80 % des molécules (Figure I.I-4).

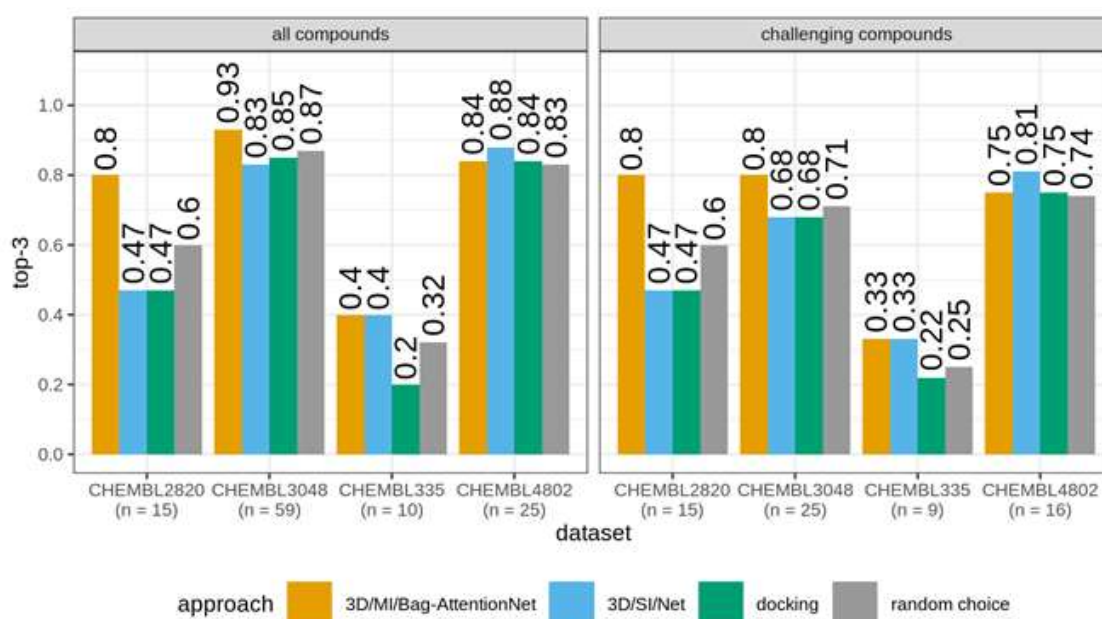


Figure I.I-4. Identification des conformères bioactifs parmi les composés du jeu de données de test pour 4 jeux de données (n représente le nombre de composés). 3D/MI/Bag-AttentionNet est un modèle 3D multi-conformères construit avec des algorithmes dits « Bag-Attention net » et 3D/SI/Net est un modèle mono-conformère. *Challenging compounds* représente un sous-jeu de données des composés du jeu de données de test présentant un RMSD moyen entre toutes les conformations générées et une conformation bioactive supérieur à 2 Å. Le R^2_{test} des modèles 3D/MI/Bag-AttentionNet est de 0.49, 0.52, 0.74 et 0.55 pour les jeux de données CHEMBL2820, CHEMBL3048, CHEMBL335 and CHEMBL4802, respectivement.

Pour conclure, l'approche de modélisation multi-conformères 3D unifiée surpasse systématiquement l'approche 3D à un seul conformère (la flexibilité conformationnelle est importante), (ii) surpasse souvent l'approche 2D (l'information 3D est importante), et (iii) identifie potentiellement les conformères "bioactifs".

I.II Modélisation de l'énantio-sélectivité d'un catalyseur à l'aide d'ensembles de conformères

La synthèse de composés énantiomériquement purs est un sujet majeur de la chimie organique moderne, en raison de l'importance pratique de ces substances, particulièrement pour la production des principes actifs de médicaments efficaces et sûrs. En 2021, B. List et D. McMillan ont reçu le prix Nobel pour le développement d'organocatalyseurs asymétriques - de petites molécules chirales capables de catalyser efficacement des réactions asymétriques. Des modèles d'apprentissage automatique prédisant l'énantio-sélectivité permettent un criblage rapide des bibliothèques de catalyseurs candidats, réduisant ainsi les ressources matérielles et humaines nécessaires pour découvrir de nouveaux catalyseurs. Ici, l'approche multi-conformères 3D a été appliquée pour développer des modèles de prédiction de l'énantio-sélectivité des catalyseurs chiraux. Chaque catalyseur était représenté par un ensemble de conformères encodés avec des triplets d'atomes 3D (Figure I.II-1a) à l'aide du package *pmapper*. Les transformations des réactions ont été transformées en un graphe condensé de réaction (CGR) encodé avec des descripteurs de fragments ISIDA 2D (Figure I.II-1b).

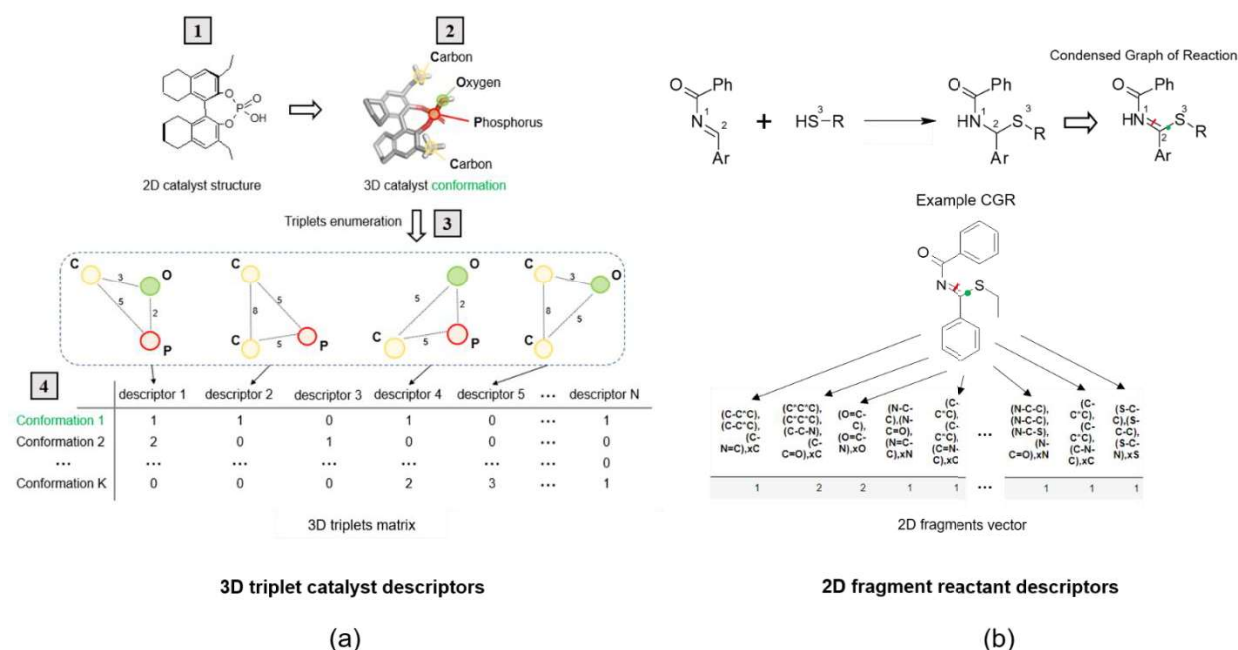


Figure I.II-1. (a) Préparation de triplets 3D de descripteurs pour un conformère de catalyseur donné impliquant les étapes suivantes : (1) génération de conformères pour une structure de catalyseur 2D ; (2) sélection d'un conformère de catalyseur 3D ; (3) énumération de triplets (la longueur de chaque arête correspond à la partie entière de la distance par paire associée en Å) ; et (4) calcul du nombre de triplets dans un conformère donné. (b) Addition de thiols aux imines et graphe condensé de réaction (CGR) associé. Le CGR est une pseudo-molécule décrite à la fois par des liaisons chimiques conventionnelles et des liaisons dynamiques décrivant des transformations chimiques. Des descripteurs fragmentaux sont générés pour le CGR.

L'application de quadruplets pharmacophoriques (H-donneur, H-accepteur, hydrophobes, ou atomes chargés positivement ou négativement - descripteurs *pmapper* par défaut) permet d'encoder la configuration stéréo d'une molécule, ce qui garantit que deux énantiomères d'une molécule ont deux vecteurs de descripteurs différents. Dans un article précédent [4], il a été démontré qu'une combinaison de quadruplets pharmacophoriques 3D et de MIL a conduit à générer des modèles précis sur des données de catalyseurs à base d'acide phosphorique (PAC) (Figure I.II-3a). Des expériences supplémentaires ont montré que les triplets d'atomes réduisent considérablement le nombre de descripteurs et conduisent à des performances encore meilleures que les quadruplets d'atomes.

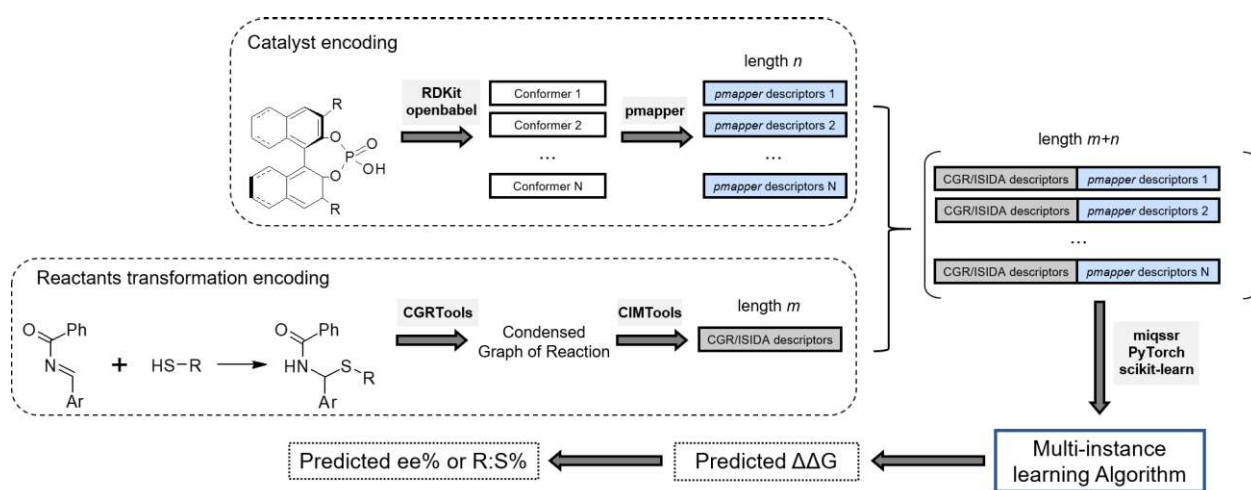


Figure I.II-2. Préparation des descripteurs qui encodent la combinaison des réactifs et des catalyseurs correspondants dans l'approche de modélisation 3D. Une transformation de réactif est encodée par m descripteurs fragmentaires CGR/ISIDA. Un catalyseur est représenté par N conformères, chacun encodé par n descripteurs 3D *pmapper*. La concaténation de m descripteurs 2D de réactifs et n descripteurs 3D de catalyseurs résulte en un ensemble de vecteurs de taille $(m+n)$. Au-dessus des flèches se trouvent les bibliothèques Python 3 utilisées pour exécuter chaque étape du protocole de modélisation.

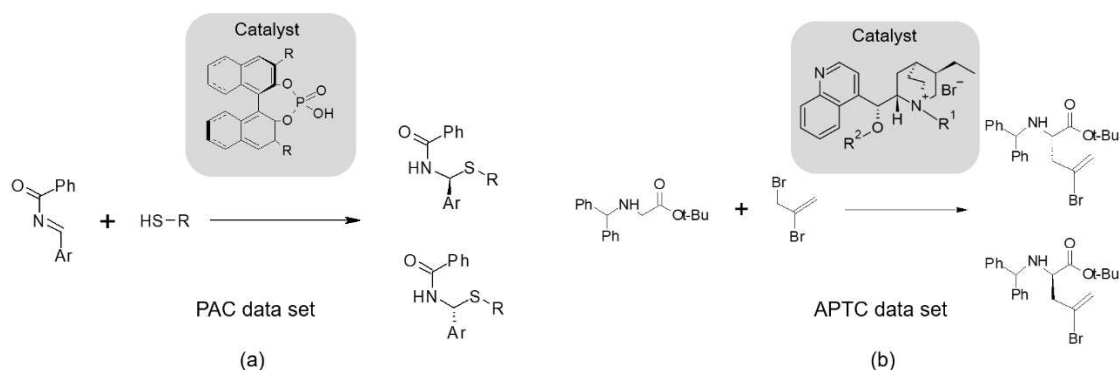


Figure I.II-3. Exemple de réactions publiées (jeux de données) prises en considération pour la modélisation dans cette étude : (a) l'addition asymétrique de thiols à des imines catalysée par des catalyseurs d'acide phosphorique chiral (PAC) et (b) l'alkylation asymétrique de bases de Schiff dérivées de la glycine catalysée par des sels d'ammonium à base d'alcaloïde de quinquina.

Des modèles 3D ont été générés avec des algorithmes MIL et comparés avec des modèles 2D classiques et d'autres approches de l'état de l'art. Une analyse comparative a été réalisée sur un nouvel ensemble de données [5] sur l'énantio-sélectivité des catalyseurs à base d'acide phosphorique (PAC) pour la réaction d'addition asymétrique de thiols aux imines (Figure I.II-3a). Cet ensemble de données concerne l'énantio-sélectivité de 43 catalyseurs pour 25 combinaisons de réactifs imine et thiol résultant en $43 \times 25 = 1075$ points de données. La concaténation de la réaction CGR et des descripteurs de catalyseur dans le processus d'entraînement produit des modèles qui peuvent être utilisés dans différents scénarios (Tableau 3) pour la prédiction de (a) l'énantio-sélectivité des réactions connues avec de nouveaux catalyseurs (certains catalyseurs étant exclus des données d'entraînement), (b) l'énantio-sélectivité de nouvelles réactions pour des catalyseurs connus (certaines réactions étant exclus des données d'entraînement), et (c) l'énantio-sélectivité de nouvelles réactions avec de nouveaux catalyseurs (certains catalyseurs et réactions étant exclus ensemble des données d'entraînement). Les modèles 2D et 3D générés ont également été comparés à l'approche 3D dépendante de la conformation publiée par Denmark [5] et l'approche 2D publiée par Glorius [6] et Miyao [7].

Tableau 3. Erreur Absolue Moyenne (MAE, kcal/mol) sur les prédictions de $\Delta\Delta G$, obtenue pour les ensembles de test générés à partir de l'ensemble de données des catalyseurs à base d'acide phosphorique (PAC).

	Modèle (descripteurs)	Test 1 Nouvelles réactions	Test 2 Nouveaux catalyseurs	Test 3 Nouvelles réactions et nouveaux catalyseurs
Approches développées	Modèle 2D (ISIDA fragments)	0.15	0.27	0.30
	Modèle 2D (CircuS fragments)	0.14	0.32	0.34
	Modèle 3D mono-conformère (Triplets d'atomes)	0.21	0.38	0.48
	Modèle 3D mono-conformère (Triplets d'atomes)	0.13	0.22	0.21
Approches alternatives publiées	Modèle 2D de Glorius (Empreintes MFFs)*	0.14	0.25	0.28
	Modèle 2D de Miyao (Mol2vec) **	0.13	0.34	0.40
	Modèle 2D de Miyao (ECFP6) **	0.14	0.22	0.21
	Modèle 3D mono-conformère (Dragon) **	0.14	0.42	0.47
	Modèle 3D mono-conformère (MOE) **	0.15	0.48	0.55
	Modèle 3D de Denmark modèle dépendant du con- formère (descripteurs ASO) ***	0.16	0.21	0.24

De façon générale, l'approche multi-conformationnelle 3D surpasse les modèles 2D (Tableau 3) dans la prédiction de l'énantio-sélectivité par de nouveaux catalyseurs – absents de l'ensemble

d'apprentissage. Ce fait indique que les modèles bénéficient de l'informations 3D pour la prédiction de l'énantio-sélectivité du catalyseur.

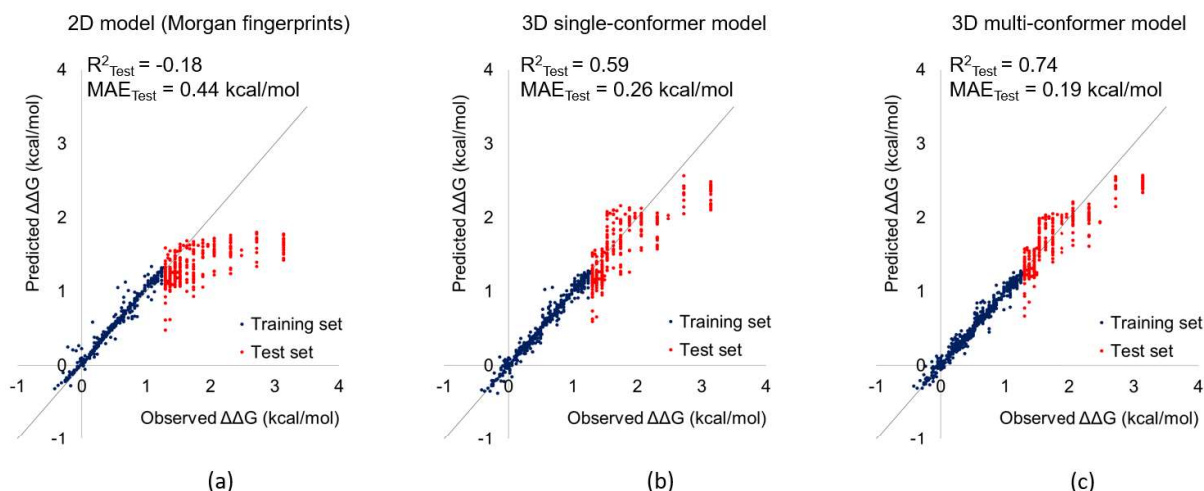


Figure I.II-4. Performance des modèles 2D et 3D (Erreur Absolue Moyenne, MAE) sur un ensemble de données de test comprenant des catalyseurs hautement sélectifs avec un excès énantiomérique > 80 %).

Afin d'examiner le potentiel des modèles à prédire les valeurs d'énantio-sélectivité au-delà de l'ensemble d'apprentissage, l'ensemble de données de 1075 réactions a été divisé en un ensemble d'apprentissage de réactions avec un *ee* (excès énantiomérique) inférieur à 80 % (718 réactions) et un ensemble de test de réactions hautement sélectives avec un *ee* supérieur à 80 % (357 réactions). Les résultats ont montré que le modèle 2D ne parvient pas à extrapoler l'*ee* des catalyseurs hautement sélectifs (Figure I.II-4) au-delà de l'ensemble d'apprentissage, alors que le modèle multi-conformères 3D fournit des prédictions très précises ($MAE_{Test}=0.19$ kcal/mol) et fonctionne encore mieux que l'approche de Denmark ($MAE_{Test}=0.33$ kcal/mol).

L'approche de modélisation 3D développée a également été appliquée à l'alkylation asymétrique de dérivés d'acides α -aminés catalysée par des catalyseurs à base d'alcaloïdes de quinquina (Figure I.II-3b) publiée par Melville [8]. Melville et ses collègues ont proposé une approche basée sur CoMFA 3D et ont rapporté un RMSE de 13,4 % sur les prédictions d'*ee* sur 18 catalyseurs de test. Le modèle 3D à conformère unique construit dans cette étude a obtenu des résultats considérablement moins bons avec un RMSE de 18 %. L'inclusion de plusieurs conformères de catalyseur dans le modèle multi-conformères 3D a considérablement amélioré la précision de la prédiction avec un RMSE de 8,8 % (Figure I.II-5c) et a surpassé les modèles 2D entraînés avec ISIDA (RMSE de 15,6 %, Figure I.II-5a) et Circus (18,5 %, Figure I.II-5b).

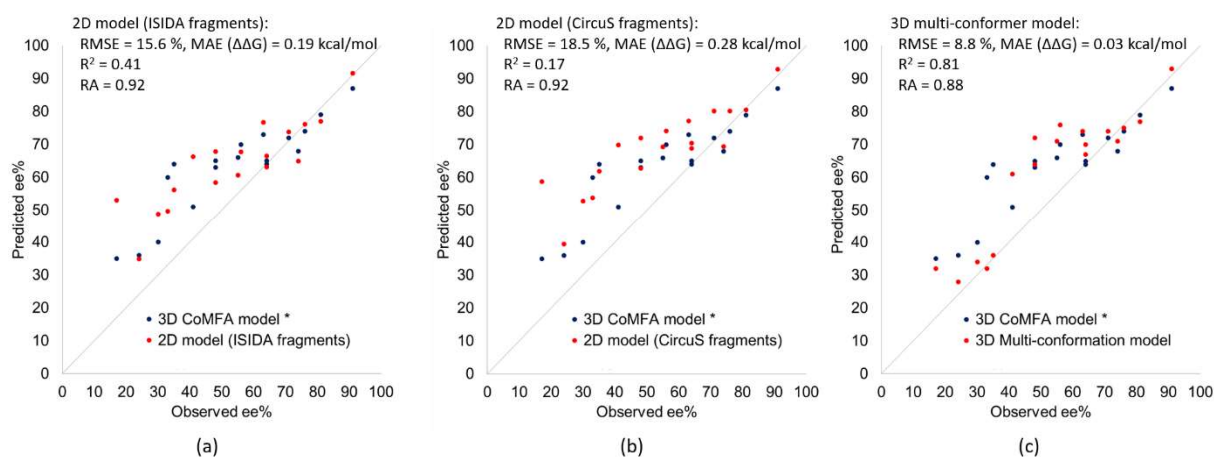


Figure I.II-5. Pourcentage de *ee* observe et prédit pour 18 catalyseurs de test à partir de l'ensemble de données APTC : (a) *modèle 3D-CoMFA vs modèle 2D (fragments ISIDA), (b) *modèle 3D-CoMFA vs modèle 2D (fragments CircuS) et (c) *Modèle 3D-CoMFA vs modèle 3D multi-conformères (triplets d'atomes). *Prédictions du modèle 3D CoMFA publiées par Melville [8].

Pour conclure, l'approche 3D proposée a des performances comparables ou supérieures aux autres approches 2D et 3D publiées, et présente plusieurs avantages. Le processus de construction de modèles 3D est entièrement automatisé et ne nécessite pas d'ajustement manuel (il ne nécessite pas de sélection et d'alignement de conformères), et plus important encore, l'approche développée est plus générale, c'est-à-dire applicable à différentes tâches avec une grande diversité de structures 3D.

Partie II. Modélisation structure-propriété avec apprentissage automatique conjugué

Les propriétés physicochimiques des molécules sont souvent liées par des relations cinétiques et thermodynamiques. Dans ce contexte, il est important de s'assurer de la validité de ces relations pour les propriétés prédites par les relations quantitatives structure-propriété (QSPR) individuelles correspondantes. Cependant, en raison de la nature statistique des modèles QSPR et de l'impossibilité de réduire les erreurs de prédiction à zéro, la réalisation de cet objectif est assez improbable même si chaque propriété associée est prédite avec une précision raisonnable. Pour résoudre ce problème, le concept de modèles QSPR *conjugués* a été récemment introduit [9], concept dans lequel les relations entre les propriétés sont explicitement intégrées dans l'algorithme d'apprentissage automatique (Figure II-1). L'apprentissage automatique conjugué a été mis en œuvre dans les algorithmes de Régression Ridge (Ridge Regression - RR) et de Réseaux de Neurones (Neural Network - NN) et appliqué (i) au problème de la prédiction simultanée de la constante tautomé-

rique binaire et de l'acidité de tautomères, (ii) à la prédiction des paramètres de l'équation d'Arrhenius pour des réactions de cycloaddition et (iii) constante de sélectivité des réactions concurrentes E2 and S_N2.

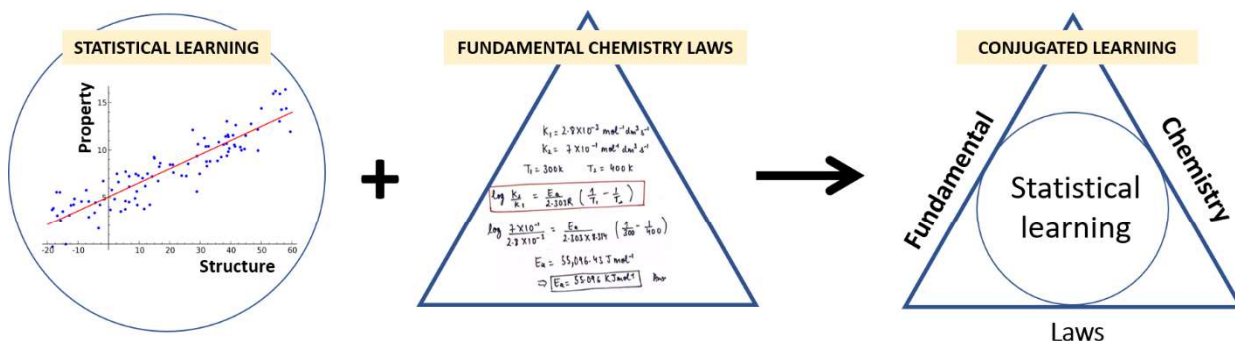


Figure II-1. Schéma conceptuel d'apprentissage machine conjugué pour le QSPR.

Dans la régression de crête ou Ridge Regression (RR), la caractéristique de réaction prédite y^{pred} est calculée en multipliant les descripteurs de réaction X par le vecteur des poids de régression w :

$$y^{pred} = Xw \quad \text{I.II-a}$$

Les poids de régression w sont estimés à l'aide de l'ensemble d'apprentissage et peuvent être calculés avec l'expression analytique :

$$w = (X^T X + \lambda I)^{-1} X^T y^{exp} \quad \text{I.II-b}$$

où X and y^{exp} forment l'ensemble d'apprentissage des réactions associées aux valeurs expérimentales de la caractéristique modélisée. L'hyperparamètre λ est un coefficient de régularisation contrôlant la complexité du modèle.

Les coefficients de régression w peuvent être trouvés en minimisant la fonction de perte, qui dans une régression de crête est la somme de l'erreur quadratique entre les variables observées y^{exp} et prédites $y^{pred} = Xw$ et le terme de régularisation :

$$Loss = \|y^{exp} - Xw\|^2 + \lambda \|w\|^2 \quad \text{I.II-c}$$

La conception de fonctions de perte spéciales intégrées aux relations cinétiques et thermodynamiques fondamentales est à la base des méthodes d'apprentissage conjugué. Le processus de construction de modèles conjugués peut être divisé en plusieurs étapes :

1) Intégrer l'équation de la caractéristique principale en construisant une fonction de perte basée sur une équation.

2) Combiner la fonction de perte basée sur une équation avec les fonctions de perte individuelles des caractéristiques connexes et les termes de régularisation de la complexité du modèle.

3) Calculer les poids de régression (paramètres) du modèle conjugué.

Les sections suivantes décrivent des exemples de construction de modèles conjugués pour les équations de trois caractéristiques : la constante tautomérique, la vitesse de réaction et la constante de sélectivité des réactions concurrentes.

II.I Modèles conjugués pour les équilibres tautomériques

La tautomérie est l'un des phénomènes les plus importants de la chimie organique et bioorganique. Cela a conduit au développement d'approches informatiques pour énumérer les tautomères possibles de composés chimiques, ainsi que pour évaluer la population de différentes formes tautomères à l'équilibre en solution.

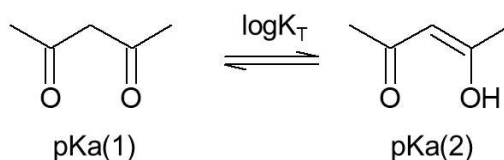


Figure II.I-1. Un exemple de tautomères binaires provenant de cette étude

Dans le cas de la tautomérie prototrope (Figure II.I-1), le logarithme de la constante tautomérique ($\log K_T$), est égal à la différence entre les constantes d'acidité (pKa) des tautomères correspondants partageant un anion commun:

$$\log K_T = pKa(2) - pKa(1) \quad \text{II.I-a}$$

Construction d'un modèle conjugué

1) Intégrer l'équation de la caractéristique principale ($\log K_T$) en construisant la fonction de perte basée sur l'équation E_T .

Modèle individuel $\log K_T$:

$$E_T(w) = \|y_T^{exp} - y_T^{pred}\|^2 = \|y_T^{exp} - Xw\|^2 \quad \text{II.I-b}$$

Modèle basé sur l'équation $\log K_T$:

$$\log K_T = pKa(2) - pKa(1) \Rightarrow y_T^{pred} = X_2w - X_1w = (X_2 - X_1)w \quad \text{II.I-c}$$

$$E_T(w) = \|y_T^{exp} - y_T^{pred}\|^2 = \|y_T^{exp} - (X_2 - X_1)w\|^2 \quad \text{II.I-d}$$

2) Combiner la fonction de perte basée sur l'équation E_T avec les fonctions de perte individuelles de la caractéristique associée (pKa of tautomers) et les termes de régularisation de la complexité du modèle :

Modèle individuel pKa :

$$E_A(w) = \|y_A^{exp} - y_A^{pred}\|^2 = \|y_A^{exp} - Xw\|^2 \quad \text{II.I-e}$$

Modèle conjugué:

$$E(w) = \alpha E_T(w) + (1 - \alpha) E_A(w) + \lambda \|w\|^2 \quad \text{II.I-f}$$

où α passe de 0 à 1 et contrôle le compromis entre la minimisation des erreurs de prédiction des constantes tautomériques et des constantes d'acidité.

3) Calculer les poids de régression (paramètres) du modèle conjugué

$$w = [\alpha(X_2 - X_1)^T(X_2 - X_1) + (1 - \alpha)X^T X + \lambda I]^{-1} [\alpha(X_2 - X_1)^T y_T^{exp} + (1 - \alpha)X^T y_A^{exp}] \quad \text{II.I-g}$$

Les poids de régression optimaux w (paramètres) peuvent également être trouvés en utilisant la méthode de descente de gradient.

Trois approches pour prédire le $\log K_T$ et le pKa des tautomères ont été envisagées : (i) construire un modèle avec un ensemble de données sur 2 371 pKa de molécules organiques pour prédire le pKa des tautomères et calculer le $\log K_T$ selon l'équation 1 ; (ii) construire un modèle avec un ensemble de données sur 639 réactions tautomériques pour prédire directement le $\log K_T$ et (iii) construire un modèle conjugué avec les deux ensembles de données pour prédire simultanément le $\log K_T$ et le pKa des tautomères pour une réaction donnée.

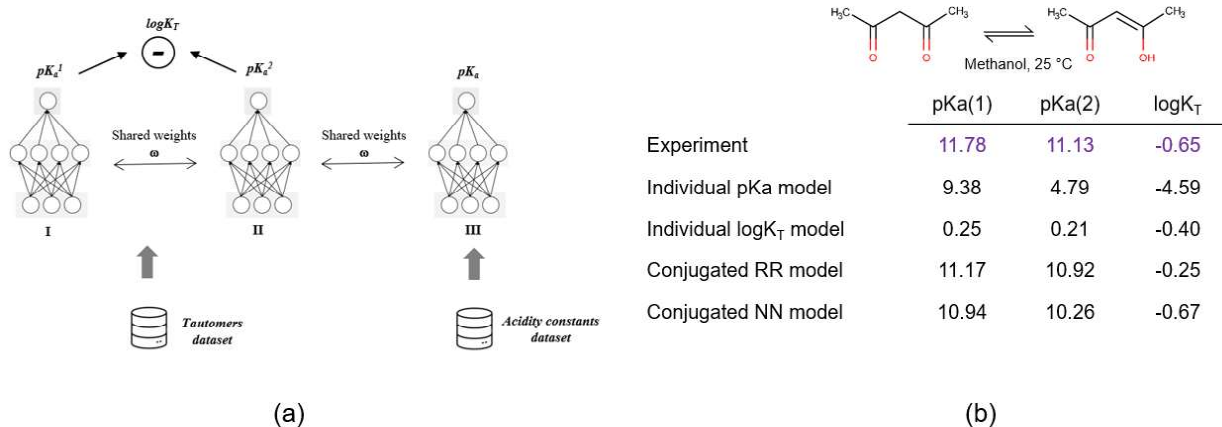


Figure II.I-2. (a) architecture d'un réseau neuronal conjugué pour la prédiction simultanée du $\log K_T$ de réaction et du pKa des tautomères; (b) valeurs expérimentales et prédites par différentes approches du $\log K_T$ pour la réaction de tautomérisation céto-énol et pKa des tautomères correspondants.

Pour illustrer les résultats, une réaction tautomérique avec un $\log K_T = -0.65$ expérimental, un $pK_a(1)$ de 11.78 et un $pK_a(2) = 11.13$ mesurés dans les mêmes conditions (méthanol, 25 °C) a été choisie. L'acidité expérimentale de l'énol $pK_a(2)$ a été déduite de la formule $pK_a(2) = \log K_T + pK_a(1)$. Le modèle pK_a individuel (Figure II.I-2b) prédit avec précision le pK_a de la cétone mais ne parvient pas à prédire le pK_a de l'énol (car aucune donnée expérimentale sur les énols n'est disponible dans les données d'entraînement), ce qui conduit à une valeur calculée inexacte du $\log K_T$ en utilisant l'Équation I.II-a. D'autre part, le modèle de $\log K_T$ prédit bien le $\log K_T$ mais le pK_a de la cétone et de l'énol sont arbitraires (Figure II.I-2b). Le modèle conjugué RR (Ridge Regression) prédit avec précision à la fois le pK_a de la cétone et celui de l'énol, et un peu moins bien le $\log K_T$ de la réaction (Figure II.I-2b). Le modèle RR est intrinsèquement linéaire ce qui ne lui permet pas de s'ajuster finement sur chaque propriété impliquée dans l'Équation I.II-a. La non-linéarité est apportée dans le modèle NN conjugué (Figure II.I-2a), ce qui conduit à des prédictions plus précises du $\log K_T$ (Figure II.I-2b).

II.II Modèles conjugués pour la cinétique de réaction

Une réaction chimique peut être décrite quantitativement par des caractéristiques cinétiques telles que la constante de vitesse ($\log k$), le facteur pré-exponentiel ($\log A$), et l'énergie d'activation (E_a) qui sont liées par l'équation d'Arrhenius:

$$\log k = \log A - \frac{E_a}{2.303RT} \quad \text{II.II-a}$$

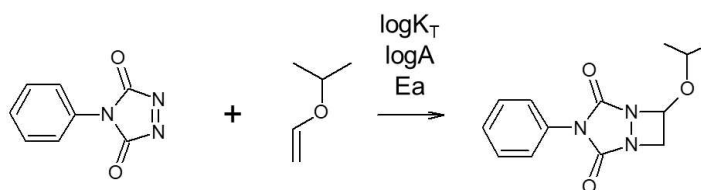


Figure II.II-1. Un exemple de réaction de cycloaddition provenant de cette étude.

Le modèle conjugué peut être construit en intégrant l'équation d'Arrhenius avec l'algorithme de régression de crête :

$$\log k = \log A - \frac{E_a}{2.303RT} \Rightarrow y_K^{pred} = Xw_A - TXw_E \quad \text{II.II-b}$$

Dans cette étude, un ensemble de données sur 1949 réactions de cycloaddition a été extrait d'une publication précédente [10] et utilisé pour générer des modèles *individuels*, *multitâches* et *conjugués* pour la prédiction du $\log k$, du $\log A$ de l' E_a pour les réactions de cycloaddition. Des

modèles individuels (à tâche unique) ont été construits séparément pour chaque caractéristique, tandis que le modèle multitâche a été entraîné en tenant compte des trois caractéristiques cinétiques simultanément. L'apprentissage multitâche peut améliorer la précision de la prédiction des caractéristiques modélisées lorsque les tâches sont corrélées ou partagent certaines informations. Enfin, les caractéristiques quantitatives des réactions sont souvent liées par des relations thermodynamiques qui peuvent être incorporées dans des modèles conjugués. L'apprentissage conjugué utilise toutes les données disponibles sur plusieurs tâches et les intègre explicitement dans une relation mathématique (ici, l'équation d'Arrhenius).

Les modèles conjugués ont été comparés à des modèles individuels entraînés indépendamment pour prédire les paramètres d'Arrhenius et des modèles multitâches, où les paramètres d'Arrhenius ont été modélisés de manière coopérative. Les performances des modèles conjugués, individuels et multitâches sont équivalentes (Tableau 4). Mais les modèles conjugués décrivent de manière beaucoup plus précise la dépendance à la température de la constante de vitesse des réactions par rapport aux modèles individuels et multitâches. En somme, les modèles conjugués ont une meilleure stabilité pour extrapoler les constantes cinétiques des réactions à des températures hors du domaine de valeurs exploré dans les données d'entraînement (températures extrêmement basses ou élevées) (Figure II.II-3).

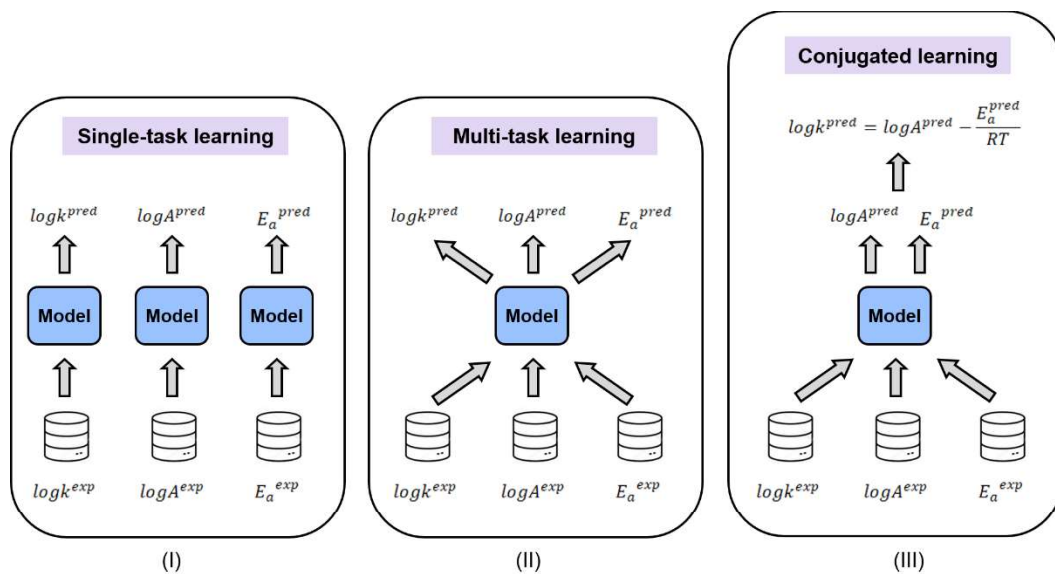


Figure II.II-2. Approches de la modélisation des paramètres de l'équation d'Arrhenius. Dans l'apprentissage mono-tâche classique (I), chaque paramètre est modélisé indépendamment. L'apprentissage multi-tâches (II) ne considère que la relation statistique entre les caractéristiques, tandis que l'apprentissage conjugué (III) intègre la relation mathématique stricte (équation d'Arrhenius) entre elles avec un algorithme d'apprentissage automatique.

Tableau 4. Coefficient de détermination (R^2_{Test}) des prédictions par des modèles individuels, multitâches et conjugués des paramètres de l'équation d'Arrhenius à partir de l'ensemble de test.

	logk	logA	Ea
Modèle individuel	0.78	0.46	0.91
Modèle multitâches	0.76	0.48	0.83
Modèle conjugué	0.75	0.57	0.90

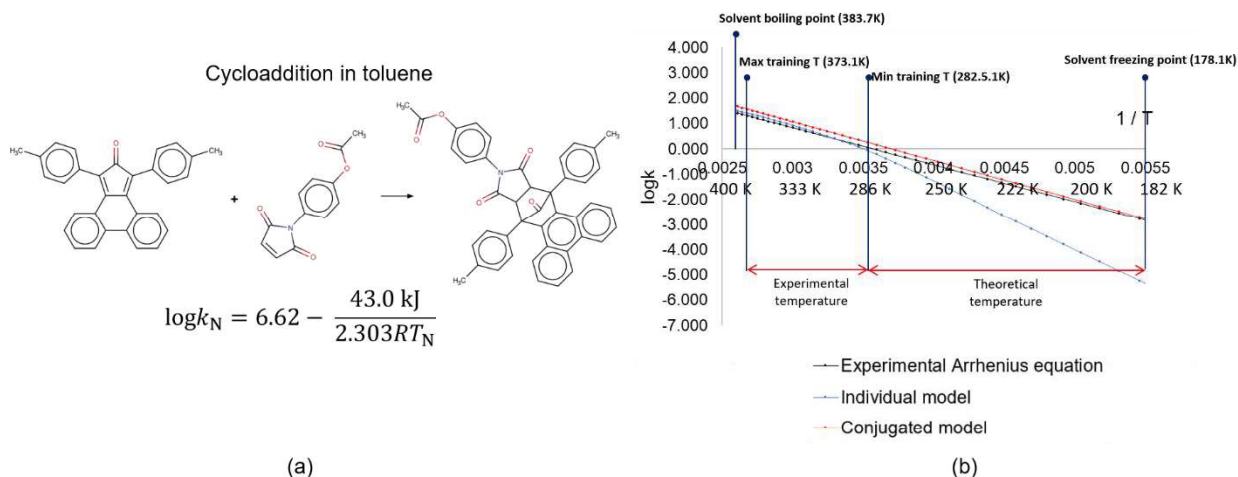


Figure II.II-3. Logk prédit et calculé avec l'équation expérimentale d'Arrhenius avec des modèles individuels et conjugués pour la réaction de cycloaddition à différentes températures dans le toluène.

II.III Modèles conjugués pour la constante de sélectivité les réactions $E2/S_N2$ concurrentes

L'élimination bimoléculaire ($E2$) et la substitution nucléophile bimoléculaire (S_N2) sont des réactions concurrentes conduisant à des produits différents (Figure II.III-1).

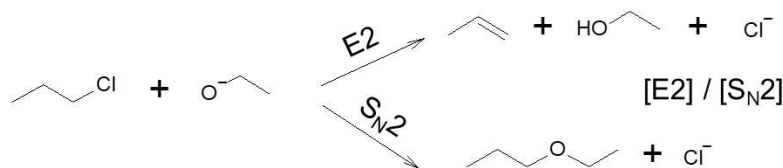


Figure II.III-1. Un exemple de réactions concurrentes $E2$ and S_N2 provenant de cette étude.

La constante de sélectivité $\log(E2/S_N2)$ des réactions concurrentes $E2/S_N2$ peut être calculée comme la différence entre les constantes de vitesse des réactions correspondantes.

$$\log(E2/S_N2) = \log k_{E2} - \log k_{S_N2} \quad \text{II.III-a}$$

et peut être intégrée à l'algorithme de régression de crête :

$$\log(E2/S_N2) = \log k_{E2} - \log k_{S_N2} \Rightarrow y_K^{\text{pred}} = Xw_E - Xw_S \quad \text{II.III-b}$$

Trois types de modèles de prédiction de la constante de sélectivité ont été comparés : le modèle *individuel*, le modèle à *base d'équations* et le modèle *conjugué*. Les performances des modèles sont rapportées dans le Tableau 5. Pour construire les modèles, un ensemble de données de 1764 réactions $E2$, un ensemble de données de 5319 réactions S_N2 et un ensemble de données sur les constantes de sélectivité pour 389 réactions $E2/S_N2$ ont été utilisés. L'ensemble de test comprenait 100 réactions $E2/S_N2$ avec des valeurs expérimentales de $\log k_{E2}$, $\log k_{S_N2}$ and $\log(E2/S_N2)$. Le modèle conjugué prédit la constante de sélectivité moins précisément que le modèle individuel standard de $\log(E2/S_N2)$, mais prédit nettement mieux les vitesses de réaction correspondantes des réactions $E2$ et S_N2 .

Tableau 5. Coefficient de détermination (R^2_{Test}) des prédictions par des modèle individuel, le modèle à base d'équations et le modèle conjugué des constantes de vitesse des réactions concurrentes $E2$ et S_N2 , et la constante de sélectivité $\log(E2/S_N2)$ à partir de l'ensemble de test.

Approche	Données de formation	$E2$	S_N2	$\log(E2/S_N2)$
Modèle individuel	$\log k_{E2}$	0.37	-	-
Modèle individuel	$\log k_{S_N2}$	-	-0.11	-
Modèle individuel	$\log(E2/S_N2)$	-	-	0.89
Modèle à base d'équations	$\log k_{E2}$, $\log k_{S_N2}$	0.37	-0.11	-0.93
Modèle conjugué	$\log k_{E2}$, $\log k_{S_N2}$, $\log(E2/S_N2)$	0.60	0.31	0.72

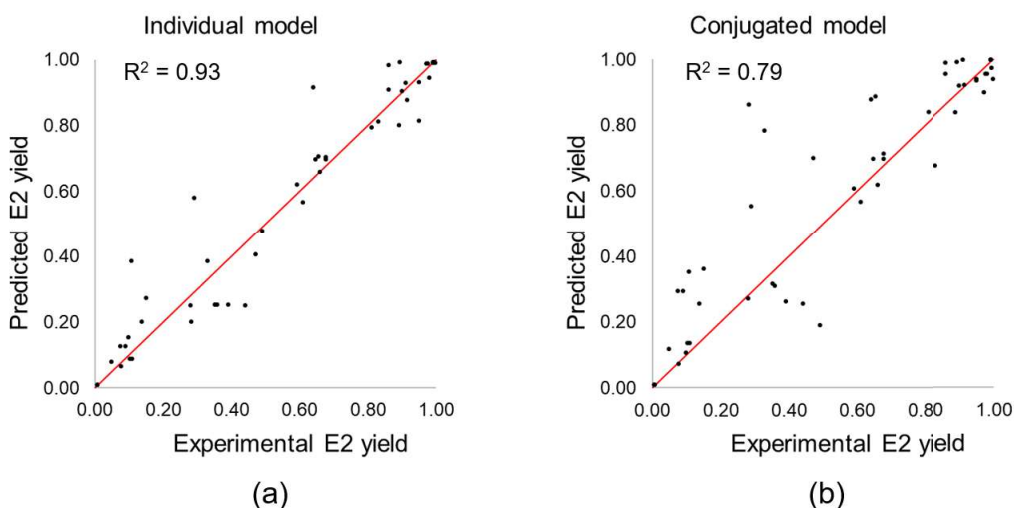


Figure II.III-2. Valeurs expérimentales et prédites de la constante de vitesse $\log k_{S_N2}$ pour 49 réactions de test.

Conclusions

Une nouvelle méthodologie basée sur une combinaison d'approches d'apprentissage multi-instance et de descripteurs 3D (*pmapper*) a été développée et appliquée avec succès à deux tâches différentes : la prédiction de la bioactivité de molécules organiques et l'énantio-sélectivité de catalyseurs chiraux. Dans les deux tâches, les modèles développés ont surpassé les modèles mono-tâche basés sur des descripteurs 2D. De plus, les modèles MIL sont capables de prédire à la fois l'activité moléculaire et d'identifier les conformères actifs. Le protocole de modélisation 3D entièrement automatisé a été écrit en Python 3. Le code source est disponible sur <https://github.com/dzankov/3D-MIL-QSAR> et <https://github.com/dzankov/3D-MIL-QSSR>.

Le concept d'apprentissage automatique conjugué permet d'intégrer des lois thermodynamiques avec l'apprentissage automatique classique. Il a été appliqué à deux tâches, chacune liée à la modélisation prédictive de plusieurs propriétés physiques liées par des équations thermodynamiques: (i) constantes d'équilibre tautomérique/pKa et (2) paramètres de l'équation d'Arrhenius. Bien que la précision des prédictions des modèles conjugués soit comparable à celle de modèles individuels, les modèles conjugués garantissent le respect des relations mathématiques entre les propriétés modélisées. Le code du programme pour la construction de modèles conjugués est disponible sur <https://github.com/dzankov/CoLearn>.

Références

- [1] T.G. Dietterich, R.H. Lathrop, T. Lozano-Pérez, Solving the multiple instance problem with axis-parallel rectangles, *Artif. Intell.* 89 (1997) 31–71. [https://doi.org/https://doi.org/10.1016/S0004-3702\(96\)00034-3](https://doi.org/https://doi.org/10.1016/S0004-3702(96)00034-3).
- [2] D. V. Zankov, M. Matveieva, A. V. Nikonenko, R.I. Nugmanov, I.I. Baskin, A. Varnek, P. Polishchuk, T.I. Madzhidov, QSAR Modeling Based on Conformation Ensembles Using a Multi-Instance Learning Approach, *J. Chem. Inf. Model.* 61 (2021) 4913–4923. <https://doi.org/10.1021/acs.jcim.1c00692>.
- [3] Kutlushina, A. Khakimova, T. Madzhidov, P. Polishchuk, Ligand-Based Pharmacophore Modeling Using Novel 3D Pharmacophore Signatures, *Molecules.* 23 (2018) 3094. <https://doi.org/10.3390/molecules23123094>.
- [4] D. Zankov, P. Polishchuk, T. Madzhidov, A. Varnek, Multi-Instance Learning Approach to Predictive Modeling of Catalysts Enantioselectivity, *Synlett.* 32 (2021) 1833–1836. <https://doi.org/10.1055/a-1553-0427>.
- [5] A.F. Zahrt, J.J. Henle, B.T. Rose, Y. Wang, W.T. Darrow, S.E. Denmark, Prediction of higher-selectivity catalysts by computer-driven workflow and machine learning, *Science* (80-.). 363 (2019).
- [6] F. Sandfort, F. Strieth-Kalthoff, M. Kühnemund, C. Beecks, F. Glorius, A structure-based platform for predicting chemical reactivity, *Chem.* 6 (2020) 1379–1390.
- [7] R. Asahara, T. Miyao, Extended Connectivity Fingerprints as a Chemical Reaction Representation for Enantioselective Organophosphorus-Catalyzed Asymmetric Reaction Prediction, *ACS Omega.* (2022).
- [8] J.L. Melville, B.I. Andrews, B. Lygo, J.D. Hirst, Computational screening of combinatorial catalyst libraries, *Chem. Commun.* (2004) 1410–1411.
- [9] D. V Zankov, T.I. Madzhidov, A. Rakhimbekova, T.R. Gimadiev, R.I. Nugmanov, M.A. Kazymova, I.I. Baskin, A. Varnek, Conjugated Quantitative Structure-Property Relationship Models: Application to Simultaneous Prediction of Tautomeric Equilibrium Constants and Acidity of Molecules, *J. Chem. Inf. Model.* 59 (2019) 4569–4576.
- [10] M. Glavatskikh, T. Madzhidov, D. Horvath, R. Nugmanov, T. Gimadiev, D. Malakhova, G. Marcou, A. Varnek, Predictive Models for Kinetic Parameters of Cycloaddition Reactions, 38 (2019) e1800077. <https://doi.org/10.1002/minf.201800077>.

Structure-property modeling with advanced machine learning techniques

Introduction

Machine Learning (ML) is a branch of Artificial Intelligence (AI) that focuses on learning and predicting from data. Machine learning is applied in finance, marketing, self-driving cars, social media, language translation, healthcare, education, drug discovery, etc. Machine learning concepts and methods often emerged as a way to solve specific problems from the real world. For example, in 1989 LeCun [1] presented the first application of Convolutional Neural Networks (CNN) trained with a backpropagation algorithm for the recognition of handwritten digits. CNN was inspired by the visual nervous systems of living organisms and is based on such operations as feature extraction, pooling, and convolution. As a result, modern CNN architectures outperform humans in the tasks of image recognition. In 1986, Rumelhart presented Recurrent Neural Networks (RNN) [2], which were enhanced by the LSTM mechanism (Schmidhuber [3], 1997) and then by the attention mechanism (Bahdanau [4], 2015). RNNs are successful in sequence modeling tasks such as text classification, language translation, voice recognition, and DNA analysis. In 1997, Dietterich introduced the concept of Multi-Instance machine Learning (MIL) [5], which deals with problems where an object cannot be represented by a single instance and a single feature vector. This pivotal work was motivated by the drug prediction problem, in which a compound can be represented by multiple alternative conformations, and it is not known which conformation is responsible for the observed bioactivity of a given compound. Dietterich proposed an Axis-Parallel Rectangles (APR) approach to solving the MIL problem and demonstrated that addressing the MIL problem can significantly increase the performance of predictive models. Since then, numerous MIL algorithms have been developed and applied in various real-world tasks, such as computer vision, time series analysis, text processing, bioinformatics, etc.

However, while MIL was first introduced for the drug activity prediction problem, it has not become a popular approach in chemoinformatics and only a few papers on the application of MIL to structure-activity modeling were known before this Ph.D. project. In this Ph.D. project, a new 3D structure-property modeling approach was developed based on ensembles of molecular conformations and multi-instance learning algorithms. This 3D approach does not require the selection and alignment of conformers and can be applied to both classification and regression tasks. Additionally, models obtained with the of this 3D approach not only predict molecular activity but are also can identify some key conformations (for example, bioactive conformations) responsible

for observed experimental values of the target property. The modeling protocol is written in Python 3 and is based only on free software packages and is fully automated, allowing the developed 3D approach to be integrated into desktop or WEB applications for the automatic construction of predictive models. The developed approach was tested in the modeling of (i) the bioactivity of compounds from the ChEMBL-23 database and (ii) the enantioselectivity of organic chiral catalysts in asymmetric synthesis - these properties critically depend on the 3D structure of the molecule.

The second part of the thesis is devoted to the development of conjugated models, which integrate thermodynamic and kinetic laws with machine learning algorithms. Some quantitative characteristics of chemical reactions are related by mathematical equations (e.g., the Arrhenius equation). In conjugated machine learning, such equation-related characteristics are embedded into the machine learning algorithm, i.e., equation-based and individual models are algorithmically combined into one conjugated model. As a result, conjugated models provide accurate predictions of reaction characteristics that strictly satisfy fundamental equations. In such a way, the chemical laws integrated with the machine learning algorithm act as a regularizer for predictive models. In this research project, conjugated machine learning was applied to three types of reactions (and equations): the tautomeric reactions (tautomeric equation), the cycloaddition reactions (Arrhenius equation), and the competing E2/S_N2 reactions (selectivity equation).

This Ph.D. project contributes to the development of machine learning approaches that consider the complexity of chemical objects (molecules) and processes (chemical reactions). Multi-instance machine learning in combination with 3D descriptors allows the construction of 3D models, which does not require the selection and alignment of conformations. Conjugated QSPR models for predicting reaction characteristics are based on thermodynamic and kinetic laws, which bridge chemistry with machine learning.

Part 1. Multi-instance machine learning in chemoinformatics and bioinformatics

Multi-Instance Learning (MIL) problem was formalized in 1997 and has since been successfully applied in drug discovery (pharmacy), classification of text documents (information retrieval), classification of images (computer vision), speaker identification (signal processing), bankruptcy prediction (economy), etc. Although one of the first applications of MIL was drug activity prediction, MIL has not become a popular approach in structure-activity modeling. On the other hand, there are many examples of MIL applications to bioinformatics tasks for modeling interactions between biological macromolecules such as proteins, DNA and RNA. However, there is still no systematic review of MIL applications in chemoinformatics and bioinformatics. For this reason, this review on the application of MIL to modeling the properties and functions of small molecules (chemoinformatics) and biological macromolecules (bioinformatics) has been prepared. It also includes a description of the MIL framework, the type of tasks in MIL, and the MIL algorithms.

1.1 Introduction

The properties of chemical compounds are a function of their structure. Structure-property modeling approaches apply special algorithms to extract the correct relationship between the structure of the molecule and its properties. In the traditional structure-property modeling approaches each molecule is encoded with a set of numerical chemical descriptors followed by the application of special algorithms like machine learning algorithms to establish the correlation between descriptors and the property values. One of the key limitations of traditional structure-property modeling is the requirement that each molecule has to be represented by a single instance with a fixed conformation, protonation state, tautomer, stereoconfiguration, etc. As a result, a molecule has to be associated with a single vector of descriptors. However, a molecule is a dynamic object and simultaneously exists in many forms/instances in equilibrium. This raises the problem of the selection of the molecular form for structure-property modeling, as the actual molecular form responsible for the observed property is often unknown.

The same problem exists in the «structure-function» modeling of biological functions of macromolecules (proteins, DNA and RNA). Biological macromolecules are sequences of “monomers” (amino acids or nucleotides) and can interact each with other to perform various biological functions. However, only particular subsequences/segments of a macromolecule of limited length are responsible for the interaction between macromolecules, and experimental information on

modeling have been published then [7–13]. As part of this Ph.D. project, a large-scale comparison of MIL models based on an ensemble of conformations and traditional 2D models based on popular 2D descriptors was published for the task of modeling the bioactivity of compounds from 175 datasets extracted from the ChEMBL-23 database [12–14]. In another part of this Ph.D. project, the first application of MIL for modeling the enantioselectivity of chiral organic catalysts in asymmetric organic synthesis [15] is recently published. Another illustrative example is paper [16], where molecules were represented by a bag of atoms (instances) for the modeling of the acidity of compounds. In bioinformatics, MIL has attracted significantly more attention, because of a large number of tasks [17–30] perfectly fitting the MIL framework.

Despite the attractiveness of the MIL approach, there is still no comprehensive review of the application of MIL in modeling the properties and functions of molecules. This part of the Ph.D. project provides a detailed description of MIL approaches and their applications. This review includes a description of the MIL framework and the main MIL algorithms, as well as examples of MIL applications in chemoinformatics and bioinformatics.

1.2 Origins of multi-instance learning

The first examples of multi-instance problems were known before Dietterich’s seminal paper in 1997 [5]. The first examples of such projects concern chemical structure determination by mass spectroscopy [31], phoneme recognition [32], recognition of handwritten characters [33], dynamic reposing in drug activity prediction [34], and modeling DNA promoter sequences [35].

The application of MIL to solve a particular machine learning problem is conditioned by the structure of the data. Multi-instance learning is a suitable learning framework in tasks where the modeled object is difficult to represent with a single feature vector. The sort of problems, where an object can exist in several alternative representations, can be attributed to *polymorphism ambiguity* (Figure 2). In structure-property modeling, this type of ambiguity arises when the molecule can be represented by alternative instances, such as conformations, tautomers, protonation states, etc. The wrong choice of the key molecular form can result in the poor performance of predictive models. MIL is a suitable framework for this problem because it can handle all available instances simultaneously.

Another problem where MIL is applicable is characterized by a *part-to-whole ambiguity* when only one or several parts of a modeled object are responsible for its observed property. A molecule can be represented as a set of connected atoms/instances and its physicochemical or

biological properties are generally influenced by a single atom or group of atoms, and it is often unknown which particular atom determines the observed molecule’s property [16].

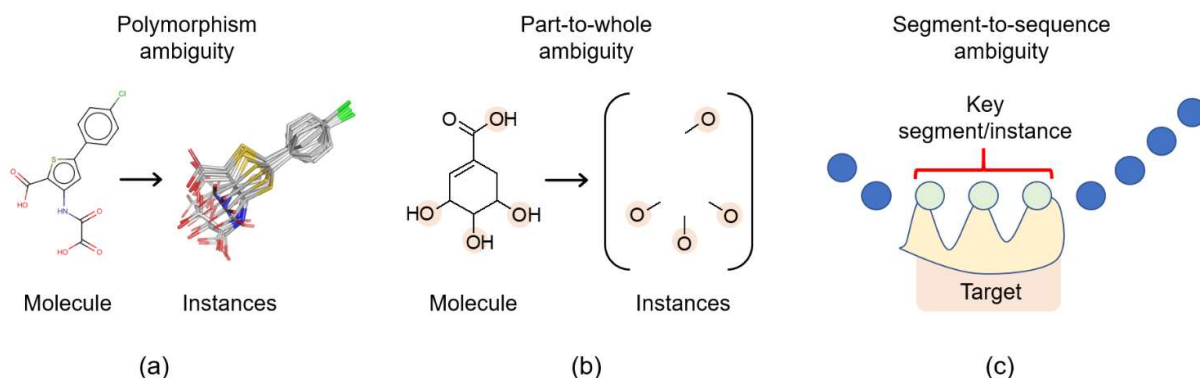


Figure 2. Types of ambiguity in molecule structure data: (a) polymorphism ambiguity, (b) part-to-whole ambiguity, and (c) segment-to-sequence ambiguity.

MIL is also a quite popular modeling approach in bioinformatics, where modeled objects are sequences, such as protein, DNA, and RNA. Often only a certain segment of the sequence is responsible for the function of the whole sequence, but the length and boundaries of such a segment may be unknown. Consequently, biological sequences can be represented by multiple segments, which can overlap with each other. Each segment of the sequence is an instance encoded with a special feature vector. This type of problem can be attributed to *segment-to-sequence ambiguity*.

Other multi-instance problems include multi-multi-instance learning [36], multi-instance multi-label learning [37], key instance detection in multi-instance learning [6], multi-instance clustering [38], multi-instance ranking [39]. Comprehensive reviews of the MIL concept and its applications can also be found in [40–47].

1.3 Multi-instance learning algorithms

The growing number of MIL algorithms requires their systematization. This review follows a categorization of algorithms similar to [44] (other types of categorization of MI algorithms are described in [42,45,48,49]) and distinguishes two major groups of MIL algorithms: instance-based and bag-based algorithms. Instance-based algorithms consider each instance as a separate training object and generate predictions for each instance in the bag, and then apply a predefined rule to aggregate the instance predictions to obtain a prediction for the entire bag.

In contrast to instance-based algorithms, bag-based algorithms consider the whole bag as a training object and do not explicitly provide predictions for individual instances. The bag-level

algorithms consider the bag as a whole object and define the distance between bags [50], bag kernels [51], bag dissimilarities [52] or explicitly pooling operators.

Naive MIL algorithms

Naive MIL algorithms such as *wrappers* transform multi-instance data into single-instance representation and apply a traditional machine learning algorithm to train the model. Following the chosen categorization of MIL algorithms, there are two types of *wrapper* algorithms: instance-based and bag-based wrapper algorithms (Figure 3).

In *Instance-Wrapper* (Figure 3a) each training instance of a bag is assigned the same label as the parent bag. This results in a standard single-instance dataset in which each instance is manually labeled and any single-instance machine learning algorithm can be applied to build the model. To obtain a prediction for a new bag, the model first predicts a label for each instance of the bag and then aggregates obtained instance predictions (e.g., averages) to produce a prediction for the given bag.

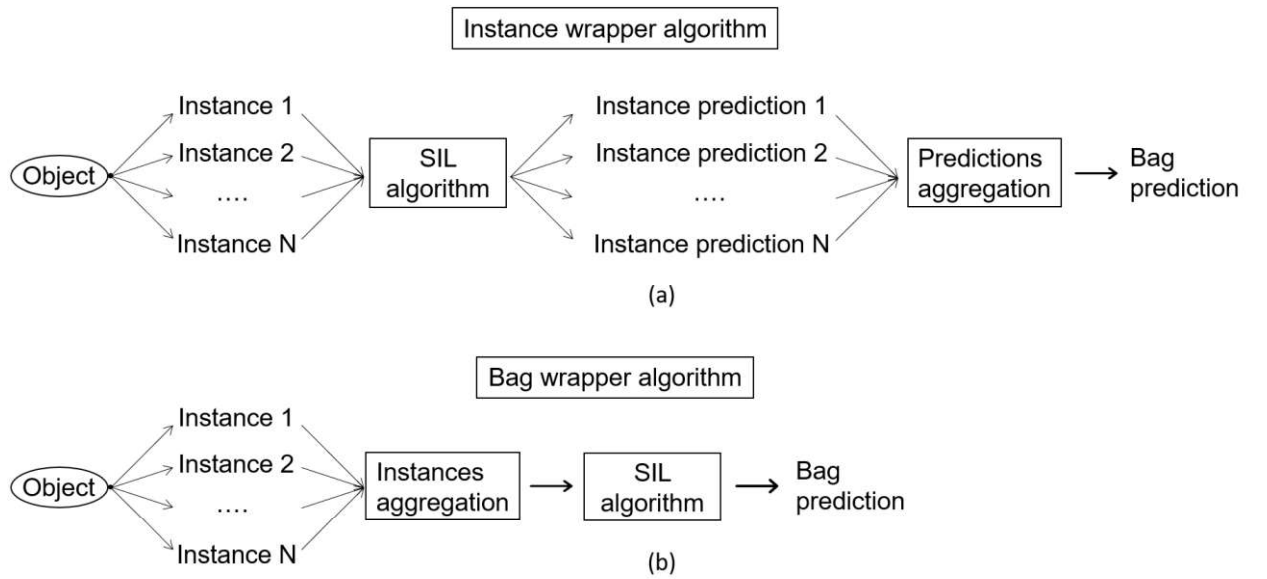


Figure 3. Prediction scheme in instance- and bag-wrapper MIL algorithms.

In *Bag-Wrapper* (Figure 3b) algorithm, there is no need to identify a label for each instance in a bag. Instead, there is an operation that aggregates the instances to obtain a single vector representing the bag. Then, any single-instance algorithm can be applied to train the model. In prediction mode, all instances of the new bag are aggregated into a single vector, which is used to obtain a prediction for a given bag.

Traditional MIL algorithms

Naive MIL algorithms such as *wrappers* transform multi-instance data into single-instance data and use a standard single-instance machine learning algorithm to train the model. However, several classic machine learning approaches have been adapted to directly process raw multi-instance data. These algorithms are instance-based approaches such as maximum likelihood-based methods [53–56], decision rules and tree-based methods [57–60], SVM-based methods [43], and evolutionary-based methods [61]. Bag-based algorithms include the adapted nearest neighbor methods [50,62] and bag-level SVM methods [43].

For example, MI Logistic Regression [63] is an adaptation of logistic regression, DPBoost [64] and MI Boosting [63] are adaptations of boosting approach, ID3-MI and RipperMI [57] are the MIL extensions of the decision tree, and decision rules approaches, MI-SVM is the multi-instance version of SVM [65], Citation-kNN [50] is a multi-instance version of standard kNN, bag-level SVM methods are based on the bag-level kernels [51]. There are also multi-instance adaptations of neural networks (section 1.3.3).

The Diverse Density [53] is a maximum likelihood-based algorithm that implements the assumption that positive instances occupy a specific area in the feature space. Diverse Density searches for the area in the feature space where the difference between the density of instances of positive and negative is maximal. For example, if one of the instances in a positive bag is close to the prototype and no negative bags are close to the prototype, then the prototype will have a high Diverse Density. The DD algorithm searches for the prototype instance that is a generalization of a positive instance. Expectation-Maximization Diverse Density (EM-DD) uses the EM algorithm to locate prototype instances more efficiently. There are several other MI algorithms based on the Diverse Density approach, such as DD-SVM [66] and MILES [67].

Neural network MIL algorithms

Neural networks are appealing for solving MIL problems. Neural networks perform multi-instance learning in an end-to-end, which takes a bag with a various number of instances as input and generates the bag label. Multi-instance neural networks were first described by Ramon et al. [68] for classification problems where instance probabilities are computed to be further aggregated by the log-sum-exp operator to calculate the bag probability. Zhou et al. [69] modified multi-instance neural networks by employing a new loss function capturing the nature of multi-instance learning, i.e. weights of the network are updated for each training bag, not for each training instance. Later, this neural network was improved by adopting feature scaling with Diverse Density and feature reduction by principal component analysis [70]. In [71] and [72] ensemble neural networks and

RBF neural networks were introduced to solve MIL problems. Zhang et al. [73] extended multi-instance neural networks by implementing a loss function for the MIL regression task.

Wang et al. [74] revisited multi-instance neural networks and proposed a series of novel neural network frameworks for MIL. In contrast to previous multi-instance networks, their method focuses on generating bag representations instead of inferring instance labels. The proposed network consists of three fully-connected layers followed by one pooling layer that aggregates instance representations learned by previous layers into a single embedding vector. A final fully-connected layer takes the obtained embedding vector as input and calculates the bag probability. The authors examined three typical pooling operators for aggregation instance feature vectors - *max*, *mean* and *log-sum-exp* pooling and concluded that all pooling operators demonstrate similar classification accuracy on benchmark datasets. Besides that, they integrated popular deep learning tricks (deep supervision and residual connections) into MIL networks, which improved the classification accuracy. The important outcome of this paper is that bag-level networks (Figure 4b) outperform instance-level networks (Figure 4a) on popular MIL benchmark datasets.

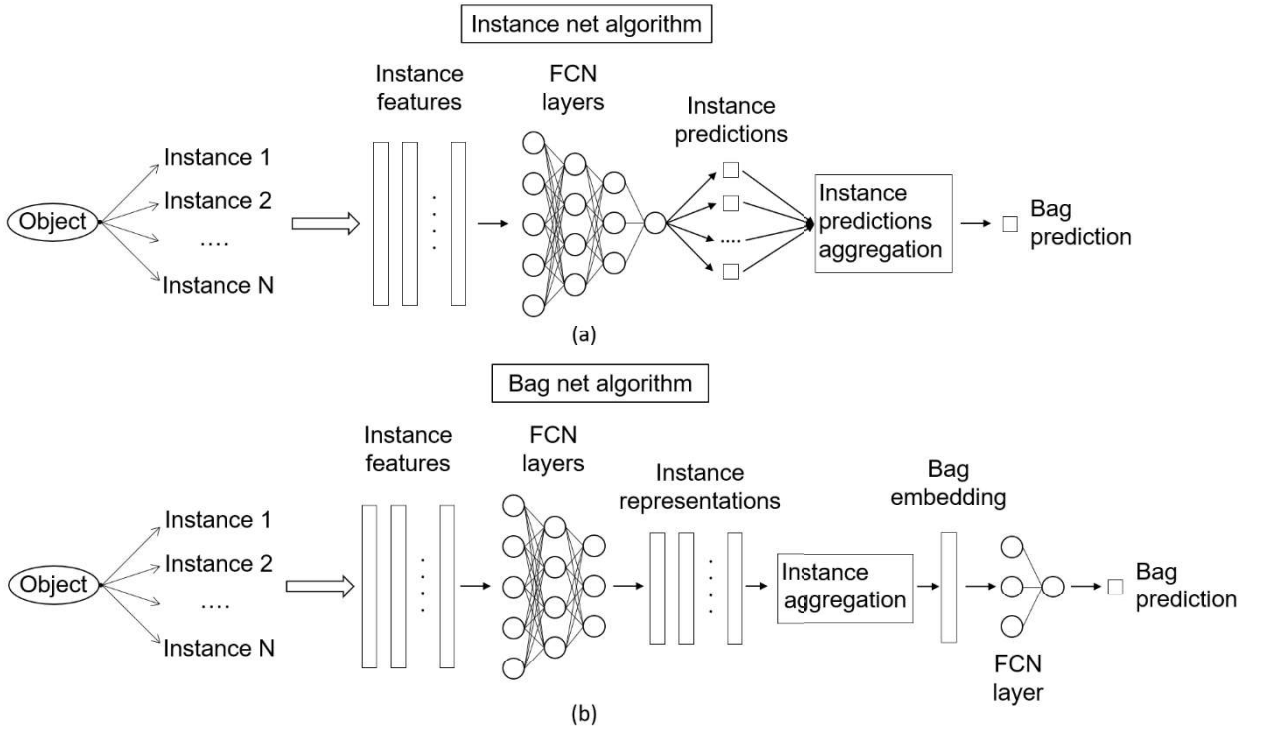


Figure 4. Examples of instance- and bag-based multi-instance neural networks.

Traditional pooling operators have a clear limitation, i.e. they are pre-defined and non-learnable. The *max*-pooling operator could be effective to aggregate instance scores but might be inappropriate for the aggregation of instance feature vectors in bag-level algorithms. Similarly, the *mean* pooling operator might be unsuitable to aggregate instance scores but could succeed in generating the aggregated bag representation. Ilse et al. proposed an attention-based pooling operator,

that replaces pre-defined pooling operators with a trainable attention network that can generate instance weights [75]. Instance weights quantify the importance of each instance and its contribution to the aggregated bag representation.

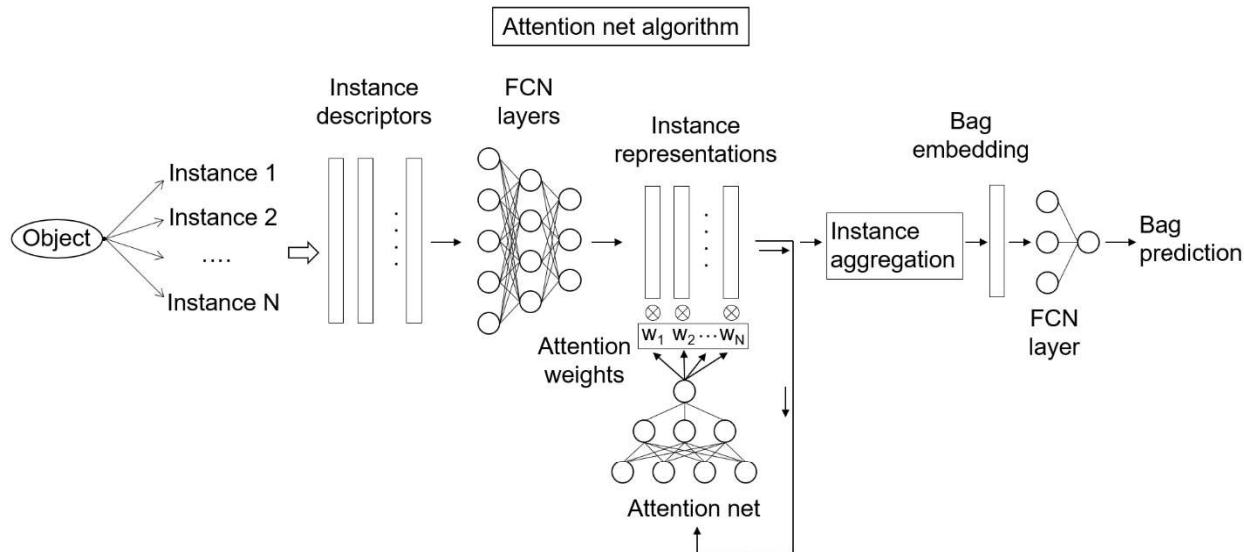


Figure 5. The architecture of the attention-based multi-instance algorithm

However, most MIL algorithms ignore the structural relationship among instances in the bag because they consider the instances as independently and identically distributed (i.i.d) samples [76]. In this context, instances are i.i.d if they have the same probability distribution and all are mutually independent. For example, considering molecules as i.i.d data samples is reasonable, but the conformation distribution of a molecule is not independent and identical because it depends on predefined physical laws. Nevertheless, multi-instance neural networks that can capture structural information within a bag have been proposed.

Tu et al. [77] proposed a multi-instance learning approach with graph neural networks. In this approach, each bag of instances is converted to an undirected graph which is processed by Graph Neural Network (GNN) to learn the aggregated bag representation. The authors claimed that the graph representation of a bag allows for capturing the structural information within the bag and demonstrated that it can improve the classification accuracy of the algorithm.

In [78] recurrent neural networks were proposed to model underlying structure among instances. In this approach each bag is converted into an unordered sequence of instances, which is processed by the recurrent neural network, that can memorize instances. In [79], a new pooling operator based on the LSTM recurrent neural network was proposed. In this pooling operator, the LSTM memory mechanism allows accumulating of information after processing each instance representation to iteratively update the bag representation.

In [80] a new dynamic pooling was proposed, which was inspired by the *Routing Algorithm* from *Capsule Networks* [81]. The dynamic pooling iteratively updates instance contribution to aggregated bag representation and captures the contextual information among instances.

Set Transformer [82], which is based on *Transformer* architecture [83], was proposed for solving problems where data samples are organized as sets of instances, including multi-instance learning. *Set Transformer* model pairwise interactions between instances in a bag using the *multi-head self-attention* mechanism. Each head in *multi-head self-attention* highlights local relationships between groups of instances in the bag.

Key instance detection algorithms

The main goal of MIL algorithms is to predict labels for bags. However, it is often desirable to predict not only the bag label but also to infer labels of the instances in the bag. It is particularly important to determine labels for the key instances that primarily contribute to the label of the bag. This problem was called Key Instance Detection (KID) and was first formalized in [6]. The development of MIL algorithms that can predict the label of a bag and identify key instances of this bag is an attractive area of research. KID problem related to the problem of explainability of MIL models. Following the categorization of [84], explainable approaches of MIL models can be divided into model-specific and model-agnostic.

Model-specific approaches include MIL algorithms that can infer instance labels or estimate the importance of instances (instance weights). These algorithms can be roughly divided into traditional and neural network-based algorithms. Most traditional instance-level algorithms can be used to identify key instances. Instance-level algorithms rely on some process, which determines the labels or probabilities of instances in a bag. In such algorithms [53,54,65,67,85,86][87], KID is a subtask and instance labels are provided as by-products of the learning process. Other algorithms are based on some key instance identification mechanism and specifically focused on solving the KID problem [6,88,89].

Multi-instance learning with key instance detection

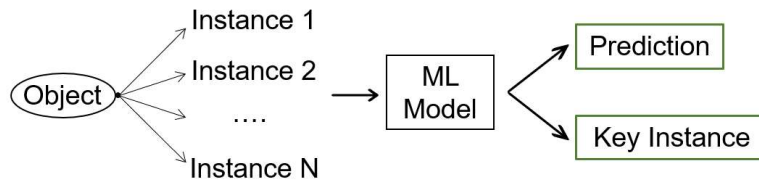


Figure 6. Multi-instance learning with key instance detection

Multi-instance neural networks are an attractive technology for solving the KID problem. An important element of such neural networks is the pooling operator, which aggregates instance representations and can also serve as a detector of key instances. In [75] Ilse et al. proposed a pooling operator based on the attention mechanism [90], which was implemented as a two-layered neural network followed by the *softmax* function that receives instance scores and generates instance weights that sum to 1 (the higher the instance attention weight, the more important the instance). The instances are then aggregated according to the attention weights. Both neural networks are trained consistently using a backpropagation algorithm. Li et al [91] proposed a deep multiple instance selection frameworks (DMIS) based on hard attention [92] with Gumbel softmax or Gumbel top-k functions. In contrast to soft attention, where continuous attention weights are assigned to the instances, including negative instances, the proposed approach selects several key instances, filtering out potential negative (non-key) instances. This approach is more efficient for some tasks than standard attention-based MIL pooling [91]. Yu et al [93] applied a neural network inversion mechanism [94] to the MIL classification problem and demonstrated that it can significantly improve KID performance. In this approach, the attention-based multi-instance neural network is first trained in standard mode and then neural network inversion is applied for each positive bag, which changes the input instances, enhancing the probable key instances and attention weights are recomputed for the updated bag. As a result, after neural network inversion, the key instances are assigned higher attention weights.

There are also multivariate neural networks based on other types of pooling that can also identify key instances. Gaussian pooling [95] applies a Gaussian radial basis function to calculate instance weights, which is the main difference from attention-based pooling, which applies *softmax* for this purpose. Inspired by the *Routing Algorithm* from *Capsule Networks* [81], a new type of pooling operator was proposed in [80], called dynamic pooling. This pooling operator iteratively updates the instance contribution to its bag representation during each feed-forward step. Based on these instance contributions, dynamic pooling highlights the key instance and models the contextual information among instances. Tu et al. [77] implemented an approach, where each instance of a bag is a node in a graph that was processed by a graph neural network (GNN) and converted to a fixed-dimensional representation by differentiable graph clustering pooling. This approach can capture interactions between instances in a bag, which can improve KID performance in some cases [77].

However, the interpretation of attention mechanisms in MIL is still an open question, since validation of KID solutions requires labeled data at the instance level, and the amount of such data is still scarce. A study [96] addresses this issue and concludes that models with high prediction

accuracy can have poor key instance identification accuracy. This fact complicates the selection of models that can be used to solve the KID problem. In the same paper [96] it was demonstrated that using an ensemble of models instead of a single model, can improve the robustness of KID models. These conclusions can be considered general and be extended to the case of other pooling operators. It is necessary to further develop approaches that will increase the validity of KID mechanisms.

The model-agnostic approach for the interpretation of any MIL model in classification tasks was proposed in [84]. This approach can be divided into methods that ignore interactions between instances and methods that recognize these interactions. The first group of methods includes simple strategies such as single instance prediction or one instance removed prediction or their combination. The second method is represented by the Multiple Instance Learning Local Interpretations (MILLI) approach, which is similar to the popular single-instance machine learning LIME and KernelSHAP approaches for model interpretability. Interestingly, model-agnostic approaches performed significantly better in the identification of key instances [84] than model-specific inherent KID mechanisms of popular MIL algorithms.

Boltzmann distribution. The distribution of conformers (fractional occupancy) in time and space is described by the Boltzmann distribution function:

$$\frac{N_i}{\sum N_i} = \frac{e^{\frac{E_i}{kT}}}{\sum e^{\frac{E_i}{kT}}} \quad (1)$$

where E is the energy of the conformer, k is the Boltzmann constant, and T is the temperature of the system. The Boltzmann distribution relates the energy of the conformer to its probability of occurring. The distribution shows that conformers with lower energy always have a higher probability of occurring. The same distribution can be applied to an ensemble of tautomers. Boltzmann's law implies that all molecular forms (conformers/tautomers) contribute to the observed property of the molecule.

Having accurate ligand-target binding energies, the Boltzmann distribution can be used for weighted averaging of the calculated or predicted properties of the molecules. For example, in [97] the Boltzmann distribution (applied to the energies of an ensemble of ligand-target complexes) was used to average the docking scores for the ensemble of each binding pose. As a result ligand ranking accuracy was improved by Boltzmann weighting applied to the energies of an ensemble compared to the straightforward averaging. The more accurate the estimated energies of the system

(conformer, tautomer, ligand-target complexes), the higher the chance of identifying the key molecular form. However, the accuracy of the assessment of these energies is limited by the high computational costs, limited force field accuracy, and technical challenges related to computational resources.

1.4 Multi-instance learning applications

Polymorphism ambiguity modeling

Bioactivity modeling with multiple tautomers. Many compounds exist as tautomers, which can exhibit different physicochemical and biological properties. There are many examples [98] where a minor tautomer binds to the target and is responsible for the observed bioactivity of a compound.

Several papers have studied the influence of tautomerism on QSAR modeling. In [99], it was demonstrated that tautomerism significantly influences the descriptor selection process, as well as in some cases the performance of QSAR models. The same authors later concluded [100] that the inclusion of keto-enol tautomerism in the modeling of antimalarial activity does not affect the performance of the models, but enables retrieving additional useful information on the relation between structure and activity. Another study [101] demonstrated that inclusion in the modeling of both the keto-form and the enol-form of compounds improves the prediction accuracy of the anxiolytic activity, in comparison to models which are built using only one of the two tautomeric forms.

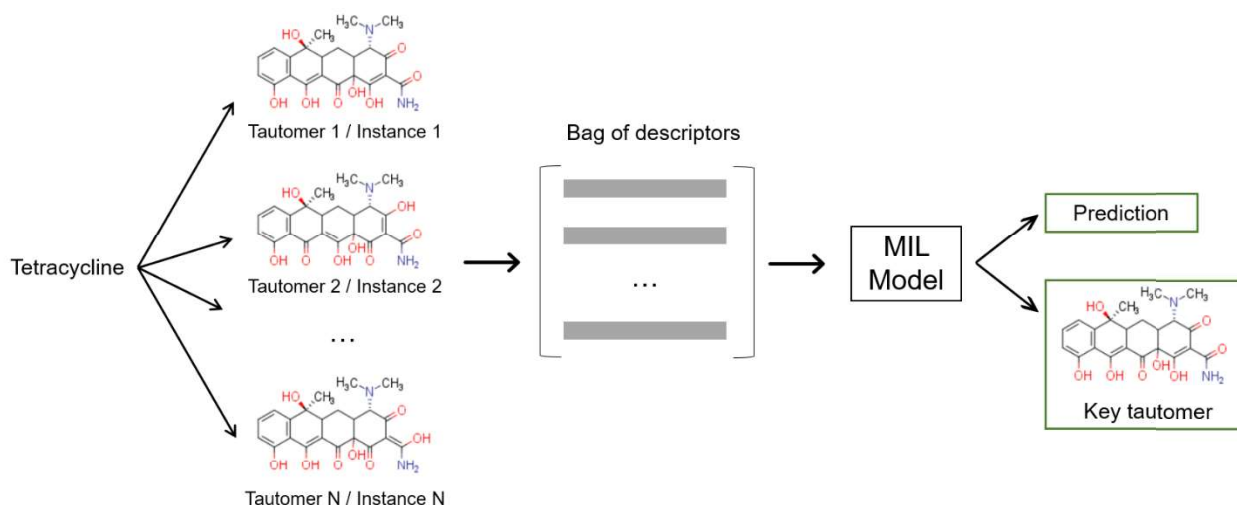


Figure 7. Possible tautomeric forms of tetracycline [102] are inputs to the MIL model. All tautomeric forms of each molecule can be assembled into bags, which are used for structure-activity modeling using multi-instance learning algorithms.

Tautomerism can affect not only the accuracy of in-house QSAR/QSPR models but also the output predictions of external models when they are applied to new compounds. It is well known that logP and pKa can differ for different tautomeric forms of the compound. Recently, the Syn-genta group has demonstrated [103] that logP and pKa predicted by industry-standard programs (clogP program and ACD software) depend on the input tautomer of the compound, and using more sophisticated QM calculations to find the correct tautomer significantly improves the accuracy of logP and pKa predictions.

Multi-instance learning can potentially solve the problem of selection of the relevant tautomer by generating models that are trained on all available tautomers of a molecule. MIL models (Figure 7) can be independent of input tautomer form and even can identify the key tautomer of the compound.

Bioactivity modeling with conformation ensembles. 2D descriptors ignore the spatial molecular structure of compounds and their conformational flexibility. Therefore, some important structural information that could increase the performance of predictive models may be lost. This issue motivated the development of 3D modeling approaches. The Achilles' heel of these approaches is that the molecule is represented by a single generated conformation, which may not be identical to the bioactive conformation. Therefore, it is important to consider the conformational flexibility of the compound, since an incorrect choice of conformation for modeling can significantly reduce the accuracy of the predictive models.

The idea of considering multiple molecule conformations in modeling bioactivity was implemented in *Compass* [34], an algorithm that automatically selects bioactive conformations and their alignments. *Compass* is based on a neural network that iteratively selects a more suitable conformation of a molecule to improve a prediction of its bioactivity. The neural network marks the best pose of each molecule according to the highest predicted activity. The best poses are then used to iteratively update the neural network weights. As a result, the trained model can simultaneously predict both the bioactivity of a compound and its bioactive pose. *Compass* first was applied to predict the human perception of musk odor. The dataset contained 102 molecules, including active (musk) and inactive (non-musk) examples. The model built with a single conformation per molecule demonstrated performance of 71%, while the model generated from multiple conformations demonstrated a significantly higher performance of 91%. This result is an illustrative example of the importance of the representation of the conformational space of molecules.

In the seminal paper [5], Dietterich et al. first introduced the problem of multi-instance learning, motivated by the task of predicting drug activity. In this work, they proposed three basic approaches for the design of axis-parallel hyper-rectangles (APR) classification algorithms, which

are based on the selection of the relevant features and the determination of optimal bounds along these features. The «standard» APR bounds the positive examples and ignores the MIL problem. The «outside-in» and «inside-out» algorithms address the MIL problem and aim to construct optimal hyper-rectangles avoiding negative examples. APR algorithms were compared on one artificial and two real Musk-1 and Musk-2 datasets. Additionally, the traditional single-instance neural network and C4.5 algorithms were chosen for comparison. The results indicated that the algorithms ignoring the instance problem performed inferior to the multi-instance APR algorithms on all three datasets. Although there were previously related works on MIL problems, Dietterich formalize the problem of multi-instance learning using drug activity prediction as an example and propose the first MIL algorithm that directly solves the MIL problem, in contrast to earlier approaches that simply converted a multi-instance problem to a single-instance one.

Although Compass and APR algorithms had proven that consideration of the MIL problem can improve the performance of models for predicting the bioactivity of compounds, MIL algorithms had not become ubiquitous. In [11] Inductive Logic Programming (ILP) approach was used to learn pharmacophores formulated as logical rules, which are used to encode conformations as a binary vector, in which 1 means that the conformation satisfies a specified rule, that is has a corresponding pharmacophore. As a result, the molecule was represented by a set of conformers encoded by binary pharmacophore features, then multi-instance regression was used to construct a linear model. The prediction of the bioactivity of a molecule can be obtained by weighted averaging of the predicted activity of its conformations. The authors tested their approach on three datasets on the activity of dopamine agonists, thermolysin inhibitors, and thrombin inhibitors and demonstrated that the models built on the multiple conformers outperform single-conformer models in all three cases.

The popular multiple-instance learning via embedded instance selection (MILES) algorithm was applied to construct models for the classification of bioactive chemical compounds [9]. MILES was applied to model the bioactivity of molecules against GSK-3, P-gp, and CB₁ receptors and demonstrated competitive with analogous approaches performance. MILES can inherently identify key instances, which can be exploited to recognize bioactive conformations. For 10 of the 12 test molecules from the GSK-3 dataset, the MILES model was able to rank the experimental bioactive conformation higher than the generated conformations. In a later paper [8], the authors proposed a modification of the MILES algorithm based on the joint instance and feature selection. The proposed approach demonstrated slightly lower classification accuracy than the original MILES, but could efficiently select a representative subset of instances and features.

Recently, the results of this Ph.D. thesis were published in a series of studies [12–14] devoted to modeling the bioactivity of compounds using conformer ensembles and multi-instance algorithms. In paper [14], an adaptation of the algorithm of Zhou and Zhang [104] was proposed to build 3D multi-conformer classification models, which were compared with traditional 2D models. A comparative analysis on a collection of >150 datasets extracted from the ChEMBL-23 database showed that 2D models outperformed 3D multi-conformer models in most cases. Nevertheless, 2D and 3D models are comparable when the dataset size is less than 1000 compounds.

Catalysts enantioselectivity modeling with conformation ensembles. In 2021 D. MacMillan and B. List received the Nobel Prize for the development of asymmetric organocatalysis. In 2000 [105,106] they contemporaneously demonstrated that small chiral organic molecules can catalyze asymmetric reactions to produce enantiopure compounds. The design of new chiral catalysts is based on the iterative improvement of the reaction enantiomeric purity by reasonable modification of the catalyst structure. This process is guided by the chemical intuition and background knowledge of the experimentalist and often culminates in the desired performance of the reaction. However, computational approaches, such as quantum chemistry [107] and chemoinformatics are especially attractive and can be used for screening virtual libraries of candidate catalysts, reducing the time and overheads needed to discover highly enantioselective catalysts.

In Quantitative Structure-Selectivity Relationships (QSSR) approach descriptors encoding catalysts structures are correlated with their experimental enantioselectivities using machine learning algorithms. The earliest studies on QSSR are based on Molecular Interaction Fields (MIF) approaches such as CoMFA [108,109]. The main problems of MIF-based 3D «structure-selectivity» modeling approaches are (i) the selection of catalyst conformers and (ii) their alignment. The selection of irrelevant conformers can reduce model performance and alignment of conformers becomes challenging if the dataset includes catalysts with different scaffolds. In the case of alignment-independent 3D descriptors, there is also (iii) the problem of the choice of relevant descriptors. In this Ph.D. project, a new 3D-QSSR approach multi-instance learning was proposed [15].

MIL algorithms can process all available catalyst conformers, solving the problem of conformers selection. Each catalyst conformer was encoded with 3D *p_mapper* descriptors, which are independent of translation and rotation of the conformer (do not require conformers alignment). The developed 3D modeling approach was validated on the reaction of asymmetric nucleophilic addition catalyzed by chiral phosphoric acids [110] and phase-transfer asymmetric alkylation catalyzed by *cinchona* alkaloid-based catalysts [111]. The 3D multi-conformer model was compared

with the state-of-the-art 3D conformer-dependent approach published by Denmark [110], and the set of traditional 2D models based on popular 2D descriptors [112,113].

Part-to-whole ambiguity modeling

Property modeling with atoms as instances. A molecule can be represented as a set of connected atoms. In this context, the molecule is characterized by part-to-whole ambiguity, where a particular atom or group of atoms is responsible for an observable property of the molecule. Within this framework, each atom of a molecule is represented by a separate vector of atom descriptors.

Bergeron et al. [39,114] introduced a novel learning framework called **Multi-Instance Ranking** (MIRank). The proposed approach was applied to the problem of identification of metabolic sites of molecules, i.e. atomic groups from which a hydrogen atom is removed. The experimental data show only to which group the removed hydrogen atom belongs, and it is not known which hydrogen atom is removed. Each hydrogen atom was represented by a set of descriptors such as the charge, the surface area, hydrophobic moment, etc. For each molecule (box), the ranking function separates at least one instance (hydrogen) of the preferred bag (group) from the remaining instances belonging to the box. Using a dataset of 227 compounds metabolized by the enzyme cytochrome CYP3A4 [115] it was demonstrated that the MIRank model performs slightly better than the standard classification model [39]. In a later work, Bergeron et al. [114] upgraded their algorithm to analyze large datasets and validated it on an extended database of 10 CYP datasets.

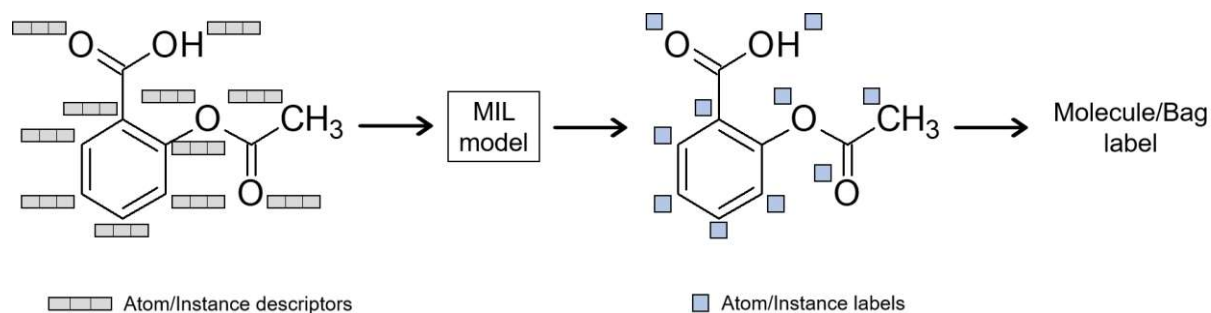


Figure 8. A general approach to multi-instance modeling of the properties of molecules represented by atom instances. Vectors of atoms can include physico-chemical or quantum-chemical descriptors or can be extracted using graph neural networks [16].

Recently, Xiong et al. [16] proposed a graph neural network based on multi-instance learning to predict both the macro-pKa of the molecule and the micro-pKa of individual atoms. In their approach, a molecule is a bag, which contains instances of the ionizable atoms of this molecule.

Each atom of the molecule is described by a vector of features extracted with a graph neural network. The extracted instance features are used to predict the micro-pKa of atoms, which are then aggregated to derive a macro-pKa. Their model predicted the acidity of organic compounds with high accuracy and provided reasonable micro-pKa of atoms.

Segment-to-sequence ambiguity modeling

Protein-protein interactions. Protein-protein interactions (PPI) play an important role in biological processes. These interactions can occur between single proteins or groups of proteins (protein complexes). In general, only particular segments of proteins (domains) determine the structure and function of the protein and are involved in the interaction between proteins. For this reason, knowledge of which domains of proteins can interact with each other enables the prediction of new protein-protein interactions.

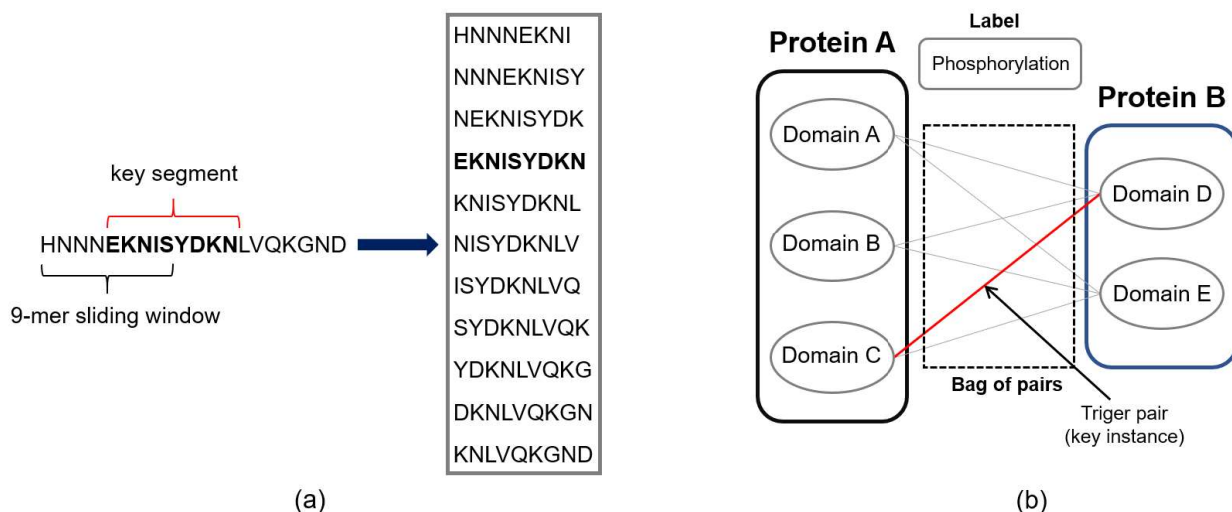


Figure 9. Macromolecule data structures (a) generating a bag from an amino acid sequence using the sliding window approach and (b) generating a bag from probable domain-domain pairs from protein-protein pairs.

Experimental PPI data provide information on the interacting protein pair and the type of interaction (activation, ingestion, phosphorylation, dissociation, etc.), but information on the interacting domains (key domains) is often not available. This scenario fits the MIL framework, where each potential domain pair is an instance (Figure 9) and the whole collection of domain pairs in a given protein-protein complex is a bag and at least one of these domain pairs interacts defining the type of interaction (e.g. phosphorylation). If the proteins do not interact, there is no pair of interacting domains in the bag.

Yamakawa et al. [116] used a dataset of 1279 PPI records labeled with ten different interaction types (state, dephosphorylation, dissociation, inhibition, phosphorylation, binding association,

indirect, activation, compound). They considered the simplified task of classification on whether the PPI is phosphorylation or not. To solve this problem, they proposed a Voting Diverse Density (VDD) algorithm based on the Diverse Density (DD) algorithm and demonstrated that their method outperformed several other popular MIL algorithms and required much less time for training [116].

Multi-domain proteins can also perform many different functions. To predict the biological functions of proteins, Wu et al. [22] used a Multi-Instance Multi-Label (MIML) framework, where instances are protein domains and the protein (bag) is associated with multiple biological functions (multiple labels). They demonstrated the applicability of the MIML approach to seven real-world datasets on the main biological systems: archaea, bacteria, and eukaryotes.

Isoform–isoform interactions. Constructing and analyzing protein-protein interactions helps to understand biological processes, enabling the development of more effective drugs. In protein biosynthesis, a gene in a DNA sequence generates a particular protein with an inherent structure and biological function. However, the alternative splicing (AS) mechanism makes it possible for the same gene to synthesize several proteins (protein isoforms) that have a similar amino acid sequence and structure but sometimes perform different biological functions. Many computational tools neglect this aspect (mainly because of the lack of experimental data on isoform-isoform interactions) and only consider the canonical (or the longest) protein derived from a gene when constructing PPIs.

This may cause interactions between canonical proteins (gene-gene interactions) to be erroneously predicted as negative (false negative), in cases where alternative proteins (isoforms) of two genes interact. This case is also suitable for the MIL framework, in which a gene (bag) generates several protein isoforms (instances). The interaction between a gene-gene pair is positive if at least one of the isoform-isoform interactions (IIIs) is positive. To address these tasks Li et al. [117] proposed a single-instance bag MIL (SIB-MIL) algorithm based on a Bayesian network classifier. SIB-MIL works at the instance level and assigns each instance (isoform pair) a probability to be positive (interactive). In SIB-MIL, the Bayesian network classifier is initially trained on positive bags with single-instance (gene pairs with single pair of isoforms) and negative instances from negative bags. The obtained classifier is then used to assign probability scores to the remaining isoform pairs in multi-instance bags. Using the obtained probability scores, a witness (key instance) is selected from each positive bag and labeled as positive. The instances with the highest probability score from the negative bags are labeled as negative. Updated labels are used to retrain the Bayesian network classifier. The instance labels are updated until the accuracy of the validation

set stops to improve. At the gene-pair level, the label of a bag is defined as the maximum probability score of its instances. Zeng et al. proposed a DMIL-III method [25] based on a deep neural network with convolutional layers. They demonstrated using a benchmark dataset that DMIL-III significantly outperforms SIB-MIL and mi-SVM algorithms.

PPIs and IIIs databases include identified interactions, whereas classification algorithms for training also require negative examples, which are usually generated artificially. This strategy often results in significantly more negative examples than positive ones, leading to imbalanced datasets. Therefore Zeng et al. [17] implemented a novel loss function to handle the imbalanced data and proposed the IDMIL-III method. They also enhanced the IDMIL-III with an attention mechanism, which improved the accuracy of identification of isoform-isoform pairs. In general, IDMIL-III improves the prediction accuracy of gene-gene pairs (bag level) in comparison to DMIL-III.

MHC-II-peptide interactions. The main function of major histocompatibility complex (MHC) protein is the binding of short peptide fragments derived from proteins produced inside (MHC-I) or outside (MHC-II) a cell and the presentation of these peptides at the cell membrane for recognition by T-cell (white blood cells of the immune system) receptors. In the context of vaccine design, it is very important to know which peptides bind to MHC molecules to initiate the desired immune response. MHC molecules have a binding groove where peptide fragments bind. MHC-I has a closed groove and usually binds peptides of lengths between 9 and 11 amino acids. In contrast to MHC-I, the binding groove of the MHC-II molecules are open at both ends and can bind peptides commonly with length from 11 to 30 amino acids [35], but it was established that for binding of protein with MHC-II is responsible a 9-mer segment of peptide and there is often no experimental information about which segment binds to the MHC-II molecule. This problem motivated studies on the application of multi-instance learning for the prediction of binding peptides.

Multi-instance learning was adapted to predict peptide binding activity to MHC-II in classification [118] and regression tasks [119]. Both approaches used bags of segments of 9 amino acids. In [21], a new multi-instance approach for predicting MHC-II binding was proposed in which flanking amino acids (11-mers) were considered in addition to the 9-mer segments. Also, the authors used experimental information that amino acids at positions 1, 4, 6, 7, and 9 may be crucial for peptide binding and integrated this information into the learning algorithm. In addition, their study revealed that amino acids at position 2 may also influence peptide binding.

Each human has multiple MHC-II molecules, which can be represented in assays. Often, experimental methods cannot precisely identify which MHC-II molecule was bound to a given peptide. Malone et al. [18] formulated the MIL problem, where the bag contains multiple MHC-II

molecules and is positive if at least one MHC-II molecule binds a given peptide and negative if there are no binding MHC-II molecules in the bag. They used a combined dataset [120] of SA (single-allele) and MA (multi-allele) data to train a transformer neural network BERTMHC and showed that models trained on SA data only are inferior to MIL models.

Calmodulin-protein interactions. Calmodulin (CaM) is a calcium-binding protein that is 148 amino acids long. CaM can interact with more than 300 proteins and peptides [121], thereby regulating many biological processes. The biological significance of CaM and the high diversity of proteins that can interact with CaM have motivated the development of computational methods for predicting both the proteins that can bind to CaM and the binding sites within these proteins.

Minhas et al. [28] used a dataset of 153 proteins with 185 experimentally annotated binding sites. In a single-instance scenario, the subsequences annotated as binding sites were marked as positive examples and all other parts of the protein (obtained using a sliding window approach) as negative. However, experimental methods do not always accurately determine the position of the binding site, which introduces ambiguity into the learning process of the classification model. Therefore, in the multi-instance model, all subsequences overlapping the binding site formed a positive bag, and all other subsequences formed a negative bag. As a result, it was demonstrated [28,122] that the MIL approach slightly improves the accuracy of binding site prediction. For CaM binding prediction, they used a dataset of experimentally identified 236 proteins that bind CaM and achieved improvement in prediction accuracy in comparison with competing methods

Modeling genomic sequences. Transcription of genes is the process of copying a DNA sequence into an RNA molecule. A Transcription Factor (TF) is a special protein that binds to a DNA sequence and activates or represses the expression of certain genes. Regions of DNA sequences that are bound by a transcription factor are called Transcription Factor Binding Sites (TFBS). Modern experimental techniques [24] enable the identification of DNA segments that are bound by the TF protein, but the precise identification of TFBS is still a challenge. Typically, a DNA sequence may contain one or more binding sites and usually, the exact location of the TF is not known (although preference information is sometimes available). Therefore, it is natural to represent the DNA sequence as a bag of possible binding sites. In the MIL classification setting, a bag (DNA sequence) is positive if it contains at least one TF and negative if it contains no TF. A bag is generated by a sliding window of length n through the whole DNA sequence. The typical length of a TF is 6-12 base pairs, which conditions the length of the subsequences (instances) included in the bag.

The *in vitro* protein binding microarray (PBM) experiments allow high-throughput screening of DNA sequences that bind to a given TF. The typical length of DNA sequences in such experiments is 35 base pairs (bp), whereas TF lengths normally vary from 6 to 12 bp. PBM data provide an excellent source for modeling TF-DNA interactions and predicting *in vivo* binding. To model *in vitro* binding, Gao and Ruan [19] used a dataset of the measured binding affinities of DNA sequences against 20 mouse TFs. This dataset was obtained from the Dialogue on Reverse-Engineering Assessment and Methods (DREAM) competition [123]. They compared SIL (whole DNA sequence) and MIL (bag of DNA subsequences) based models. For building MIL models, they used the Instance-Wrapper algorithm implemented in the WEKA package with the C4.5 decision tree as the basic single-instance algorithm. They considered each candidate binding site with a length of 5-8 ba as an instance and all possible subsequences as a bag. Consequently, the MIL model outperformed the SIL model for each of the 20 mouse TFs (average AUC score 0.94 vs. 0.71). Later Gao and Ruan [27] proposed a MIL version of the TeamD (one of the best algorithms in the DREAM5 competition) algorithm. Using a PBM dataset of 86 mouse TFs as in their previous work, they demonstrated that for 78 of the 86 TFs, MIL-TeamD outperformed SIL-TeamD (average AUC score 0.94 vs. 0.90).

Zhang continued to further improve the performance of models to predict TF-DNA binding. They considered DeepBind [29] algorithm based on a deep convolutional neural network (CNN), which has been successfully applied to predict DNA- and RNA-protein binding, and proposed its MIL version called Weakly-Supervised CNN (WSCNN). A single-instance learning algorithm (SIL-CNN), had the same architecture as DeepBind. They took the same PBM dataset of 86 mouse TFs and found that the SIL-CNN model performed better than the MIL-TeamD. However, as expected the WSCNN (MIL-CNN) model performed better than the SIL-CNN.

Another source of information on TF-DNA binding sites is *in vivo* experiments performed in living cells. Compared with *in vitro* PBM data, *in vivo* DNA sequences can be a few hundred bp (genome-scale studies) in length, which makes their experimental analysis and modeling challenges. However, DNA-protein binding models built on PBM data can be applied to predict binding DNA *in vivo* data. It was demonstrated in the works described above that MIL algorithms (MIL-TeamD, WSCNN) built on PBM or directly on *in vivo* data can significantly improve the accuracy of DNA binding predictions *in vivo* experiments.

Pan and Shen proposed the iDeepE method [26] based on MIL and deep convolutional neural networks. In their approach, instances are generated from RNA sequences using a sliding window method and the bag is positive if the RNA interacts with the protein. For validation of their method,

they used the RBP-24 dataset (<http://www.bioinf.uni-freiburg.de/Software/GraphProt>) that includes 24 experiments of 21 RNA-protein binding sites and RBP-47 which reports 502 178 binding sites for 67 RNA-protein pairs. They compared iDeepE with eight of its modifications (based on convolutional neural network, long-term memory network, and residual net) and three alternative machine learning-based approaches (GraphProt, Deepnet-rbp, Pse-SVM). The authors concluded that iDeepE performs better than its eight variants and other four state-of-the-art approaches and demonstrated that iDeepE can identify binding motifs.

RNA modification is the process by which the nucleotides in synthesized RNA are chemically modified. Traditional supervised learning approaches for predicting RNA modifications require base-resolution data, which often are not available. Huang et al. [124] proposed the weakly supervised learning framework (WeakRM) for modeling RNA modifications from low-resolution datasets. Each RNA was considered as a bag consisting of regions (instances) obtained by a sliding window approach. They examined their approach to three different types of RNA modification and demonstrated that WeakRM outperforms traditional supervised approaches and can identify regions containing the RNA modifications (key instances).

miRNA-mRNA interactions. mRNA regulates the synthesis of the peptides during gene expression, while microRNAs (short non-coding RNA with 18-25 nucleotides) binds to the specific sites of the target mRNA, and deactivates part of the mRNA or initiate its degradation and thereby inhibit gene expression. mRNA has a large number of potential binding sites (PBS) that can be bound by given miRNA, but experimental identification of functional binding sites (FBS, actual binding 2-8 nucleotide segments) is time- and money-consuming. In this context, computational approaches for predicting miRNA targets and their binding sites are highly desirable. In the MIL framework, each miRNA-mRNA pair is considered as a bag and each PBS of target mRNA as an instance. In the classification task, a bag is positive if it contains at least one FBS (key instance), and negative if there is no FBS in the bag (given that miRNA-mRNA does not interact).

Using the MIL framework, Bandyopadhyay et al. [30] developed the MBSTAR (Multiple instance learning of Binding Sites of miRNA TARgets) approaches, which is based on the MIL Random Forest algorithm (MIL-RF) and can predict both miRNA-mRNA pairs (bag predictions) and target binding sites (instance predictions). They compared MBSTAR with popular miRNA target prediction tools: TargetScan, miRanda, MirTarget2, and SVMicrO. As a result, they demonstrated that MBSTAR outperforms competing algorithms in accuracy in predicting miRNA-mRNA interactions (bag level), and especially by a large margin in predicting binding sites (instance level).

1.5 Toolkits and software

The availability of new machine learning algorithms for testing them on different problems and comparison with other algorithms is very important. Due to the rapid development of MI methods in recent decades, many of their open-source implementations in different programming languages and tools have been proposed.

WEKA [125] is a freely available software for data analysis, building machine learning models, and visualization of results of experiments. WEKA is written completely in Java and has a simple API and user-friendly graphical interface. WEKA supports several popular MI classifiers, including the aforementioned CitationKNN, Diverse Density algorithm, multi-instance extensions of SVM, and wrappers.

KEEL (Knowledge Extraction based on Evolutionary Learning) [126], is another open-source machine learning software written in Java and supported by a graphical interface. KEEL provides a set of tools for building predictive models using machine learning algorithms, including multi-instance learning algorithms. KEEL provides different variations of the APR algorithm and several popular multi-instance methods, such as EM-DD, G3PMI, CitationKNN, and methods based on evolutionary algorithms.

JCLEC (Java Class Library for Evolutionary Computation) [126] is a Java framework for evolutionary computing that is executed via the command-line interface. JCLEC provides implementation of grammar-based genetic programming (GGP) algorithm.

MATLAB implementations of multi-instance algorithms can be found in the Matlab Toolbox for Multiple Instance Learning [127]. Multiple-Instance Learning Python Toolbox [128] is inspired by MATLAB Toolbox and provides popular multi-instance algorithms written in Python.

Various multi-instance modifications of SVM [43] methods are available online in Python. Also, a lot of implementations of multi-instance deep neural networks can be obtained from GitHub repositories: classical multi-instance neural networks [12], multi-instance neural networks with attention mechanisms (<https://github.com/AMLab-Amsterdam/AttentionDeepMIL>), graph multi-instance neural networks (<https://github.com/KostiukIvan/Multiple-instance-learning-with-graph-neural-networks>), and Transformer-based multi-instance architectures (https://github.com/juho-lee/set_transformer).

Part 2. 3D structure-property modeling with multi-instance machine learning

A key limitation of traditional 3D structure-property modeling approaches is that the molecule has to be represented by a single conformer and a single vector of descriptors. The most popular strategy is to represent the molecule with a lowest-energy conformer, which, however, may differ from the true conformer that is responsible for the observed property of the molecule. The representation of molecules by irrelevant conformers makes it difficult to establish the correct relationship between the 3D structure of the molecule and its property. This problem can be solved by the application of Multi-Instance machine Learning (MIL), in which an object (molecule) is represented by a bag of instances (conformers) each encoded with its vector of chemical descriptors. Within this Ph.D. project, a new 3D structure-property modeling protocol has been developed. It is based on an ensemble of conformers and multi-instance learning algorithms, which does not require the selection and alignment of conformers. Furthermore, this 3D modeling approach generates models that not only predict the property of molecules but also can identify the key conformers responsible for the observed molecular property.

2.1 Methodological developments

This chapter provides a detailed description of the 3D structure-property modeling approach based on multi-instance machine learning.

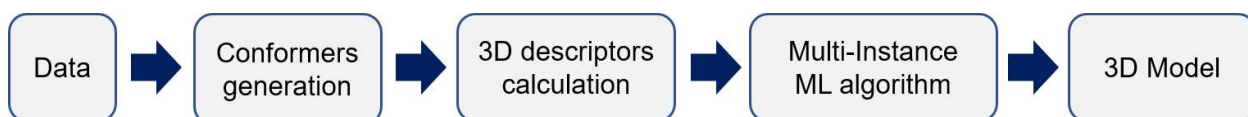


Figure 10. The pipeline of generation of 3D multi-instance models

The process of building 3D models includes several steps (Figure 10). First, for a given molecule, a set of conformers is generated which are encoded with alignment-independent 3D descriptors. The sets of 3D descriptors are then used to build the model using special multi-instance algorithms.

1) Data. The input data can be stored in any standard format, e.g. as a CSV table (Figure 11), which contains the SMILES of the molecule and the value of the target property. The implemented 3D modeling approach handles both regression and classification tasks, that is, the target property can be defined as a continuous or binary variable. The implemented MIL algorithms can

also be extended to solve multi-instance multi-task problems, where a molecule is represented by multiple instances and is associated with multiple properties that are modeled cooperatively.

	SMILES	Y
1	<chem>COc1cc2c(Nc3ccc(Br)cc3F)ncnc2cc1OCC1CCN(C)CC1</chem>	5.7
2	<chem>Cc1cc(Nc2cc(N3CCN(C)CC3)nc(Sc3ccc(NC(=O)C4CC4)cc3)n2)[nH]n1</chem>	6.2
3	<chem>COc1cc(Nc2ncc(F)c(Nc3ccc4c(n3)NC(=O)C(C)(C)O4)n2)cc(OC)c1OC</chem>	7.5
4	<chem>CS(=O)(=O)N1CCN(Cc2cc3nc(-c4ccc5n[nH]cc45)nc(N4CCOCC4)c3s2)CC1</chem>	5.6
5	<chem>CN1CCN(c2ccc3nc(-c4c(N)c5c(F)cccc5[nH]c4=O)[nH]c3c2)CC1</chem>	6.5
6	<chem>CCN1CCN(Cc2ccc(NC(=O)Nc3ccc(Oc4cc(NC)ncn4)cc3)cc2C(F)(F)F)CC1</chem>	6
7	<chem>CCN1CCN(c2ccc(Nc3ncc(Cl)c(Nc4ccc5n[nH]cc5c4)n3)cc2)CC1</chem>	7.5
8	...	...
N	<chem>O=C(O)c1csc2c1NCCNC2=O</chem>	4.7

Figure 11. Example of an input data table for building 3D MIL models

2) Conformer generation. Conformers representing each molecule were generated using the distance geometry algorithm implemented in RDKit [129], which is claimed by its authors to be able to reproduce bioactive conformations of ligands from the Protein Data Bank (PDB) database with reasonable accuracy. This algorithm is based on stochastic conformer generation which is constrained by geometric patterns derived from experimental data. Precise bond lengths, bond angles, and torsion angles are used to determine lower and upper distance bounds for all pairs of atoms in the molecule. These distance bounds are collected in a distance bounds matrix, which is used in combination with a conformation optimization using a Merck Molecular Force Field (MMFF). If the RDKit algorithm failed to generate the conformers, then a systematic conformer generator from the Open Babel package [130] is used and the full energies of obtained conformers are recalculated using RDKit.

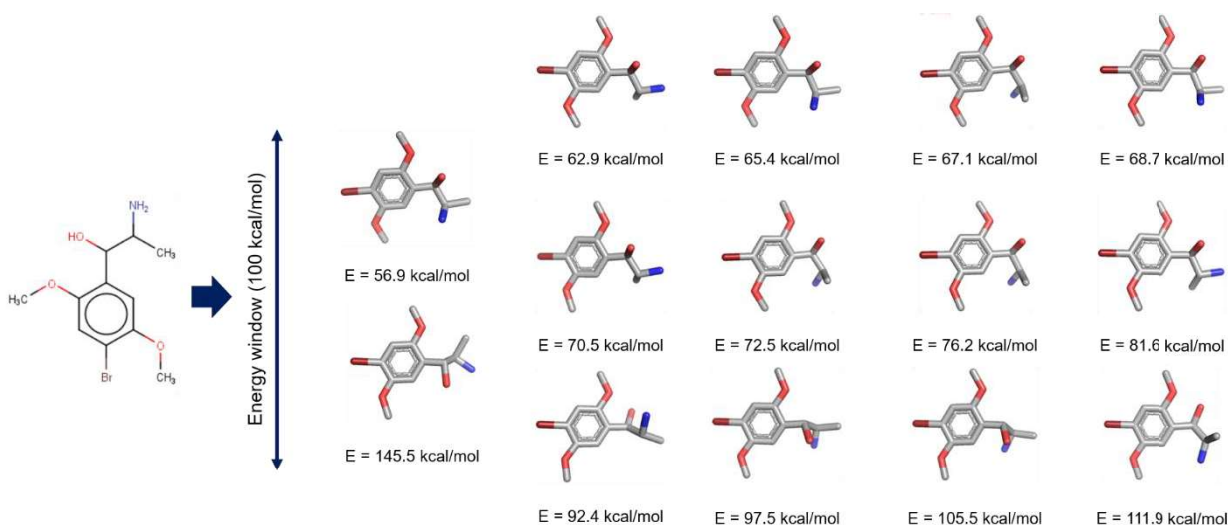


Figure 12. Conformers generated by RDKit for the example molecule. Only conformers within the energy window of 100 kcal/mol are selected for modeling.

The diversity of the generated conformers depends on the width of the energy window, which is specified manually. All conformers that differ in energy from the most stable conformer more than the width of the energy window are discarded. Conformations with RMSD values below 0.5 Å to the remaining ones are removed to reduce redundancy. Figure 12 demonstrates an example input molecule and the corresponding generated conformers using the RDKit package.

3) Descriptors. The generated conformers of the molecule then can be encoded using 3D descriptors. Several 3D alignment-independent descriptors (WHIM [131], GETAWAY [132], MORSE [133], RDF [134]), which do not depend on the translation and rotation of molecules in 3D space are implemented in RDKit. The problem with the majority of alignment-independent 3D descriptors developed so far is that not all of them can distinguish stereoisomers and not all of them are interpretable. The developed 3D approach is based on novel 3D pharmacophore descriptors [135], which are implemented in the *pmapper* package (<https://github.com/Drr-Dom/pmapper>).

Modeling biological activity
Quadruplets of pharmacophore features

```

1 # lines started from # are comments
2 #
3 # each line is SMARTS and feature label
4 #
5 # legend of feature labels:
6 # a - aromatic
7 # A - H-bond acceptor
8 # D - H-bond donor
9 # H - hydrophobic
10 # P - positive
11 # N - negative
12
13 # aromatic
14 alaaaaal a
15 alaaaaal a
16
17 # HBD
18 [#7]H0&12 (N-[BX4] (=O) (=O) [CX4] (F) (F) F) D
19 [#8]H0&12 ([OH] [C, S, P] =O) D
20 [#4]H0 D
21
22 # HBA
23 [#7&12 ([NX3]) &12 ([NX3] - *([#6]) &12 ([NX3] - [a]) &12 ([NX4]) &12 (N=C([C, N]) N)) A
24 [#5([O]) &12 ([OX2] (C) C=O) &12 (*(-a) -a)) A
25
26 # positive
27 [#5([NX3] ([CX4]) ([CX4, #1]) [CX4, #1]) &12 ([NX3] - *([#6]) P
28 [#5([CX3] (#N) (-N) ([N]) (#N) -N) P
29 N=[CX3] (N) -N P
30 [#5([+, +2, +3]) &12 (*[-, -2, -3]) P
31
32 # negative
33 cinn[nH]n1 N
34 [#5([RX4, RX4] (=O) (=O) [O-, OH]) (=O) [O-, OH] N
35 [#5([CX3, RX3, RX3] (=O) [O-, OH]) (=O) [O-, OH] N
36 [#5([-, -2, -3]) &12 (*[+, +2, +3]) N
37
38 # hydrophobic
39 alaaaaal H
40 alaaaaal H
41
42 [#5([CH3X4, CH2X3, CH1X2, F, Cl, Br, I]) &12 (*([CH3X4, CH2X3, CH1X2, F, Cl, Br, I]) H
43 *([CH3X4, CH2X3, CH1X2, F, Cl, Br, I]) ([CH3X4, CH2X3, CH1X2, F, Cl, Br, I]) ([CH3X4, CH2X3, CH1X2, F, Cl, Br, I]) H
44 [C&#31]-[C&#31]-[C&#31]-[C&#31] H
45 [C&#31]-[C&#31]-[C&#31]-[C&#31]-[C&#31] H
46 [C&#31]-[C&#31]-[C&#31]-[C&#31]-[C&#31]-[C&#31] H
47 [C&#31]-[C&#31]-[C&#31]-[C&#31]-[C&#31]-[C&#31]-[C&#31] H
48 [C&#31]-[C&#31]-[C&#31]-[C&#31]-[C&#31]-[C&#31]-[C&#31]-[C&#31] H
49 [CH2X4, CH1X3, CH0X2] - [CH3X4, CH2X3, CH1X2, F, Cl, Br, I] H
50 [#5([CH2X4, CH1X3, CH0X2] - [#5([#1]), #5([CH2X4, CH1X3, CH0X2])]) - [CH2X4, CH1X3, CH0X2] - [CH2X4, CH1X3, CH0X2] H

```

(a)

Modeling catalyst enantioselectivity
Triplets of atoms features

```

1 # lines started from # are comments
2 #
3 # each line is SMARTS and feature label
4 #
5 # legend of feature labels:
6 # a - aromatic
7 # A - H-bond acceptor
8 # D - H-bond donor
9 # H - hydrophobic
10 # P - positive
11 # N - negative
12
13 # aromatic
14 alaaaaal a
15 alaaaaal a
16
17 # atoms
18 [C, N, O, S, P, F, Cl, Br, I] H
19

```

(b)

Figure 13. Examples of input files containing SMARTS of combinations of atoms that are encoded for a given 3D structure: (a) pharmacophore features used to build 3D models for prediction of bioactivity of molecules and (b) individual atom features used to build 3D models for prediction of catalyst enantioselectivity.

In the default setting of the *pmapper* package, each conformation is encoded by a set of pharmacophore features (H-bond donor/acceptor, the center of positive/negative charge, hydrophobic, and aromatic) determined by the corresponding SMARTS notation. For a given conformation, all possible quadruplets of predefined features were enumerated. Distances between features are binned to allow fuzzy matching of quadruplets with small differences in the position of features. In the default setting binning step of 1 Å is used as it demonstrated reasonable performance in previous studies [15,135–137]. Then 3D pharmacophore signatures are generated for each quadruplet according to the algorithm in details described in the original publication [135]. These signatures encode distances between features and their spatial arrangement to recognize the stereo configuration of quadruplets. The number of identical 3D pharmacophore quadruplet signatures is counted for each conformation and the obtained vectors are used as descriptors for model building.

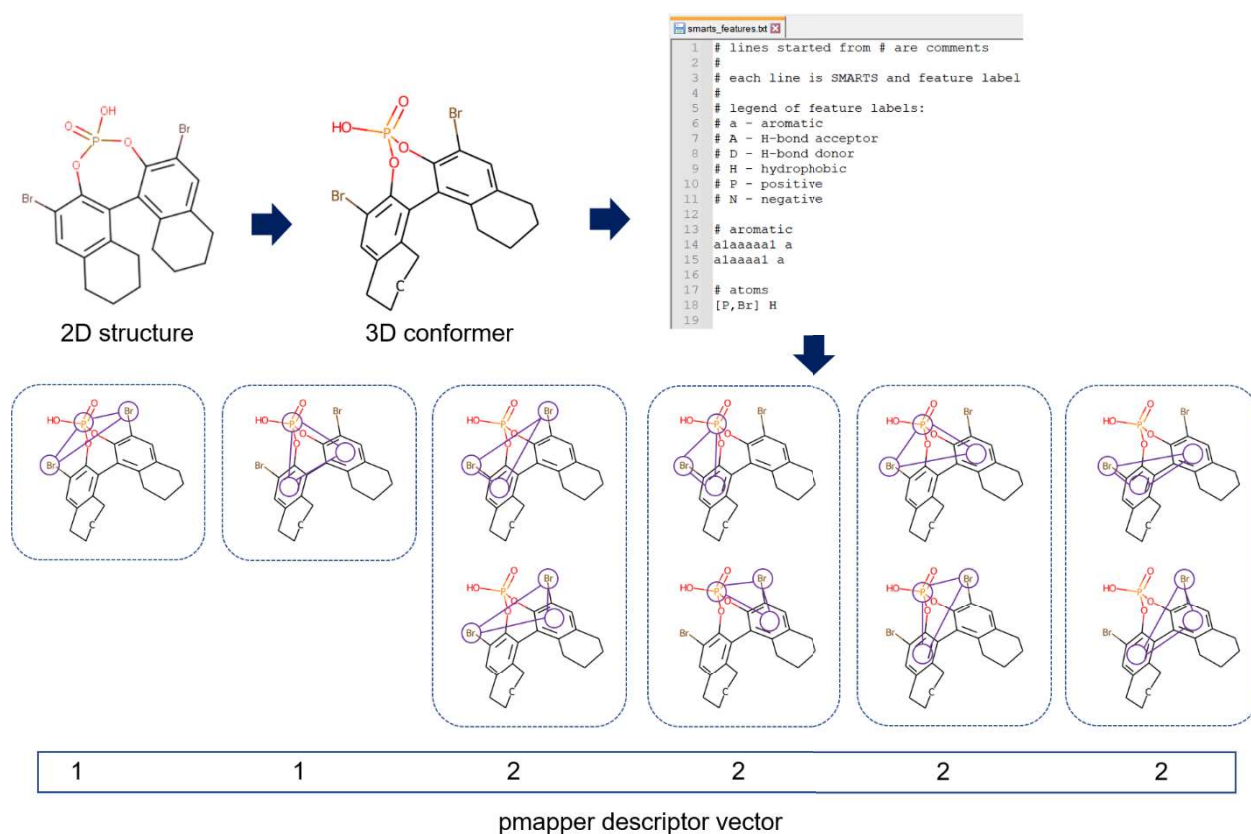


Figure 14. An example of the calculation of the pmapper descriptor vector for a phosphoric acid catalyst. For demonstration, combinations of three features (SMARTS:alaaaaal (aryl) and [P, Br]) were set in the input file.

However, the *pmapper* descriptors are customizable and any combination of atoms and groups of atoms that encode the relevant 3D patterns in a given structure can be used instead of

the default pharmacophore features. For example, the original 3D pharmacophore descriptors (Figure 13a) were used in this project to model the bioactivity of compounds extracted from the ChEMBL-23 database. In another study on modeling the enantioselectivity of chiral organic catalysts, more abstract triplets of individual atoms were chosen (Figure 13b, Figure 14).

4) Multi-instance learning algorithms. The implemented in this research MIL algorithms can be divided into two groups. The first group includes two wrapper algorithms (Instance-Wrapper and Bag-Wrapper), which transform a multi-instance dataset into a single-instance dataset that can be processed by any traditional single-instance machine learning method. The second group of algorithms includes MIL algorithms that can directly process a multi-instance dataset. These algorithms are either adaptations of traditional ML algorithms, or algorithms specially designed [5] to solve MIL problems. In this project, MIL adaptations of neural networks (Instance-Net, Bag-Net, and BagAttentionNet) were implemented and tested in several studies. A basic component of some MIL algorithms is a pooling operator that aggregates instances (bag-level algorithms) or instance predictions (instance-level algorithms). The pooling operators used in this study were *mean*, *max*, *log-sum-exp*, and *attention-based* pooling.

Traditional pooling operators. In bag level algorithms *mean* pooling aggregates instances by averaging the instance vectors resulting in an embedding vector, which is used for predicting bag labels. In instance-level algorithms, *mean* pooling averages instance predictions to produce a bag prediction. *Max* pooling selects the max value of each descriptor across all instance vectors in bag-level algorithms or the max value of instance predictions in instance-level algorithms. The convex version of *max* pooling is the *log-sum-exp* operator [68].

Attention-based pooling operators. Key instances define the observed bag label. In the context of modeling the bioactivity of molecules with MIL approaches, it is considered that a molecule is bioactive if at least one of its conformers is bioactive (binds to the target), and inactive if none of the conformers is bioactive. Therefore, it is desirable not only to predict molecule property but identify key conformations responsible for observed target property.

Traditional pooling operators (*mean*, *max*) are predefined and ignore the importance of individual instances. This motivated the development of advanced pooling operators that adapt during training and focus on the most important instances. In bag-level algorithms, these pooling operators generate instance weights, which determine the contribution of each instance to the final embedding vector. Such pooling operators are especially desirable because they make MIL models interpretable, i.e., they allow not only the prediction of a bag label but also the identification of key instances.

Attention-based pooling. In [75] Ilse et al. proposed a pooling operator based on the attention mechanism [90], which was implemented as a two-layered neural network followed by the *softmax* function that receives instance scores and generates instance weights that sum to 1 (the higher the instance attention weight, the more important the instance). Instances are then aggregated according to the attention weights (weighted mean). In this project, the attention neural net is coupled with a fully-connected three-layered neural network, which generates instance representations and predicts bag labels based on bag embedding. Both neural networks are trained consistently using a backpropagation algorithm.

GatedAttention-based pooling. The default version of the Attention-based neural network includes a tangent hyperbolic activation function (*tanh*), which is approximately linear for x in the range of $[-1, 1]$. Therefore, in the same paper [75] Ilse and co-workers also proposed to use of a gating mechanism [138] to increase the non-linearity of learned relationships. GatedAttention-based pooling consists of two neural networks: one with a *tanh* and another with a *sigmoid* activation function and the resulting representation is calculated as element-wise multiplication $\tanh \odot \text{sigmoid}$.

Self-attention pooling. Attention-based MIL pooling is flexible and suitable for aggregating information from individual instances. However, the contribution of each instance in the label of the bag is evaluated by the attention neural network independently of the other instances in the bag. This is an acceptable scenario when considering a standard assumption, where a bag is given a positive label if it contains at least one positive instance. More complicated is the threshold-assumption, when a bag is positive only when it contains at least N positive instances. The Presence-based assumption assumes that a bag is positive if it contains several instances of different concepts. For example, the standard assumption is suitable for predicting the bioactivity of a compound represented by multiple conformations, since a compound is active if at least one of its conformations is bioactive, i.e. binds to the target. Another example relates to the presence-based assumption. Let a compound is active when it contains an amide group, which consists of C, O, and N atoms. In this case, the MIL method must be forced not only to identify the C, O, and N atoms separately but also to be sensitive to cases when instances representing atoms C, O, and N occur in the bag simultaneously. To handle tasks in which threshold - and presence-based assumptions prevail, more advanced pooling types are needed. These pooling functions must take into account interactions between instances in the bag.

One of the approaches to solving this problem is to apply the *self-attention* mechanism. The main idea of *self-attention* is to take into account the similarity between instances when calculating the attention weights of bag instances. Thus, the weight of each instance depends on the

composition of the bag, i.e. the presence of other instances in the bag. A possible architecture of a MIL neural network combines a *self-attention* mechanism with attention-based pooling. First, the input bag runs through a set of fully connected neural network layers, resulting in learned representations of the instances. Next, the *self-attention* layer accepts the representations of the bag instances as input and outputs new vectors of instance features, which contain information about the interdependencies of instances. New vectors of instances generated by the *self-attention* layer are fed into attention-based pooling, which aggregates them into an embedding vector under the attention weights of the instances.

Attention weights regularization. Since only a few instances are responsible for the observed bag label, the distribution of attention weights across instances is supposed to be sparse and sharp, i.e., the attention mechanism must focus mainly on the key instances. The sparsity requires that most of the attention weights are close to 0.0. The sharpness requires that the attention weight of the key instances should be as high as possible. However, examples from other machine learning tasks [139] and preliminary results obtained in this research project demonstrate that the standard version of the attention mechanism tends to generate uniformly distributed attention weights with a poor focus on key instances. This motivated the development of regularization techniques that constrain the weights distribution, forcing the attention mechanism to focus on the fewest instances. Details of the regularization techniques implemented within this project are provided in this section.

Temperature softmax. In the standard attention mechanism, the weight α_i of instance i is calculated using the softmax function:

$$\alpha_i = \frac{\exp(z_i)}{\sum_{j=1}^K \exp(z_j)} \quad (2)$$

The modification of standard *softmax* is a *temperature softmax*, which includes the parameter of temperature $T > 0$:

$$\alpha_i = \frac{\exp(z_i/T)}{\sum_{j=1}^K \exp(z_j/T)} \quad (3)$$

The lower the T value, the sharper the attention weights distribution, and the higher the T value, the more uniform the distribution. At $T = 1$, the *temperature softmax* is identical to the standard *softmax*.

Gumbel-Softmax. Originally, the *Gumbel-Softmax* function was proposed by Jang et al. [140] to provide a continuous approximation to sampling from the categorical distribution in a way that is differentiable and suitable for backpropagation algorithm in deep learning:

$$\alpha_i = \frac{\exp((\log(z_i) + g_i)/T)}{\sum_{j=1}^K \exp((\log(z_j) + g_j)/T)} \quad (4)$$

Gumbel-Softmax combines the deterministic part of sampling with the stochastic part g by adding Gumbel noise $(0, 1)$, which can be sampled as two logs of some uniform distribution.

Minimum Entropy Regularizer. In the attention-based mechanism sparse and sharp weights distribution has low entropy, which is calculated as:

$$Entropy = - \sum_i^K \alpha_i \log(\alpha_i) \quad (5)$$

Thus, minimizing the entropy of attention weights during the training of the neural network forces the attention mechanism to generate a sharp attention weights distribution.

Attention weights dropout. In *attention weights dropout* the weights generated by the attention mechanism are sorted and $N\%$ (N is set manually) of the instances with the lowest attention weights are discarded. The attention weights of the remaining instances are recalculated again using a softmax so that they sum to 1. As a result, only a fixed number of instances with the highest attention weights contribute to the embedding vector.

Other pooling operators. There are other types (non-attention) of pooling operators that can estimate instance weights.

Gaussian weighting. Another type of pooling based on an additional neural network is pooling with Gaussian weighting [95]. Gaussian pooling applies a Gaussian radial basis function to calculate instance weights, which is the main difference from attention-based pooling, which applies *softmax* for this purpose. As a result, each weight can independently take values from 0 to 1. This variant of pooling can be considered soft pooling in comparison with attention-based one.

Dynamic pooling. Inspired by the *Routing Algorithm* from *Capsule Networks* [81], a new type of pooling operator was proposed in [80], called dynamic pooling. This pooling operator iteratively updates the instance contribution to its bag representation during each feed-forward step. Based on these instance contributions, dynamic pooling highlights the key instance and models the contextual information among instances. The multi-instance neural network with dynamic

pooling is optimized with the *margin loss* in an end-to-end manner. Besides the ability to highlight the key instance, the dynamic pooling function makes instance-to-bag relationships interpretable.

5) Model optimization. The developed 3D modeling protocol is fully automated, but some parameters of this protocol (Table 1) can be configured manually for each particular task. Table 1 lists recommended values for the parameters of the modeling protocol which were obtained based on preliminary experiments, except for the parameter «Feature composition (input SMARTS)», which has to be specified for each task or kept as default.

Table 1. The main parameters of the developed 3D multi-instance modeling protocol.

Parameters		Default value
Conformer generation		
Number of conformers	From 1 to 200 (or more)	50 or 100
Energy window	From 10 to 100 kcal/mol	100 kcal/mol
Pmapper Descriptors		
Number of feature points	Atom pairs (2), triplets (3), quadruplets (4)	Quadruplets (4)
Binning parameter	1 or more (less probable)	1
Feature composition (input SMARTS)	Any combinations	[C, N, O, S, P, F, Cl, Br, I]
MIL algorithm		
Descriptors scaling	No or Yes	Yes
Type of algorithm	Instance-Wrapper, Bag-Wrapper, Instance-Net, Bag-Net, BagAttention-Net, BagDynamic-Net, etc.	Instance-Wrapper

6) Software. The developed 3D modeling protocol is based on open-source packages available using Python 3. The in-house modules of the modeling protocol are also written in Python 3. The program code for the developed modeling protocol was organized in a *miqsar* python package (<https://github.com/cimm-kzn/3D-MIL-QSAR>) (Figure 15).

MIL Wrappers. The simplest algorithms that convert a multi-instance dataset into a single-instance dataset. Then any standard ML algorithm is used to build the model (standard neural network as default).

Instance-Wrapper. The algorithm transforms a multi-instance dataset into a single-instance dataset by assigning all instances labels of the parent bag. Then any single-instance ML algorithm is used to build the model. For a new object, the predictions of each instance are obtained, which are then averaged to get the bag prediction.

Bag-Wrapper. The algorithm transforms a multi-instance dataset into a single-instance dataset by mapping (i.e. averaging) a bag of instances to a single embedding vector. Then any single-instance ML algorithm is used to build the model.

Upgraded MIL algorithms. These are multi-instance adaptations of the SVM algorithm (MISVM, miSVM, NSK, STK, MissSVM, MICA, sMIL, stMIL, sbMIL) published by Doran and Ray [43] (<https://github.com/garydoranjr/misvm>).

MI neural networks. Neural networks adapted to MIL framework.

Instance-Net. Hidden layers of neural networks transform instance features into instance representations, from which instance scores are derived, that are aggregated to final bag prediction.

Bag-Net. Hidden layers of neural networks transform instance features into instance representations, that are aggregated by pooling operator to a single embedding vector, which are processed to derive bag prediction.

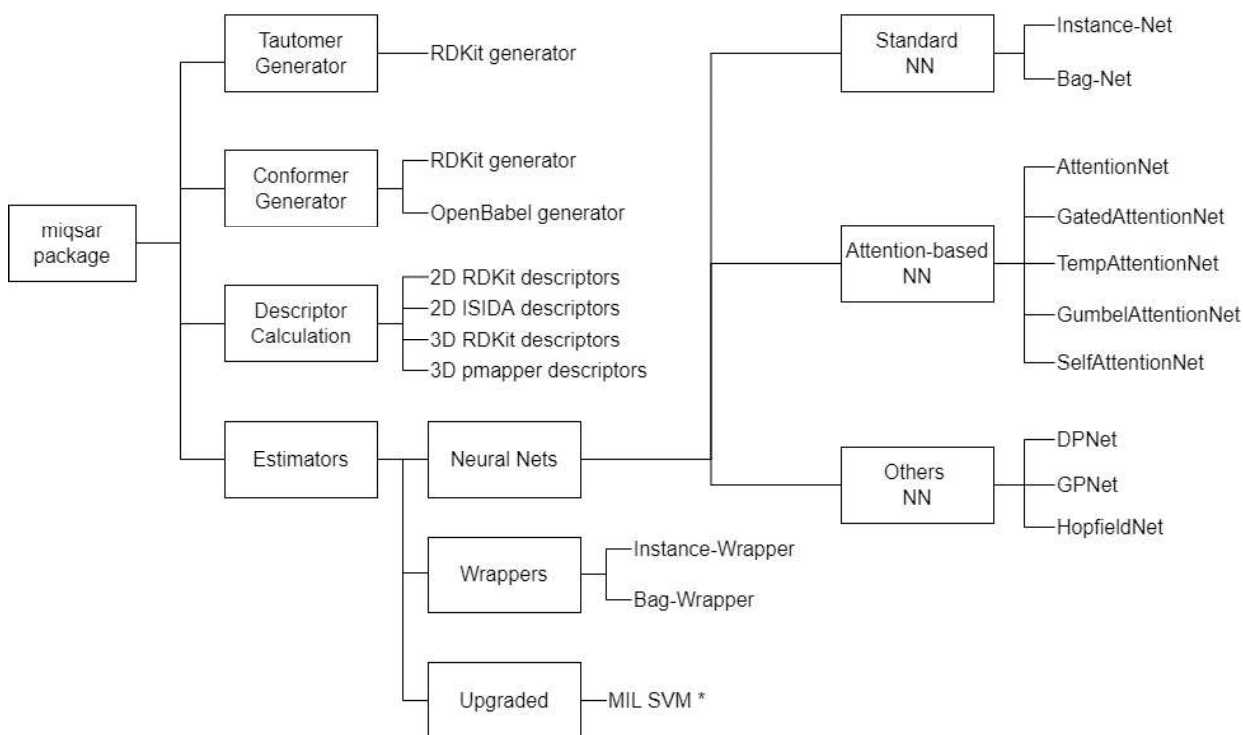


Figure 15. Structure of the *miqsar* package for building 3D models using machine learning algorithms.

AttentionNet. Hidden layers of neural networks transform instance features into instance representations, that are aggregated by an *attention-based pooling* operator (weighted mean) to a single embedding vector, that is processed to derive bag prediction.

GatedAttentionNet. Two types of hidden layers are used to transform instance features into instance representations: one with a *tanh* and another with a *sigmoid* activation function and the resulting instance representations are calculated as element-wise multiplication $\tanh \odot \text{sigmoid}$.

Instance representations that are aggregated by an attention-based pooling operator (weighted mean) to a single embedding vector, are processed to derive bag prediction.

TempAttentionNet. The algorithm applies *temperature softmax* instead of a standard *softmax* function to calculate attention weights in attention-based pooling. The temperature parameter is used to adjust the sharpness of attention weights distribution.

GumbelAttentionNet. The algorithm applies *Gumbel softmax* instead of a standard *softmax* function to calculate attention weights.

SelfAttentionNet. Hidden layers of neural networks transform instance features into instance representations, that are aggregated by a *self-attention-based pooling* operator to a single embedding vector, that are processed to derive bag prediction.

DPNet. Hidden layers of neural networks transform instance features into instance representations, that are aggregated by a *dynamic pooling* operator to a single embedding vector, that are processed to derive bag prediction.

GPNet. Hidden layers of neural networks transform instance features into instance representations, that are aggregated by the *gaussian weighting pooling* operator (weighted mean) to a single embedding vector, that are processed to derive bag prediction.

2.2 Multiple conformer descriptors for QSAR modeling

Multi-instance algorithms can be categorized into instance-based and bag-based algorithms. Instance-based algorithms apply a predefined rule to aggregate the predicted instance scores to obtain a single prediction for the entire bag. Bag-based algorithms aggregate instances of the bag into a single vector, resulting in single-instance representation. Mapped bag-based algorithms use a special mapping function, to transform multi-instance data into single-instance representations of bags. Mapping methods can be based on bag statistics, representative instance concatenation, counting, or distance [42]. In this study MIL-kmeans algorithm, which is similar to the approach published by Zhou and Zhang [104] was developed and validated for the task of classification of bioactive compounds.

In MIL-kmeans algorithm, all conformers of all compounds represented by corresponding 3D descriptors are clustered using the k-means algorithm. The obtained clusters are used to generate a new descriptor vector of a given compound (mapping process): the descriptor value was equal to 1 if at least one conformer of the molecule fell into the corresponding cluster or 0 otherwise. As a result, a new descriptor matrix of the size (the number of molecules) $\times$ (the number of clusters) is generated. Any conventional regression or classification machine learning algorithm then can then be applied to build models based on this descriptor matrix. Two approaches were considered as alternatives for comparison. MIL-mean algorithm averages the descriptor vectors of conformers transforming multi-instance data to single-instance data and applies the Random Forest algorithm to build a model. The MIL-max approach also transforms data to single-instance representation by a selection of the maximum value of each descriptor over conformers of a particular compound and then applies the Random Forest algorithm to build a model.

3D MIL classification models based on the proposed MIL algorithm were compared with single-conformer models and 2D models based on 2D descriptors available in RDKit (Morgan fingerprints, pharmacophore fingerprints, and physicochemical descriptors). The comparison was performed on three types of datasets extracted from the ChEMBL-23 database: (i) collection of 6 chiral datasets containing only chiral molecules, (ii) collection of 5 achiral datasets containing only achiral molecules, and (i) collection of 162 datasets, including both chiral and achiral molecules. Compounds were labeled active if their pKi or pIC50 was ≥ 6 for enzyme targets and ≥ 7.5 for membrane proteins, and inactive otherwise.

Multiple Conformer Descriptors for QSAR Modeling

Aleksandra Nikonenko,^[a] Dmitry Zankov,^[b] Igor Baskin,^[c] Timur Madzhidov,^{*,[b]} and Pavel Polishchuk^{*,[a]}

Abstract: The most widely used QSAR approaches are mainly based on 2D molecular representation which ignores stereoconfiguration and conformational flexibility of compounds. 3D QSAR uses a single conformer of each compound which is difficult to choose reasonably. 4D QSAR uses multiple conformers to overcome the issues of 2D and 3D methods. However, many of existing 4D QSAR models suffer from the necessity to pre-align conformers, while alignment-independent approaches often ignore stereoconfiguration of compounds. In this study we propose a QSAR modeling approach based on transforming chirality-aware 3D pharmacophore descriptors of individual con-

formers into a set of latent variables representing the whole conformer set of a molecule. This is achieved by clustering together all conformers of all training set compounds. The final representation of a compound is a bit string encoding cluster membership of its conformers. In our study we used Random Forest, but this representation can be used in combination with any machine learning method. We compared this approach with conventional 2D and 3D approaches using multiple data sets and investigated the sensitivity of the approach proposed to tuning parameters: number of conformers and clusters.

Keywords: 4D QSAR · multiple instance learning · 3D pharmacophore descriptors

1 Introduction

The quantitative structure-activity relationship (QSAR) modeling is a universal approach applicable for predicting activity of compounds as well as their side effects, ADME, toxicity, metabolites, physicochemical, and other properties.^[1] The classical methodology of building QSAR models encodes each molecule as a set of descriptors and then applies machine learning to find the correlation between descriptors and investigated activity. This gives rise to one of the key limitations of conventional structure-property modeling: the requirement that each molecule has to be represented by a single instance with fixed conformation, protonation state, tautomeric form, etc. In other words, a molecule has to be associated with a single set of descriptors. However, molecules are dynamic objects and simultaneously may exist in many forms (conformational, tautomeric, protonation states, mixtures of stereoisomers, etc.) in equilibrium. These forms are important for their biological response or physicochemical properties. For example, only particular conformers of a compound may bind to a protein target to produce desired response.^[2] Spatial configuration of compounds can also affect their biological activity. Biological activity of different stereoisomers may differ up to several orders of magnitude.^[3] About half of marketed drugs are chiral compounds^[4] and about 25% of them have several tautomeric forms.^[5] The same compound may bind in different tautomeric forms if the energy difference between them is less than 1 kcal/mol.^[6] Representation of each molecule as a single fixed instance neglects complexity of molecular objects. Thus, choosing the proper instance to represent a molecule becomes important and may affect model accuracy.

There are many workarounds to overcome this issue. For example, the recommended approach for treatment of tautomers is canonicalization and using a single canonical tautomer for each compound in QSAR modeling.^[7] However, this canonicalization is mainly formal and does not take into account stability of tautomers. It is difficult to accurately predict tautomer stability in conditions, similar to those in which biological assays are conducted. Some compounds, e.g. sydnones, cannot be represented by a single canonical structure at all. The similar workaround is applied to represent protonation states of a molecule – the most abundant (by prediction) protonation state is chosen. Thus, if a compound exists in several protonation states with comparable abundance only one will be chosen reducing information supplied to a model. Bonachera et al. proposed to cope with this problem by weighting pharmacophore descriptors for different protonation states and tautomers according to their stability,^[8] but later problems

[a] A. Nikonenko, P. Polishchuk
Institute of Molecular and Translational Medicine, Faculty of Medicine and Dentistry, Palacký University and University Hospital in Olomouc, Hnevotinska 5, 77900 Olomouc, Czech Republic
E-mail: pavlo.polishchuk@upol.cz

[b] D. Zankov, T. Madzhidov
A.M. Butlerov Institute of Chemistry, Kazan Federal University, Kremlevskaya Str. 18, 420008 Kazan, Russia
E-mail: timur.madzhidov@kpfu.ru

[c] I. Baskin
Department of Materials Science and Engineering, Technion-Israel Institute of Technology, 3200003 Haifa, Israel

Supporting information for this article is available on the WWW under <https://doi.org/10.1002/minf.202060030>

with quality of tautomer prediction of applied tools were reported.^[9]

Treatment of stereoisomers is more elaborated, and it also has some issues. At the 2D level encoding of stereo-configuration is complicated because molecules are represented as molecular graphs and explicit information about conformation and spatial arrangement of atoms is absent. Different approaches were suggested to overcome this issue. However, all developed approaches of stereochemistry encoding at the 2D level are mainly limited to represent chiral centers and cis-trans isomerism of double bonds.^[10–15] At the 3D level of representation molecules are considered as particular conformers. This enables more natural encoding of compound chirality by direct incorporation of spatial arrangement of atoms in the course of calculation of chirality-aware descriptors. However, the biggest issue of 3D QSAR modeling approaches is the selection of proper conformers, because this greatly determines success of modeling. This can be partially overcome if one knows or can reasonably suggest conformation of a compound responsible for studied activity. Therefore, in the first publications on 3D modeling researchers used conformationally rigid molecules as templates to align other molecules of a dataset. This was implemented in the first 3D QSAR approach Comparative Molecular Field Analysis (CoMFA).^[16] It suffered from atom-based alignment of structurally diverse compounds that limits its applicability to mainly congeneric compound series. Many other 3D QSAR approaches were developed since then, e.g. Molecular Shape Analysis (MSA),^[17] GRID,^[18] Hypothetical Active Site Lattice (HASL),^[19] Comparative Molecular Similarity Indices Analysis (CoMSIA),^[20] Comparative Molecular Surface Analysis (CoMSA),^[21] Continuous Indicator Fields.^[22] But to some extent all of them suffered from the alignment issue inherent to CoMFA. Many alignment-independent 3D approaches were developed to overcome this issue. One of them, Comparative Molecular Moment Analysis (CoMMA), suggested to use zeroth, first-, and second-order spatial moments of the charge and the mass distribution as descriptors in 3D QSAR studies instead of interaction fields.^[23] Other alignment-independent 3D descriptors include WHIM,^[24] GETAWAY,^[25] MORSE,^[26] RDF,^[27] etc. Application of alignment-independent descriptors does not solve the issue related to proper selection of conformers for modeling. They do not encode stereoconfiguration of compounds which can be, in principle, reflected in alignment-dependent 3D QSAR approaches.

4D QSAR approaches were developed to enable representation of a single compound by a set of conformers.^[28–29] These approaches can also be divided into alignment-dependent and alignment-independent ones. The first alignment-dependent 4D approach stochastically searched for the best alignment among generated conformer ensembles using genetic algorithm.^[28] However, this modeling strategy is not feasible for large datasets which are quite common nowadays. SOM-4D-QSAR is another

alignment-dependent method. It takes pre-aligned conformers and maps them to 2D self-organized Kohonen maps. Occupancy of neurons or mean charges are used to build PLS models.^[30–31]

Another group of 4D approaches calculates 3D alignment-independent descriptors for individual conformers and combines them in order to obtain a single vector of compound descriptors. The most widely used combining schemes are summation or averaging of descriptors of individual conformers or summation, weighted by a normalized Boltzmann distribution of conformers by energy.^[29,32–33] But application of such schemes looks somewhat artificial. First, only few conformers can fit a binding pocket. Thus, conformers do not contribute equally to the activity. Second, the distribution of conformers by their energy calculated in vacuum or even in water solution does not necessarily resemble their distribution in protein-bound state.

The approach of multi-instance learning (MIL) was proposed by Dietterich et al.^[34] to address the issue of representation of compounds by multiple instances, in particular, conformers. The central idea is that each molecule can be represented by a bag (set) of instances. Each bag is associated with activity value but it is unknown which instances contribute to the activity. Each instance is represented by a vector of descriptors and the task is to build a model that finds correlation between the set of vectors corresponding to the instances of the bag and the end-point value associated with this bag. This idea did not receive much attention in chemoinformatics community and only few papers were published so far.^[35–39] But it attracted much attention in other fields, like text or signal processing, information retrieval, computer vision, etc.^[40]

There are two major groups of MIL modeling approaches: instance-based and bag-based.^[41] Instance-based methods classify each instance individually and combine the predicted instance labels to assign a bag label. Bag-based approaches operate by whole bags of instances and assign labels to the bags. The latter group can be divided into two subgroups. The first one is based on calculation of similarity/distances between bags. This was implemented in methods based on k-nearest neighbors algorithm^[42–43] or SVM-based algorithms.^[44] The second subgroup includes embedding-based approaches, which transform a set of feature vectors of individual instances of a bag into a single feature vector representing the whole bag.^[45]

It is important to note that conventional 4D QSAR methods are a subset of embedding-based MIL approaches. Descriptors of individual conformers in 4D QSAR are averaged or summed up to create a single feature vector representing the set of conformers. This is one of trivial embedding schemes. An example of more advanced scheme is unsupervised embedding implemented in SOM-4D-QSAR. Conformers of compounds are projected to a new space which is used for model building. The drawback of SOM-4D-QSAR implementation is that it requires com-

pound alignment and thus it is applicable to congeneric series only.^[30–31]

In this study we propose a new embedding scheme similar to SOM-4D-QSAR. It is based on clustering of compound conformers. Cluster membership of conformers of a compound is encoded by the bit vector, which is then used to build a model. The major difference from SOM-4D-QSAR was the descriptors chosen, chirality-aware 3D pharmacophore quadruplets, and no need for alignment of molecules. We compare performance of the proposed approach with conventional single instance 2D and 3D models on multiple datasets and investigate sensitivity of model performance to tuning parameters: number of conformers and clusters.

2 Materials and Methods

2.1 Data Sets

Two groups of data sets were collected and prepared based on ChEMBL. The first ones consisted of only achiral compounds to make a more fair comparison with 2D QSAR models built on conventional descriptors which cannot encode stereoconfiguration. Five data sets of compounds with measured K_i or IC_{50} values against different targets were collected (Table 1). To split compounds into active and inactive classes we used thresholds recommended for different families of protein targets in the paper of Bose et al.^[46] Compounds were labeled active if their pK_i or pIC_{50} was ≥ 6 for enzyme targets and ≥ 7.5 for membrane proteins, and inactive otherwise. This resulted in well-balanced classification data sets.

The second group comprised data sets consisting of only chiral compounds with known configuration of all chiral centers and double bonds. These data sets were

chosen to investigate the ability of 3D models to predict activity of chiral compounds which is tricky to encode at 2D level of representation (Table 2).

We chose these two extreme cases, 2D-friendly and 2D-unfriendly, to better investigate applicability of different approaches. Of course, in the real applications there is often a mixture of chiral and achiral compounds in data sets. To evaluate performance of different approaches in more realistic conditions we built models for 162 additional data sets extracted from ChEMBL which comprised achiral compounds as well as chiral compounds with known and unknown configurations. They were processed identically to the previously described data sets.

Structures of compounds were curated with previously developed workflow which is publicly available at <https://bitbucket.imtm.cz/projects/STD/repos/std/browse>. To build 3D and ML models we generated up to 50 conformers using RDKit.^[47] RDKit was chosen because it reasonably well reproduces bioactive conformations of compounds in their bound state that is important for success of 3D modeling studies.^[48] Conformers with the root mean squared distance less than 0.5 Å were discarded. All data sets are available in Supplementary materials.

2.2 Descriptors

For 2D QSAR models we used three groups of descriptors: (i) binary Morgan fingerprints of radius 2 and length 2048, (ii) binary 2D pharmacophore fingerprints and (iii) physicochemical descriptors including EState indexes, the number of different pharmacophore features, rings systems, functional groups and fragments. All descriptors were calculated with RDKit.^[47] Definitions of pharmacophore features used for descriptor calculation were taken from our previously published study describing the development of

Table 1. Data sets consisting of only achiral compounds.

Target ChEMBL ID	Protein name	Protein family	Activity type	Total count	Actives count	Inactives count
CHEMBL253	Cannabinoid CB2 receptor	Membrane protein	K_i	1385	746	639
CHEMBL2409	Epoxide hydratase	Enzyme	IC_{50}	725	385	340
CHEMBL3155	Serotonin 7 (5-HT ₇) receptor	Membrane protein	K_i	641	335	306
CHEMBL3594	Carbonic anhydrase IX	Enzyme	K_i	1327	618	709
CHEMBL3717	Hepatocyte growth factor receptor	Enzyme	IC_{50}	584	249	335

Table 2. Data sets consisting of only chiral compounds.

Target ChEMBL ID	Protein name	Protein type	Activity type	Total count	Actives count	Inactives count
CHEMBL214	Serotonin 1a (5-HT _{1a}) receptor	Membrane protein	pK_i	355	229	126
CHEMBL217	Dopamine D2 receptor	Membrane protein	pK_i	892	312	580
CHEMBL232	Alpha-1b adrenergic receptor	Membrane protein	pK_i	158	67	91
CHEMBL233	Mu-opioid receptor	Membrane protein	pK_i	802	486	316
CHEMBL2971	Tyrosine-protein kinase JAK2	Enzyme	pIC_{50}	780	332	448
CHEMBL4235	11-beta-hydroxysteroid dehydrogenase 1	Enzyme	pIC_{50}	486	182	304

pmapper software.^[49] The full list of 2D descriptors of the third group is given in Supplementary materials (Table S1).

To encode conformers for 3D and MIL modeling we chose two groups of 3D descriptors. The first group consisted of 3D descriptors available in RDKit: asphericity, eccentricity, inertial shape factor, NPR1, NPR2, PMI1, PMI2, PMI3, radius of gyration, sphericity index, WHIM, PBF, Autocorr3D, RDF, MORSE and GETAWAY all together.^[47] These descriptors cannot discriminate stereoisomers and were used as a baseline for comparison purposes. The second type of descriptors was 3D pharmacophore descriptors. They were implemented based on the previously developed 3D pharmacophore signature generation code within *pmapper* software.^[49] Within this approach each conformer is represented by a complete graph where the corresponding pharmacophore features are vertices and edges are binned distances between features. Binning is required to enable fuzzy matching of pharmacophores. In this study we used binning step equal to 1 Å. We enumerated all possible quadruplets of pharmacophore features. Canonical signatures were generated for quadruplets using the algorithm described in the previous paper which took into account composition and configuration of features of the quadruplet.^[49] Thus, each conformer was encoded by a feature vector of counts of pharmacophore quadruplets having identical signatures. Since there are a lot of possible 3D pharmacophore quadruplets the obtained matrices were very sparse. To reduce their size we discarded quadruplets which occurred in less than 5% of compounds.

2.3 Modeling

To build single instance 3D models we used conformers with the lowest energy. To build multi-instance 3D models we used several approaches. The first one is simple averaging of descriptors among conformers of a particular compound (MIL-mean approach) which is used in many conventional 4D QSAR approaches. The second one is selection of maximum value of each descriptor among conformers of a particular compound (MIL-max approach).

The third approach creates new latent descriptors based on clustering of conformers (Figure 1). It is an adaptation of the approach published by Zhou and Zhang.^[50] All conformers of all compounds represented by corresponding 3D descriptors are clustered together using the k-means algorithm. The number of clusters is a tuning parameter and can be optimized in the course of cross-validation. For each molecule a new vector of latent variables is created. Its length is equal to the number of clusters. Feature values were equal to 1 if at least one conformer of the molecule fell into the corresponding cluster or 0 otherwise. Thus, a new descriptor matrix of the size (the number of molecules) $\times$ (the number of clusters) was created. Any conventional machine learning methods can be applied to build models based on this feature matrix. Hereinafter, this approach will be referred as MIL-kmeans.

All models within this study were built using the Random Forest (RF) algorithm.^[51] All models contained 250 trees. The optimal number of selected variables was determined by the search within the following range: 10%, 20%, 30%, log2 or squared root of the total number of descriptors. There are two additional tuning parameters of MIL models – the number of conformers and the number of clusters. The number of conformers was chosen to be from 5 to 50 with the step 5. To select the required number of

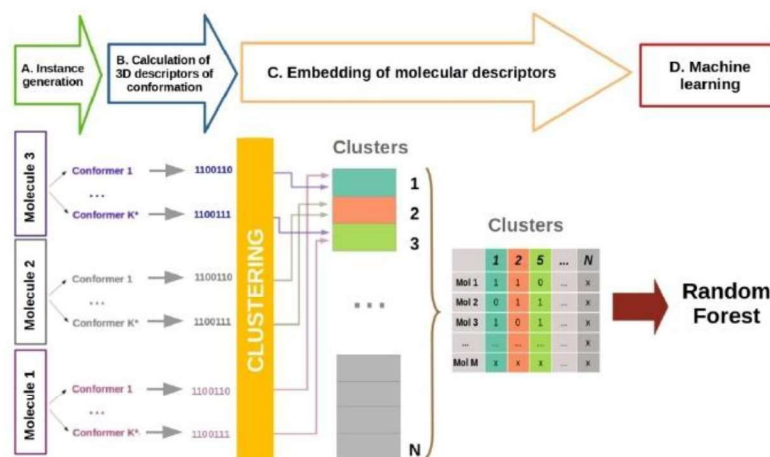


Figure 1. Generation of multiconformer descriptors based on clustering of compounds and their conformers.

conformers from up to 50 previously generated ones, all conformers were ordered by their energy calculated with MMFF94^[52] implemented in RDKit and the corresponding number of uniformly distributed conformers was selected. The number of clusters varied from 2 to 10 with step 1, from 10 to 100 with step 5, from 100 to 500 with step 50, from 500 to 1000 with step 100. To select optimal hyperparameters we applied grid search using five-fold cross-validation with random splits. The model with the highest accuracy estimated by cross-validation was chosen for comparative studies.

To estimate predictive ability of models we made five independent train/test splits for each data set. For achiral data sets splits were done randomly. For chiral data sets splits were done randomly but with restriction-all stereoisomers of a compound should be set to either a train or a test set but not to both simultaneously. This should give less biased estimation of predictive ability of models. Statistics was calculated for each test split and averaged among them. We calculated balanced accuracy as a measure of the predictive ability which is an average of sensitivity and specificity of a model.

It was reasonable to test consensus of multiple MIL-kmeans models because this could improve the predictive ability. Two consensus approaches were applied. The first one is consensus of top 10, 15, 25, 50 or 100 models with the highest cross-validation performance within all generated MIL-kmeans models (MIL-kmeans-consensus-top). Alternatively, we calculated consensus for all models within reasonable ranges of tuning parameters: the number of conformers from 20 to 50 and the number of clusters from 100 to 1000. Overall 182 models were combined within this consensus approach (MIL-kmeans-consensus-range). This approach does not require selection of best performing models that may simplify overall modeling workflow. Consensus prediction was made by majority voting.

3 Results and Discussion

3.1 Comparison of the Predictive Ability of 2D, 3D and MIL QSAR Models on Achiral and Chiral Data Sets

We selected models with the highest cross-validation performance and compared their accuracy on test sets. Models trained on 2D descriptors, in particular Morgan fingerprints, demonstrated high performance on achiral and chiral data sets (Tables 3 and 4). MIL models trained on 3D pharmacophore descriptors had similar performance to 2D models and in several cases outperformed them. Single instance models trained on 3D pharmacophores had lower performance than corresponding MIL models. MIL-mean and MIL-max models trained on 3D RDKit demonstrated performance similar to 3D pharmacophore MIL models, whereas MIL-kmeans models trained on 3D RDKit descriptors had substantially lower performance than corresponding models based on 3D pharmacophores. In general, there were no obvious advantage of any single combination of representation and modeling approach, regardless of data sets being achiral or chiral.

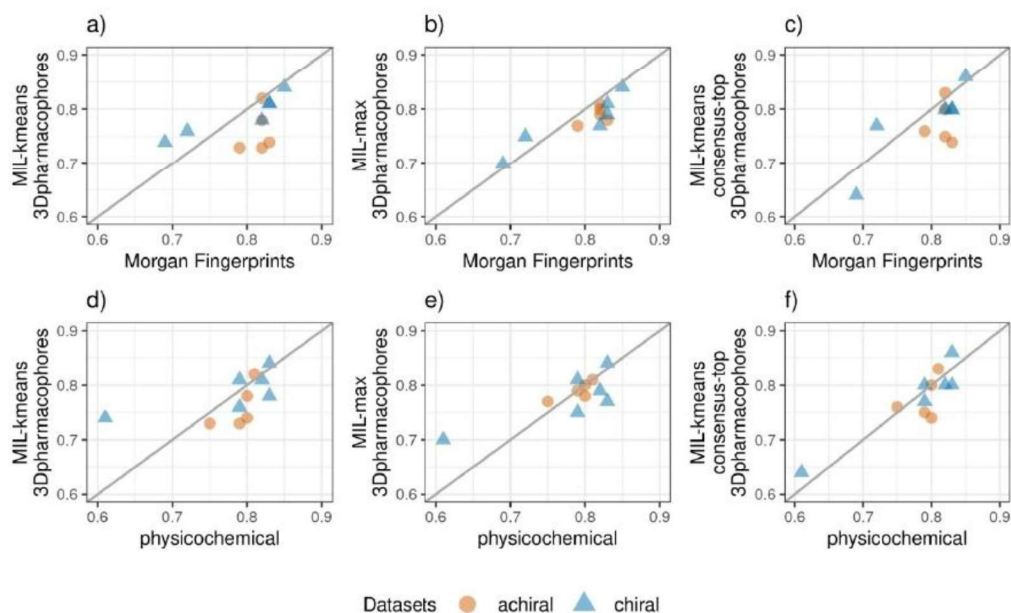
To better compare different approaches we plotted accuracies of the most accurate 2D classification models versus the best MIL models (Figure 2). In many cases 2D models already achieved high performance and, expectedly, MIL models could not substantially improve model performance in these cases. However, in some cases of chiral data sets, where 2D models demonstrated moderate accuracy, MIL models were able to improve prediction accuracy.

Table 3. Balanced accuracy averaged across five test sets for QSAR models on achiral data sets (standard deviation is in brackets).

Descriptor name	model algorithm	CHEMBL2409	CHEMBL253	CHEMBL3155	CHEMBL3594	CHEMBL3717
2D Morgan Fingerprints	single instance	0.82 (±0.045)	0.82 (±0.015)	0.83 (±0.063)	0.79 (±0.025)	0.82 (±0.043)
2D pharmacophore	single instance	0.79 (±0.043)	0.77 (±0.023)	0.78 (±0.050)	0.74 (±0.015)	0.81 (±0.035)
2D physicochemical	single instance	0.79 (±0.021)	0.80 (±0.016)	0.80 (±0.042)	0.75 (±0.021)	0.81 (±0.035)
3D pharmacophores	MIL-kmeans	0.73 (±0.017)	0.78 (±0.015)	0.74 (±0.045)	0.73 (±0.036)	0.82 (±0.041)
	MIL-kmeans-consensus-range	0.76 (±0.025)	0.80 (±0.016)	0.74 (±0.069)	0.76 (±0.026)	0.82 (±0.039)
	MIL-kmeans-consensus-top	0.75 (±0.027)	0.80 (±0.008)	0.74 (±0.063)	0.76 (±0.030)	0.83 (±0.039)
	MIL-max	0.79 (±0.028)	0.80 (±0.013)	0.78 (±0.067)	0.77 (±0.021)	0.81 (±0.028)
	MIL-mean	0.75 (±0.038)	0.78 (±0.012)	0.78 (±0.044)	0.74 (±0.026)	0.82 (±0.029)
3D RDKit	single instance	0.73 (±0.039)	0.76 (±0.013)	0.73 (±0.040)	0.76 (±0.018)	0.77 (±0.031)
	MIL-kmeans	0.66 (±0.031)	0.65 (±0.026)	0.63 (±0.023)	0.63 (±0.034)	0.64 (±0.051)
	MIL-kmeans-consensus-range	0.69 (±0.018)	0.69 (±0.037)	0.64 (±0.020)	0.65 (±0.031)	0.66 (±0.054)
	MIL-kmeans-consensus-top	0.68 (±0.025)	0.68 (±0.027)	0.65 (±0.038)	0.66 (±0.029)	0.66 (±0.049)
	MIL-max	0.76 (±0.026)	0.78 (±0.029)	0.75 (±0.041)	0.76 (±0.012)	0.78 (±0.054)
	MIL-mean	0.77 (±0.040)	0.80 (±0.032)	0.78 (±0.042)	0.74 (±0.028)	0.79 (±0.042)
	single instance	0.72 (±0.036)	0.75 (±0.028)	0.72 (±0.041)	0.75 (±0.018)	0.73 (±0.052)

Table 4. Balanced accuracy averaged across five test sets for QSAR models on chiral data sets (standard deviation is in brackets).

Descriptor name	model algorithm	CHEMBL214	CHEMBL217	CHEMBL232	CHEMBL233	CHEMBL2971	CHEMBL4235
2D Morgan Finger-prints	single instance	0.72 (±0.068)	0.83 (±0.022)	0.69 (±0.151)	0.82 (±0.017)	0.85 (±0.009)	0.83 (±0.036)
2D pharmacophore	single instance	0.74 (±0.066)	0.78 (±0.024)	0.62 (±0.137)	0.80 (±0.041)	0.83 (±0.029)	0.82 (±0.029)
2D physicochemical	single instance	0.79 (±0.068)	0.79 (±0.022)	0.61 (±0.122)	0.83 (±0.012)	0.83 (±0.032)	0.82 (±0.057)
3D pharmacophores	MIL-kmeans	0.76 (±0.049)	0.81 (±0.021)	0.74 (±0.152)	0.78 (±0.037)	0.84 (±0.024)	0.81 (±0.025)
	MIL-kmeans-consensus-range	0.75 (±0.062)	0.81 (±0.014)	0.66 (±0.161)	0.81 (±0.021)	0.85 (±0.025)	0.82 (±0.046)
	MIL-kmeans-consensus-top	0.77 (±0.042)	0.80 (±0.014)	0.64 (±0.179)	0.80 (±0.022)	0.86 (±0.025)	0.80 (±0.028)
	MIL-max	0.75 (±0.033)	0.81 (±0.013)	0.70 (±0.103)	0.77 (±0.046)	0.84 (±0.024)	0.79 (±0.034)
	MIL-mean	0.71 (±0.029)	0.81 (±0.026)	0.72 (±0.148)	0.77 (±0.025)	0.84 (±0.026)	0.81 (±0.050)
	single instance	0.68 (±0.049)	0.78 (±0.027)	0.67 (±0.150)	0.74 (±0.031)	0.82 (±0.022)	0.78 (±0.041)
	MIL-kmeans	0.65 (±0.081)	0.63 (±0.029)	0.62 (±0.078)	0.71 (±0.042)	0.69 (±0.033)	0.73 (±0.045)
	MIL-kmeans-consensus-range	0.66 (±0.089)	0.64 (±0.015)	0.61 (±0.095)	0.72 (±0.044)	0.72 (±0.019)	0.75 (±0.042)
	MIL-kmeans-consensus-top	0.69 (±0.102)	0.65 (±0.028)	0.67 (±0.111)	0.72 (±0.039)	0.73 (±0.028)	0.75 (±0.050)
	MIL-max	0.72 (±0.072)	0.76 (±0.023)	0.66 (±0.107)	0.81 (±0.026)	0.81 (±0.006)	0.80 (±0.054)
	MIL-mean	0.74 (±0.070)	0.78 (±0.021)	0.70 (±0.134)	0.82 (±0.032)	0.82 (±0.013)	0.79 (±0.047)
	single instance	0.70 (±0.056)	0.71 (±0.013)	0.61 (±0.133)	0.80 (±0.035)	0.81 (±0.025)	0.78 (±0.057)

**Figure 2.** Test set balanced accuracy of the best performing 2D and MIL models.

3.2 Influence of the Number of Conformers and the Number of Clusters on the Predictive Ability of MIL Models

First, we studied the influence of the number of clusters on model predictive performance. We chose MIL-kmeans models trained on 3D pharmacophore descriptors because they had higher accuracy than models trained on 3D RDKit descriptors. In the majority of cases we observed a similar trend between cross-validation and test set prediction performances. Therefore, we used only cross-validation performance values for chosen data sets to illustrate the influence of the number of clusters on model accuracy (Figure 3). The full plots for all data sets were provided in Supplementary materials (Figures S1–2).

We observed that in some cases model performance substantially decreased with increasing of the number of clusters (Figure 3). We explain this effect by the small total number of conformers with distinct 3D pharmacophore feature vectors. If the total number of conformers with distinct feature vectors increases, the model cross-validation performance also increases and reaches the plateau. A smaller number of distinct conformers may result in less populated clusters and lower generalizing ability of models. For future studies we suggest to choose the number of clusters at least 3–10 times less than the total number of conformers with distinct 3D pharmacophore feature vectors. Setting the number of clusters to >1000 did not improve prediction performance of models even for data sets having a large number of distinct conformers.

We chose the same data sets and models to demonstrate the influence of the number of conformers on the predictive ability of models (Figure 4). Plots for all models are provided in Supplementary materials (Figures S3–4). For each number of conformers we selected the model with the highest cross-validation performance among those

having different number of clusters. In general models were not very sensitive to the number of conformers (Figure 4). MIL-kmeans and MIL-max behaved most consistently on almost all data sets. Their cross-validation and test set performance was close and demonstrated slight improvement with increasing of the number of conformers. MIL-mean models were more sensitive to the number of conformers. We supposed that in this case descriptor values were changed more often with addition of new conformers, whereas for MIL-max and MIL-kmeans approaches those changes happened less frequently. ChEMBL232 data set resulted in models with the most variable performance for cross-validation as well as for test sets. This could be explained by the small number of compounds in the data set. Thus, we recommend using at least 20 conformers per compound to build MIL models and aware of the need to investigate model performance with respect to the number of conformers for small data sets.

3.3 Comparative Study of Predictive Performance on Additional Data Sets

To evaluate the predictive ability of 2D and MIL models we applied them to 162 additional data sets. We chose Morgan fingerprints to build 2D models and 3D pharmacophores to build MIL-max and MIL-kmeans. MIL-mean approach demonstrated comparable performance to MIL-max and we decided to not use it. For building of MIL-kmeans models we generated 50 conformers for all data sets and chose the number of clusters as a 1/5 of the total number of unique descriptor strings in a data set, but if this value was greater than 1000 we set the number of cluster to 1000. We created five test sets for each data set using the same procedure as described above and calculated average balanced accuracy

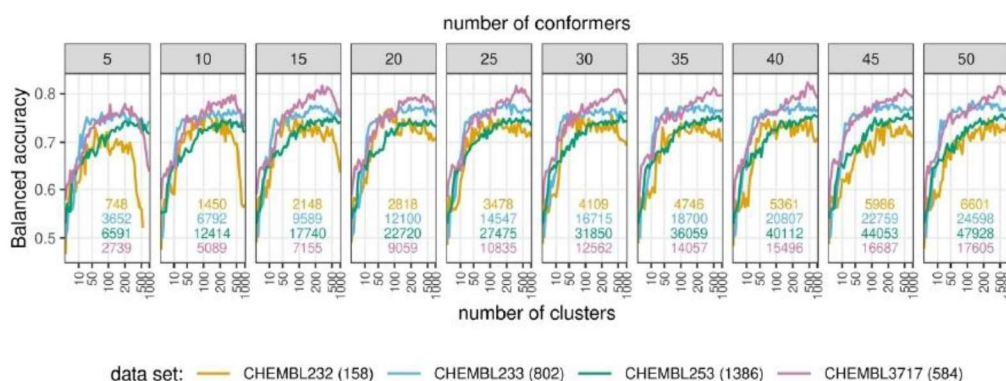


Figure 3. Cross-validation performance of MIL-kmeans models built on for achiral (ChEMBL253 and ChEMBL3717) and chiral (ChEMBL232, ChEMBL233) data sets using 3D pharmacophore descriptors. Numbers in brackets are the number of compounds in data sets. Colored numbers on the plots are the total number of conformers with distinct vectors of 3D pharmacophore descriptors within the corresponding data set.

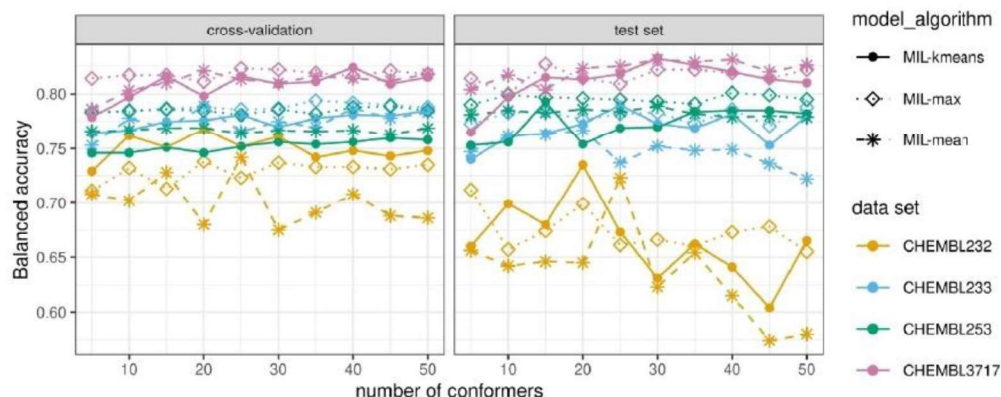


Figure 4. Cross-validation and test set performance for models built for achiral (CHEMBL253 and CHEMBL3717) and chiral (CHEMBL232 and CHEMBL233) data sets. For each number of conformers the model with the highest cross-validation performance was selected among models having different number of clusters.

for each data set. Statistical parameters for all data sets are provided in Supplementary materials (Table S2).

We removed from consideration 21 data sets which had balanced accuracy below 0.7 for all models. For the remaining 141 models we compared their predictive performance (Figure 5). Variance of MIL model performance relative to 2D models was lower in the case of MIL-max than for MIL-kmeans models. For the MIL-max approach the most notable improvement was observed only for the CHEMBL4361 data set where balanced accuracy was increased from 0.63 to 0.72, but in general MIL-max scheme gave only marginal improvement. Large discrepancy of model performance between 2D and MIL-kmeans may indicate importance of proper tuning of hyperparameters which are the number of clusters and conformers. However, even with the chosen default parameters MIL-kmeans

models demonstrated certain improvement over 2D models.

We analyzed factors which can determine successfulness of different types of models. The significant difference was observed for the size of data sets (p-value in t-test was below 0.05) (Figure 6). The data sets with more than 1000 molecules were better modeled using the conventional 2D approach than 3D MIL ones. There was no significant difference for the number of rotatable bonds but data sets with greater average number of rotatable bonds in molecules were better modeled by 2D descriptors (Supplementary Figure S5).

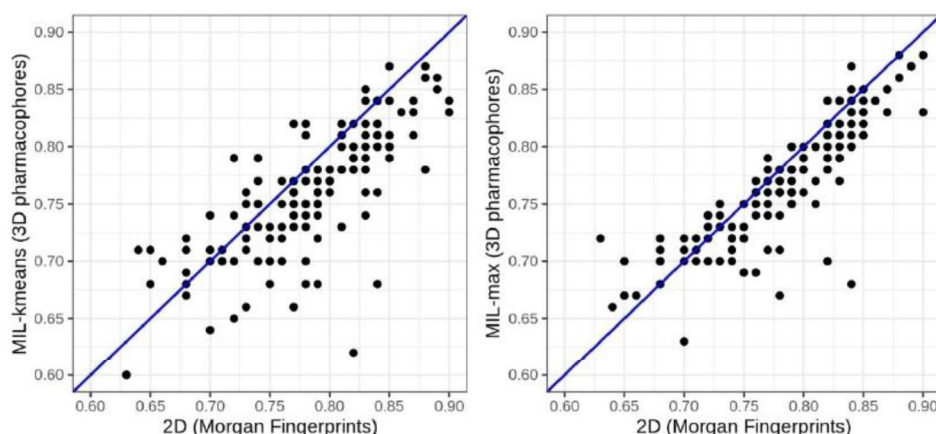


Figure 5. Average test set balanced accuracy for 2D and MIL models.

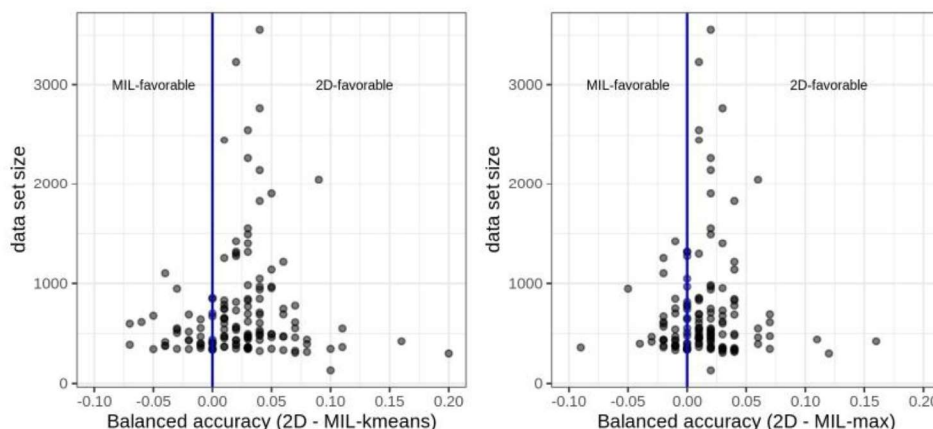


Figure 6. Differences between test set balanced accuracy of 2D (Morgan fingerprints) and MIL models versus the size of data sets.

3.4 Performance of 3D MIL Approaches to Solve Regression Tasks

We also studied performance of the suggested MIL approaches based on 3D pharmacophore representation to build regression models. However, we did not observe substantial improvement relatively to conventional 2D QSAR models based on Morgan fingerprints. Statistical parameters of models are provided in Supplementary materials (Table S3). This can be due to the chosen procedure to create a latent representation based on clustering. Clustering is more suitable to discriminate actives from inactives rather than highly active molecules from moderately active ones. Therefore, clustering may result in many clusters with compounds having different activity that will reduce the discriminative ability of MIL models. Thus, we do not recommend using the suggested MIL approach for regression tasks.

3.5 Comments on Interpretability of MIL Models

We believe that all QSAR models are interpretable and it is possible to retrieve useful knowledge from any model.^[53] The only issue is that not every interpretation approach can be applied to every model. Since the suggested modeling approach is alignment-independent, one can apply corresponding interpretation approaches to MIL models, e.g. one can retrieve a contribution of particular substructures by removing or masking fragments of interest from the studied molecules.^[54–55] We did not study interpretability of MIL models, but there are no technical or theoretical restrictions to interpret them as any other alignment-independent model. Specific approaches developed for interpretation of alignment-dependent models, like contribution of steric and electronic factors in CoMFA¹⁶, cannot

be applied in this particular case. However, this does not reduce the value of suggested MIL approaches.

4 Conclusions

In this study we demonstrated that conventional QSAR models based on 3D descriptors trained on single-conformer representation cannot outperform 2D models. Representations based on multiple conformations using MIL approach can result in models with similar or even better performance than 2D models and almost always better than single-conformer based 3D QSAR models. This was especially notable in the case of data sets consisting of stereoisomers with different activity which cannot be captured by models trained on chirality-unaware 2D/3D descriptors. Using 3D pharmacophore descriptors which are chirality-aware resulted in substantial improvement of predictive ability of models in several cases.

As we demonstrated, conventional 4D QSAR modeling approaches which are commonly used for modeling of compounds represented by several conformers can be considered as a particular case of multiple instance learning. The averaging scheme used by conventional 4D QSAR approaches may be not optimal and the predictive ability of models can be improved by using other embedding schemes. Here, we used maximization and clustering MIL approaches. They performed almost as good as or better than traditional averaging of descriptors of individual conformers and can be recommended as a viable alternative. In spite of complexity of the MIL-kmeans approach due to an additional tuning parameter (the number of clusters) it may outperform MIL-max in some cases. The optimum number of conformers in MIL-max and MIL-kmeans models depends on flexibility of training set compounds. We suggest to generate at least 30 conformers and to choose the number of clusters 3–10 times greater than the total number of

conformers with distinct 3D pharmacophore feature vectors.

We verified applicability of implemented MIL approaches in combination with 3D pharmacophore descriptors on the number of additional data sets and demonstrated that MIL classification models can be competitive to conventional 2D models if data sets are not large (less 1000 compounds). For larger data sets conventional 2D models were consistently better. Unfortunately, the developed MIL approaches demonstrated poor performance on regression tasks and cannot be recommended for this kind of modeling. Overall, 3D pharmacophore descriptors in combination with MIL approaches can be considered as a reasonable choice for classification data sets where 2D models fail or result in low predictive accuracy.

Statement of Contribution

A.N.: implementation of the approach, building models and their interpretation, writing the draft manuscript. D.Z.: collection and curation of data sets, writing the draft manuscript. I.B.: development of the approach. T.M.: development of the approach, design the study, manuscript editing. P.P.: development of the approach, design the study, writing and editing of the manuscript.

Acknowledgements

This research was funded by the Ministry of Education, Youth and Sports of the Czech Republic within the INTER-EXCELLENCE LTARF18013 project (agreement number MSMT-5727/2018-2) and by the Ministry of Education and Science of the Russian Federation, agreement No14.587.21.0049 (unique identifier RFMEFI58718X0049).

Conflict of Interest

None declared.

Data Availability Statement

The data that supports the findings of this study are available in the supplementary material of this article

References

- [1] E. N. Muratov, J. Bajorath, R. P. Sheridan, I. V. Tetko, D. Filimonov, V. Porokov, T. I. Oprea, I. I. Baskin, A. Varnek, A. Roitberg, O. Isayev, S. Curtalolo, D. Fourches, Y. Cohen, A. Aspuru-Guzik, D. A. Winkler, D. Agrafiotis, A. Cherkasov, A. Tropsha, *Chem. Soc. Rev.* **2020**.
- [2] W. Jorgensen, *Science* **1991**, 254, 954–955.
- [3] N. Schneider, R. A. Lewis, N. Fechner, P. Ertl, *ChemMedChem* **2018**, 13, 1315–1324.
- [4] L. A. Nguyen, H. He, C. Pham-Huy, *Int. J. Biomed. Sci.* **2006**, 2, 85–100.
- [5] Y. C. Martin, *J. Comput.-Aided Mol. Des.* **2009**, 23, 693.
- [6] F. Milletti, A. Vulpatti, *J. Chem. Inf. Model.* **2010**, 50, 1062–1074.
- [7] D. Fourches, E. Muratov, A. Tropsha, *J. Chem. Inf. Model.* **2010**, 50, 1189–1204.
- [8] F. Bonachera, B. Parent, F. Barbosa, N. Froloff, D. Horvath, *J. Chem. Inf. Model.* **2006**, 46, 2457–2477.
- [9] T. R. Gimadiev, T. I. Madzhidov, R. I. Nugmanov, I. I. Baskin, I. S. Antipin, A. Varnek, *J. Comput.-Aided Mol. Des.* **2018**, 32, 401–414.
- [10] P. Polishchuk, E. Mokshyna, A. Kosinskaya, A. Muats, M. Kulinsky, O. Tinkov, L. Ognichenko, T. Khristova, A. Artemenko, V. Kuz'min, in *Advances in QSAR Modeling: Applications in Pharmaceutical, Chemical, Food, Agricultural and Environmental Sciences*, Vol., Ed: Roy K, Springer International Publishing, Cham **2017**, pp. 107–147.
- [11] H. P. Schultz, E. B. Schultz, T. P. Schultz, *J. Chem. Inf. Comput. Sci.* **1995**, 35, 864–870.
- [12] A. Golbraikh, D. Bonchev, A. Tropsha, *J. Chem. Inf. Comput. Sci.* **2001**, 41, 147–158.
- [13] J. Aires-de-Sousa, J. Gasteiger, *J. Mol. Graphics Modell.* **2002**, 20, 373–388.
- [14] A. Golbraikh, A. Tropsha, *J. Chem. Inf. Comput. Sci.* **2003**, 43, 144–154.
- [15] P. Carbonell, L. Carlsson, J.-L. Faulon, *J. Chem. Inf. Model.* **2013**, 53, 887–897.
- [16] R. D. Cramer, D. E. Patterson, J. D. Bunce, *J. Am. Chem. Soc.* **1988**, 110, 5959–5967.
- [17] A. J. Hopfinger, *J. Am. Chem. Soc.* **1980**, 102, 7196–7206.
- [18] M. Pastor, G. Cruciani, K. A. Watson, *J. Med. Chem.* **1997**, 40, 4089–4102.
- [19] J. R. Woolfrey, M. A. Avery, A. M. Doweyko, *J. Comput.-Aided Mol. Des.* **1998**, 12, 165–181.
- [20] G. Klebe, U. Abraham, T. Mietzner, *J. Med. Chem.* **1994**, 37, 4130–4146.
- [21] J. Polanski, B. Walczak, *Computers & Chemistry* **2000**, 24, 615–625; *Chemistry* **2000**, 24, 615–625.
- [22] G. V. Sitnikov, N. I. Zhokhova, Y. A. Ustynuk, A. Varnek, I. I. Baskin, *J. Comput.-Aided Mol. Des.* **2015**, 29, 233–247.
- [23] B. D. Silverman, D. E. Platt, *J. Med. Chem.* **1996**, 39, 2129–2140.
- [24] R. Todeschini, P. Gramatica, *SAR QSAR Environ. Res.* **1997**, 7, 89–115.
- [25] V. Consonni, R. Todeschini, M. Pavan, *J. Chem. Inf. Comput. Sci.* **2002**, 42, 682–692.
- [26] J. H. Schuur, P. Selzer, J. Gasteiger, *J. Chem. Inf. Comput. Sci.* **1996**, 36, 334–344.
- [27] M. C. Hemmer, V. Steinhauer, J. Gasteiger, *Vib. Spectrosc.* **1999**, 19, 151–164.
- [28] A. J. Hopfinger, S. Wang, J. S. Tokarski, B. Jin, M. Albuquerque, P. J. Madhav, C. Duraiswami, *J. Am. Chem. Soc.* **1997**, 119, 10509–10524.
- [29] A. Vedani, M. Dobler, *Prog. Drug Res.*, Vol., Ed: Jucker E, Birkhäuser Basel, Basel **2000**, pp. 105–135.
- [30] J. Polanski, A. Bak, *J. Chem. Inf. Comput. Sci.* **2003**, 43, 2081–2092.
- [31] A. Bak, J. Polanski, *J. Chem. Inf. Model.* **2007**, 47, 1469–1480.

- [32] V. A. Potemkin, R. M. Arslambekov, E. V. Bartashevich, M. A. Grishina, A. V. Belik, S. Perspicace, S. Guccione, *J. Struct. Chem.* **2002**, *43*, 1045–1049.
- [33] V. E. Kuz'min, A. G. Artemenko, P. G. Polischuk, E. N. Muratov, A. I. Khromov, A. V. Liahovskiy, S. A. Andronati, S. Y. Makan, *J. Mol. Model.* **2005**, *11*, 457–467.
- [34] T. G. Dietterich, R. H. Lathrop, T. Lozano-Pérez, *Artificial Intelligence* **1997**, *89*, 31–71.
- [35] C. Bergeron, J. Zaretzki, C. Breneman, K. P. Bennett. (2008). Multiple instance ranking, Proceedings of the 25th international conference on Machine learning (pp. 48–55). Helsinki, Finland: Association for Computing Machinery.
- [36] R. Teramoto, H. Kashima, *J. Mol. Graphics Modell.* **2010**, *29*, 492–497.
- [37] G. Fu, X. Nan, H. Liu, R. Y. Patel, P. R. Daga, Y. Chen, D. E. Wilkins, R. J. Doerksen, *BMC Bioinf.* **2012**, *13*, 53.
- [38] F. U. A. A. Minhas, A. Ben-Hur, *Bioinformatics* **2012**, *28*, i416–i422.
- [39] Z. Zhao, G. Fu, S. Liu, K. M. Elokely, R. J. Doerksen, Y. Chen, D. E. Wilkins, *BMC Bioinf.* **2013**, *14*, S16.
- [40] M.-A. Carbonneau, V. Cheplygina, E. Granger, G. Gagnon, *Pattern Recognition* **2018**, *77*, 329–353.
- [41] J. Amores, *Artificial Intelligence* **2013**, *201*, 81–105.
- [42] J. Wang, J.-D. Zucker. (2000). Solving the Multiple-Instance Problem: A Lazy Learning Approach, Proceedings of the Seventeenth International Conference on Machine Learning (pp. 1119–1126): Morgan Kaufmann Publishers Inc.
- [43] M. Zhang. (2010, 27–29 Oct. 2010). A k-Nearest Neighbor Based Multi-Instance Multi-Label Learning Algorithm. Paper presented at the 2010 22nd IEEE International Conference on Tools with Artificial Intelligence.
- [44] G. Doran, S. Ray, *Machine Learning* **2014**, *97*, 79–102.
- [45] C. Yixin, B. Jinbo, J. Z. Wang, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2006**, *28*, 1931–1947.
- [46] N. Bosc, F. Atkinson, E. Felix, A. Gaulton, A. Hersey, A. R. Leach, *J. Cheminf.* **2019**, *11*, 4.
- [47] RDKit: Open-Source Cheminformatics Software 2017.09 (2017). <http://rdkit.org/>.
- [48] S. Riniker, G. A. Landrum, *J. Chem. Inf. Model.* **2015**, *55*, 2562–2574.
- [49] A. Kutlushina, A. Khakimova, T. Madzhidov, P. Polishchuk, *Molecules* **2018**, *23*, 3094.
- [50] Z.-H. Zhou, M.-L. Zhang, *Knowledge and Information Systems* **2007**, *11*, 155–170.
- [51] L. Breiman, *Machine Learning* **2001**, *45*, 5–32.
- [52] T. A. Halgren, *J. Comput. Chem.* **1996**, *17*, 490–519.
- [53] P. Polishchuk, *J. Chem. Inf. Model.* **2017**, *57*, 2618–2639.
- [54] P. G. Polishchuk, V. E. Kuz'min, A. G. Artemenko, E. N. Muratov, *Mol. Inf.* **2013**, *32*, 843–853.
- [55] M. Matveieva, P. Polishchuk, *J. Cheminf.* **2021**, *13*, 41.

Received: July 14, 2021

Accepted: July 19, 2021

Published online on August 3, 2021

Conclusion

In this study, the clustering-based classification algorithm MIL-kmeans was compared with simpler alternative MIL-mean and MIL-max algorithms. The MIL-kmeans and MIL-max algorithms perform similarly to or better than the traditional MIL-mean algorithm. MIL-kmeans is a more sophisticated method that requires the optimization of additional hyperparameters but can outperform the simpler MIL-max algorithm in some cases. Based on the comparison results, MIL-kmeans was chosen as the main algorithm for analyzing 3D models trained with multiple conformers.

3D models based on a single conformer were expectedly worse than 2D models. The inclusion of multiple conformers in combination with the MIL-kmeans algorithm significantly increased the accuracy of 3D models in almost all cases. A comparison of 3D multi-conformer models and traditional 2D models was performed on three collections of datasets: 5 achiral, 6 chiral, and 162 mixed datasets. For 4 of the 5 achiral datasets, 2D models outperformed 3D multi-conformer models based on 3D pharmacophore descriptors. In the case of chiral datasets, 3D multi-conformer models significantly improved prediction accuracy only for the ChEMBL232 dataset, whereas in the other datasets, 2D models based on Morgan fingerprints or physicochemical descriptors from RDKit were the best. For an additional collection of 162 datasets containing both achiral and chiral molecules, 2D models outperformed 3D multi-conformer models in most cases. Nevertheless, 2D and 3D models are comparable when the dataset size is less than 1000 compounds. In larger datasets (>1000), 2D models are consistently better.

In general, the developed MIL-kmeans algorithm in combination with 3D pharmacophore descriptors can be considered as an alternative approach for modeling the bioactivity of compounds in cases where traditional 2D models fail to accurately classify bioactive compounds.

2.3 Modeling of compounds bioactivity with conformation ensembles

A common technique in ligand-based modeling approaches is based on correlating the ligand structure with their experimental bioactivity using machine learning methods. The structure of ligands can be encoded with 2D or 3D chemical descriptors. 2D descriptors are the more popular because they are quick and easy to calculate as well as often predictive models based on 2D descriptors demonstrate good performance. But, in special cases where the bioactivity of the molecule is strongly related to the 3D structure, 3D descriptors are preferable.

However, the wide application of 3D descriptors is limited by a long-standing problem related to the selection of probable bioactive conformers of the molecule. Molecules can be represented by multiple alternative conformers, but only a single bioactive conformer, which binds to the target, is responsible for the observed bioactivity. Bioactive conformers can be determined experimentally (e.g. with X-ray or NMR methods), but the amount of experimental data is still limited. Therefore, often the lowest-energy conformer, generated using methods of geometry optimization, is selected for modeling. However, the independently optimized lowest-energy conformer can significantly differ from the actual bioactive conformer, which makes it difficult to establish a correct relationship between the structure and bioactivity of the compound.

To overcome this problem, a new 3D modeling approach based on multi-instance machine learning (MIL), which does not require the selection of conformers, was developed within this research project. In this approach, all available conformers of the molecule are processed simultaneously by special MIL algorithms, some of which can also automatically identify bioactive conformers. In this study, 3D multi-conformer models were compared with 3D single-conformer models as well as with traditional 2D models based on popular 2D descriptors. A large-scale comparison analysis was performed on 175 datasets on the bioactivity of compounds extracted from the ChEMBL-23 database. In addition, 4 datasets including experimental 3D ligand structures from Protein Data Bank (PDB) database were used to test MIL algorithms in the task of identification of bioactive conformers.

QSAR Modeling Based on Conformation Ensembles Using a Multi-Instance Learning Approach

Dmitry V. Zankov, Mariia Matveieva, Aleksandra V. Nikonenko, Ramil I. Nugmanov, Igor I. Baskin, Alexandre Varnek,* Pavel Polishchuk,* and Timur I. Madzhidov*



Cite This: *J. Chem. Inf. Model.* 2021, 61, 4913–4923



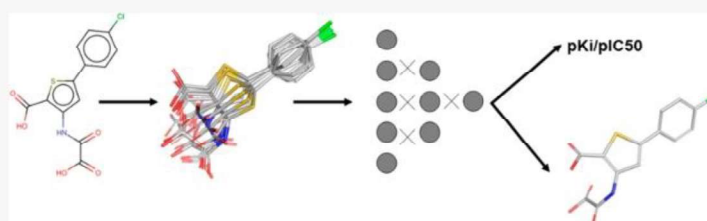
Read Online

ACCESS |

Metrics & More

Article Recommendations

Supporting Information



ABSTRACT: Modern QSAR approaches have wide practical applications in drug discovery for designing potentially bioactive molecules. If such models are based on the use of 2D descriptors, important information contained in the spatial structures of molecules is lost. The major problem in constructing models using 3D descriptors is the choice of a putative bioactive conformation, which affects the predictive performance. The multi-instance (MI) learning approach considering multiple conformations in model training could be a reasonable solution to the above problem. In this study, we implemented several multi-instance algorithms, both conventional and based on deep learning, and investigated their performance. We compared the performance of MI-QSAR models with those based on the classical single-instance QSAR (SI-QSAR) approach in which each molecule is encoded by either 2D descriptors computed for the corresponding molecular graph or 3D descriptors issued for a single lowest energy conformation. The calculations were carried out on 175 data sets extracted from the ChEMBL23 database. It is demonstrated that (i) MI-QSAR outperforms SI-QSAR in numerous cases and (ii) MI algorithms can automatically identify plausible bioactive conformations.

INTRODUCTION

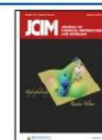
A typical QSAR model establishes a relationship between bioactivity and molecular structure represented by a vector of molecular descriptors. Meanwhile, one can consider descriptors of different dimensionality: 0D (derived from the empirical formula), 1D (derived from a vector of values, e.g., fingerprints), 2D (derived from a molecular graph), 3D (derived from a single conformation), and 4D (usually derived from a molecular-dynamic trajectory). Although 2D descriptors are a gold standard in QSAR modeling because of the simplicity of their calculation, 2D representation does not directly encode the spatial structure of molecules which is important for protein–ligand recognition. Ignoring this information may reduce the performance of QSAR models. This motivates the development of 3D-QSAR methods which consider explicitly the spatial structure of the molecules.¹

The first proposed 3D-QSAR method was Comparative Molecular Field Analysis (CoMFA),² which correlates the biological activity of organic molecules with their electrostatic and “steric” fields represented as interaction energies with special probes placed at grid nodes around an aligned set of molecules. To build a CoMFA model, a single conformation (3D structure) should be chosen for each molecule, followed

by their alignment in space and calculation of interaction energies considered as descriptors. Choosing irrelevant conformations and/or alignment may result in a substantial decrease in model performance. This issue becomes critical for flexible molecules possessing several rotatable bonds and, as a consequence, many possible conformations. Following CoMFA, most 3D-QSAR methods rely on the choice of a single “bioactive” conformation for a molecule, which can be determined from the structures of protein–ligand complexes. Although such conformations can be determined in X-ray, NMR, and EM (electron microscopy) studies or computed using molecular modeling techniques (docking, molecular dynamics, etc.), there is a clear indication that using “receptor-bound” conformations might be a bad choice for building QSAR models.³ Although multiple 3D-QSAR approaches have

Received: June 17, 2021

Published: September 23, 2021



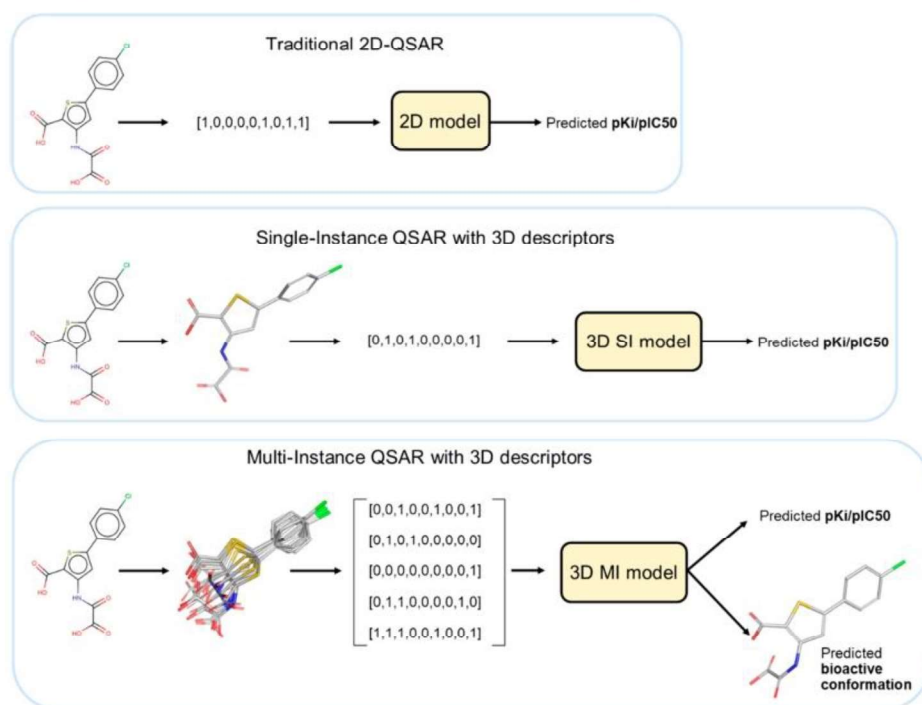


Figure 1. Approaches to constructing models for predicting the bioactivity of compounds. Multi-instance 3D MI-QSAR models can predict bioactivity and identify the relevant conformation.

been developed so far, all of them suffer from the issue of selection of relevant conformations.^{4–12}

The concept of 4D-QSAR,^{13,14} in which a molecule is represented by an ensemble of conformations, was introduced to overcome the limitations of the 3D-QSAR approach associated with the choice of a single conformation for each molecule. Such ensembles may be extracted from molecular-dynamics trajectories, sampled from Monte Carlo simulations, or obtained using conformer generators. Most 4D-QSAR approaches compute 3D alignment-independent descriptors for individual conformations and combine them by some schemes to obtain a single vector of descriptors for each molecule to be used in conventional machine learning methods. The most widely used schemes are summation or averaging of 3D descriptors of individual conformations¹⁵ and summation of 3D descriptors weighted by the Boltzmann factor estimated for conformations in vacuum or water solution.^{16–18} Considering that the (a) energy assessment of the conformations is subjected to high errors in the parametrization of force fields and (b) energy of the receptor-bound conformations of ligands may be rather high and hence their Boltzmann factor may be very low, the Boltzmann averaging schemes may introduce significant noise to the data.

Multi-instance (MI) machine learning approaches can be used to solve the issues of representation of each molecule by multiple conformations (instances) and automatic selection of the most relevant ones (Figure 1). In the multi-instance approach, an example (i.e., a molecule) is presented by a bag of instances (i.e., a set of conformations), and a label (a bioactivity value) is available only for a bag (a molecule) but not for individual instances (conformations). MI learning was

first introduced for recognizing handwritten numbers¹⁹ but became better known after the paper by Dietterich et al.,²⁰ where the authors developed a model to predict the odors of compounds, which were represented by multiple conformations. The Compass algorithm²¹ is another example where MI learning significantly improved the performance of models in comparison with single-instance (SI) learning on the task of predicting the bioactivity of the compounds. The Compass algorithm implemented the idea of representing a molecule by multiple conformations, which were used to train a neural network. Though MI learning was initially developed for modeling the properties/activities of chemical compounds, this methodology has not found wide application in the QSAR area, although it has become widely adopted in other fields.²² Only a few studies with the application of MI learning to predict the bioactivity of the compounds have been published so far in mathematics and bioinformatics journals.^{20,23–25} Moreover, recently proposed deep learning-based multi-instance approaches have not been used in the chemistry domain except in our recent work.²⁶ Recently, we demonstrated the applicability of unsupervised²⁷ and supervised clustering-based MI approaches to bioactivity predictions on several data sets.²⁸ However, a proper comparison of MI learning approaches to conventional ones has not been made so far.

The main goal of MI learning algorithms is to predict a label for an object represented by a bag of instances. However, it is often desirable not only to predict the bag label (in our case, to assess the bioactivity of a given molecule) but also to identify the key instances in the bag (i.e., to assess bioactive conformations). This problem, called Key Instance Detection (KID), was first formalized in a prior publication.²⁹ The

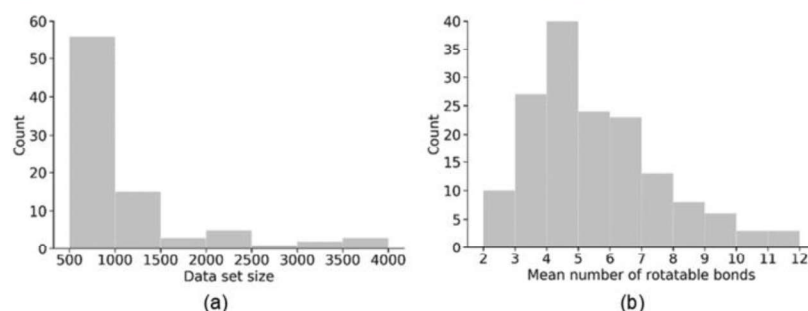


Figure 2. Characteristics of data sets. Number of data sets of a particular size (a) and with the particular mean number of rotatable bonds (b).

identification of the conformation responsible for the observed bioactivity of the molecule provides deeper insight into the interaction mechanism between the ligand and the target protein.

In this study, we show that the application of MI learning can be used to solve the long-standing problem of 3D-QSAR—the selection of relevant (or biologically active) conformations for modeling. Instead of a single conformation, MI learning considers the whole conformational ensemble, which significantly improves the predictive performance of the models based on 3D descriptors. Here, several 3D MI QSAR approaches were implemented and their performances compared on numerous data sets. It has been demonstrated that in most cases, the 3D MI QSAR models outperformed conventional 2D models. We also identified the physicochemical characteristics of compounds impacting the performance of 3D MI or 2D models. In addition, we studied the ability of MI models based on attention neural networks to identify relevant bioactive conformations.

METHODS

Data Sets. One hundred seventy-five data sets of compounds with measured pK_i or pIC_{50} values were extracted from the ChEMBL23 database. The size of the data sets varied from several hundred to several thousand compounds (Figure 2a). Molecules with a molecular weight greater than 700 (3% of the total number of molecules) were discarded. Because the performance of the 3D models may depend on the flexibility of the studied compounds, the average number of rotatable bonds for molecules in each data set was calculated using RDKit (Figure 2b). Most molecules in the data sets can be considered as low to moderately flexible with the average number of rotatable bonds within 3–6.

In addition, in the collected data sets, we identified compounds deposited in the Protein Data Bank (PDB) and retrieved their conformations. These PDB conformations were used as references to compare with the conformations predicted by MI models to provide the largest contribution to biological activity.

Conformation Generation. Conformations representing each molecule were generated using the algorithm implemented in RDKit,³⁰ which is claimed by its authors to be able to reproduce bioactive conformations observed for ligands in PDB complexes with reasonable accuracy. This ability is important because it may improve the performance of the obtained models, may make them more reasonable, and in the case of MI modeling approaches would increase the probability of identifying the most relevant/contributed conformations to

the studied end point. It also increases the chance to find conformations similar to those observed in the X-ray structures of protein–ligand complexes if the latter are available. In our study, we generated up to 100 conformations and removed conformations with RMSD values below 0.5 Å to the remaining ones to reduce redundancy.

Descriptors. For the descriptor representation of the conformations, we used previously developed 3D pharmacophore signatures.³¹ Each conformation is represented by a set of pharmacophore features (H-bond donor/acceptor, center of positive/negative charge, hydrophobic, and aromatic) determined by applying the corresponding SMARTS patterns. All possible quadruplets of features of a particular conformation were enumerated. Distances between features were binned to allow fuzzy matching of quadruplets with small differences in the position of features. Here, we used the 1 Å bin step as it demonstrated reasonable performance in our previous studies.^{26,31–33} Three-dimensional pharmacophore signatures were generated for each quadruplet according to the algorithm described in our previous publication.³¹ These signatures consider distances between features and their spatial arrangement to recognize the stereoconfiguration of the quadruplets. We counted the number of identical 3D pharmacophore quadruplet signatures for each conformation and used the obtained vectors as descriptors for model building. The three-dimensional pharmacophore descriptors used in this study were implemented in the pmapper Python package (<https://github.com/DrrDom/pmapper>). Since the pharmacophore descriptors were very sparse, we kept only those quadruplets that occurred in at least 5% of all conformations of the data set molecules.

To build 2D models, we chose binary Morgan fingerprints (MorganFP) of radius 2 and size 2048 calculated with RDKit because they are widely used 2D descriptors and demonstrated high performance in previous benchmarking studies.³⁴ For comparative purposes, we also used 2D physicochemical descriptors (PhysChem) and binary 2D pharmacophore fingerprints (PharmFP) calculated with RDKit. The former included EState indexes, the number of different pharmacophore features, rings systems, functional groups, and fragments (the full list is provided in the Supporting Information). To calculate the 2D pharmacophore descriptors, we used the same definitions of the pharmacophore features as in pmapper to make the comparison more robust. Afterward, pharmacophore triplets were enumerated using default binning of the topological distances (0–2, 2–5, 5–8, 8+).

Algorithms. In conventional SI-QSAR, each molecule is represented by a single vector of 2D descriptors computed for

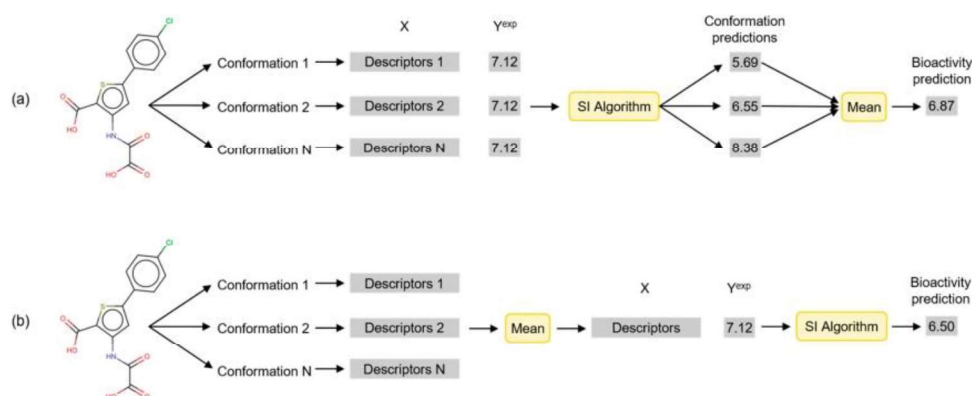


Figure 3. MI wrapper algorithms: (a) Instance-Wrapper and (b) Bag-Wrapper. Learning algorithm (SI Algorithm in the figure) was a three-layer fully connected neural network having 256, 128, and 64 neurons in hidden layers.

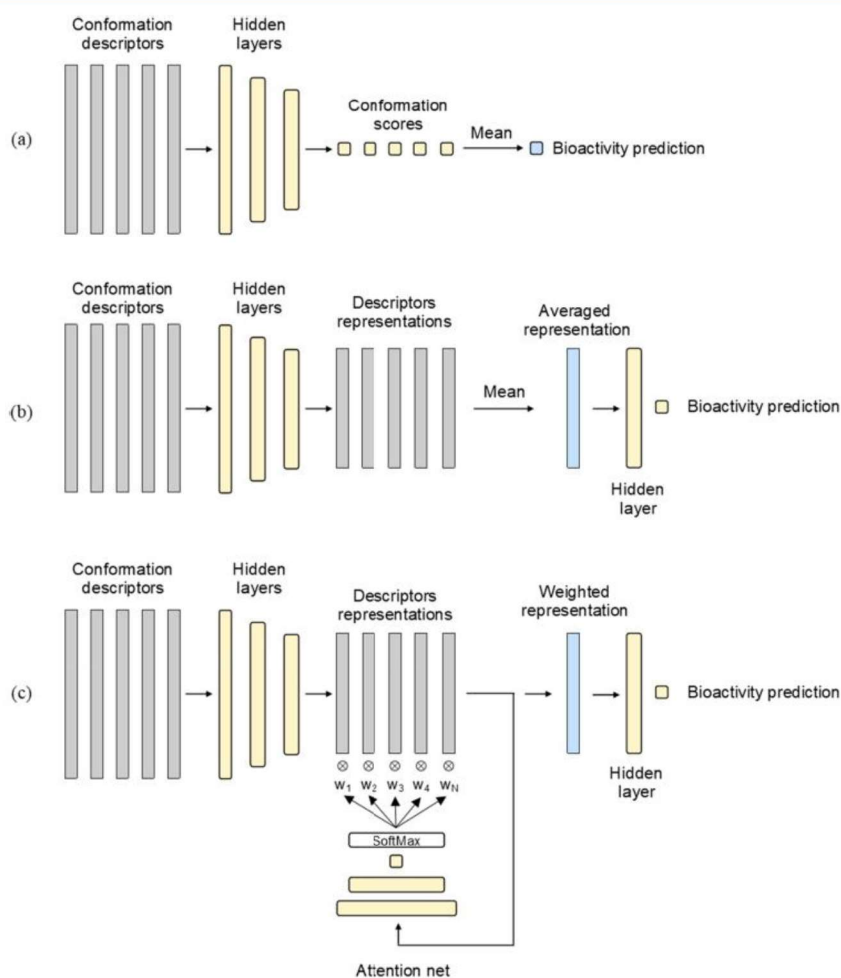


Figure 4. Multi-instance neural networks: (a) Instance-Net, (b) Bag-Net, and (c) Bag-AttentionNet.

the corresponding molecular graph or 3D descriptors for its lowest energy conformation. In MI-QSAR, a molecule is represented by a set of conformations and a set of associated

vectors of descriptors which forms a bag of instances. To build a model in this case special algorithms should be applied. All of the considered MI algorithms can be divided into two groups:

instance based and bag based.³⁵ Instance-based algorithms consider each conformation as a separate training instance. Bag-based algorithms, on the contrary, represent a molecule by a single vector of descriptors, which is produced from the vectors of the conformation descriptors.

Single-Instance Algorithms. We considered traditional SI learning as a baseline approach, where a molecule is described by a vector of 2D descriptors or a vector of 3D descriptors associated with the lowest energy conformation. We used a three-layered fully connected neural network with ReLU activation to construct SI-QSAR models. Our tests show that such architecture gives quite high and stable results across different data sets and does not require additional hyperparameter adjustment for a particular data set.

Multi-Instance Wrappers. The learning process in instance-based algorithms occurs at the instance level. Instance-level learning is applicable if it is possible to assign a label to individual instances in a bag. Also, it is assumed that there is a rule that aggregates the predictions for each instance to get the prediction for the entire bag. The simplest instance-based MI algorithm is Instance-Wrapper, where each training instance of a bag is assigned the same label as for the whole bag. This means, for example, that if a molecule is bioactive, it is assumed that all of its conformations are bioactive. As a result, one gets a data set where each conformation is an individual training object and any conventional ML algorithms can be applied to build the model. Given a new molecule, the bioactivity is predicted for each conformation and predictions are averaged to get the final predicted bioactivity of the molecule (Figure 3a). This approach has an obvious drawback because assigning the same bioactivity to all conformations of a molecule in a training set can bring some noise into the learning process because the fact that a molecule is bioactive does not mean that all of its conformations are biologically relevant and responsible for protein–ligand recognition.

The learning process of bag-based algorithms occurs at the bag level. In bag-based algorithms, there is no need to identify a label for each instance in a bag. Instead, there is an operation that aggregates the instances to get a single vector representing the entire bag. Our implementation of the Bag-Wrapper algorithm averaged descriptor values across all conformations and supplied this single vector of descriptors to a conventional SI machine learning method—a three-layer fully connected neural network (Figure 3b). The Bag-Wrapper algorithm has a drawback similar to Instance-Wrapper because aggregation of the descriptor vectors of all conformations to the resultant vector may introduce additional noise due to the contribution of irrelevant conformations.

Multi-Instance Neural Networks. Multi-instance neural networks learn in an end-to-end way and take a bag of instances as input and directly output bag prediction. All parameters in MI networks are optimized via backpropagation. Wang et al.³⁶ revisited MI neural networks and proposed a series of novel neural network frameworks for MI learning. They considered two types of MI neural networks: mi-Net (hereafter Instance-Net) and MI-Net (hereafter Bag-Net). We implemented both of these neural network architectures. In Instance-Net (Figure 4a), instances are running through fully connected layers and an output neuron. Then, instance predictions are averaged in the pooling layer to obtain a bag prediction, and its error is calculated and backpropagated to adjust model weights. Bag-Net (Figure 4b) consists of three fully connected layers followed by one pooling

layer. The pooling layer averages instance representations learned by previous layers into a single embedding vector as a bag representation. The last fully connected layer takes the embedding vector as input and outputs the bag prediction. Wang et al.³⁶ examined three typical pooling operators—max pooling, mean pooling, and log-sum-exp pooling—and concluded that all of them provided a similar performance on benchmark data sets. Our tests also supported this conclusion for bioactivity prediction; thus, only mean pooling was applied.

The Bag-Net uses an unlearnable mean pooling function, and as mentioned above, the irrelevant conformations can contribute noise to the prediction and reduce model performance. This drawback can be eliminated using more flexible types of pooling, such as weighted averaging pooling, known as attention. This type of pooling was proposed in another publication,³⁷ where an additional two-layered neural network was used to obtain weights of instances. In the Bag-AttentionNet (Figure 4c), all instances are first fed to three fully connected layers. Then, the learned instance representations are used by the attention network with a single hidden layer. In the attention network, the number of output neurons is equal to the number of instances. The output layer of attention has the Softmax activation function and predicts instance weights. Finally, the instance weights given by the attention network are used for weighted averaging of instance representations to get the embedding vector that is used to produce the bag prediction. Implementation of weighted pooling enables the Bag-AttentionNet to automatically identify probable bioactive conformations.

Experimental Setup. A large-scale comparative analysis of the above MI approaches was carried out using 175 data sets extracted from the ChEMBL database. Each data set was randomly divided into a training, validation, and test set. The test set comprised 20% of the molecules of the initial data set; the rest was used as a modeling set. In turn, the latter was divided into a training set (80% of modeling set) and a validation set (20%) used for hyperparameter adjustment.

The Bag-AttentionNet provides attention weights that determine the contribution of each conformation to the predicted bioactivity. We applied regularization of attention weights to force the Bag-AttentionNet network to more strongly highlight key conformations during training. In each training epoch, instances (conformations) were ranked by the attention unit of Bag-AttentionNet. Then X percent of instances ($X = 10\%, 20\%, 40\%, 60\%, 80\%, 90\%$, and 95%) was discarded and followed by recalculation of the attention weights for the remaining key instances. The number of discarded instances was a hyperparameter adjusted during the training.

To compare several algorithms on multiple data sets, we follow the recommendations from refs 38 and 39. First, we performed the Friedman test to reject the null hypothesis, which is that there were no significant differences in the performance of the models. Then, we performed a pairwise comparison of models using the Wilcoxon–Holm test with a significance level of 5%. The results of a pairwise comparison of the models were visualized with a critical difference diagram.^{38,40} The horizontal lines on the critical difference diagrams connect models that are not significantly different in performance. The pipeline of construction of critical difference diagrams is reported in more detail in the Supporting Information.

For clarity, a special name was assigned to each type of model. It consists of several parts: Representation Level/Learning Type/Algorithm. The Representation Level denotes the descriptors type (2D or 3D). The Learning Type distinguishes between single- and multi-instance learning schemes (SI or MI) or in the case of 2D models the type of descriptors used. The Algorithm is the machine learning method that was used to build the model. For example, 2D/MorganFP/Net denotes a neural network model based on 2D Morgan fingerprints. The 3D/SI/Net model was trained on the single lowest energy conformations of molecules represented by 3D pharmacophore descriptors using an ordinary multilayer neural network as the learning algorithm.

For analysis of groups of models and pairwise model comparison, we excluded the data sets for which all compared approaches resulted in models with low performance on the test set ($R^2_{\text{test}} < 0.4$). Thus, the total number of compared data sets differs as a function of the list of compared models.

We consider that the threshold 0.4 is reasonable for several reasons. (i) We did not tweak every model too much; therefore, we believe there is a room to improve them with tight tuning. (ii) We performed only a comparison between models and do not suggest using them for predictive modeling. (iii) The results and conclusions do not change if we will choose threshold 0.5, but this will decrease the number of considered data sets. Nevertheless, all data are disclosed in the Supporting Information (Tables S1 and S4), and these conclusions can be verified.

RESULTS AND DISCUSSION

In this section, we present the results of a comparative analysis of single- and multi-instance learning approaches. For clarity, we first present the results of benchmarking MI learning algorithms to choose the best MI models. Then, we compare the best 3D MI models with 3D SI models as well as with conventional 2D SI models and evaluate the ability of MI models to identify relevant conformations in comparison to docking.

Benchmarking of Multi-Instance Algorithms. For 45 data sets out of 175, no MI models achieved the required performance of $R^2_{\text{test}} > 0.4$. These “non-modellable” data sets were excluded from further consideration, and benchmarking analysis was performed on the remaining 130 data sets. Among the two simplest approaches represented by wrapper algorithms, Instance-Wrapper performs significantly better than Bag-Wrapper (Figure 5). Thus, considering each conformation as an individual training example represents a better strategy than averaging descriptor vectors of individual conformations.

MI neural networks represent a group of methods specially modified to solve MI problems. In Bag-Net, the mean pooling operation is performed not on descriptors of particular

conformations (as in Bag-Wrapper) but on their embeddings, resulting from descriptors transformation by three fully connected layers of the neural network. Comparative analysis shows that there is no significant difference in performance between the Bag-Net and the Bag-Wrapper models. To increase the contribution of the relevant conformations during training of the model, the Bag-Net architecture was enhanced by the attention mechanism (Bag-AttentionNet). This, however, does not lead to a significant increase in the predictive performance of the model (Figure 5).

Overall, the analysis shows that the Instance-Wrapper algorithm largely outperforms all other studied MI algorithms. Other algorithms demonstrated comparable performance, despite the substantial differences in their architecture.

Comparison of 2D and 3D Models. There is an ongoing discussion about the preference of 2D and 3D descriptors in QSAR. An important step in building QSAR models with 3D descriptors concerns the selection of the bioactive conformation, which is hard to do reasonably without some additional information. An MI model is free from the problem of arbitrary selection of conformations. It considers all conformations and automatically selects the most relevant ones. We compared MI models with 2D models to estimate the importance of accounting for 3D information in bioactivity prediction and to assess contributions of particular conformations.

Six approaches were compared: three classical approaches based on 2D molecular descriptors, a 3D single-instance approach based on 3D pharmacophore descriptors calculated for the lowest energy conformations, and two 3D multi-instance approaches based on all generated conformations of each molecule represented by 3D pharmacophore descriptors. Among the MI approaches we chose the best performing Instance-Wrapper algorithm and the most advanced Bag-AttentionNet algorithm. For the sake of clarity, 33 “non-modellable” data sets for which none of the considered 2D and 3D models had $R^2_{\text{test}} > 0.4$ were excluded, and the analysis was performed based on the remaining 142 data sets.

Table 1 presents the mean R^2_{test} of models across the chosen 142 data sets. The 3D SI models built with one conformation

Table 1. Performance Comparison of 2D and 3D Models^a

model	mean	median	top 1	top 2
3D/MI/Instance-Wrapper	0.524 ± 0.131	0.526	69	105
3D/MI/Bag-Attention	0.468 ± 0.161	0.474	12	57
2D/MorganFP/Net	0.464 ± 0.199	0.502	39	66
2D/PhysChem/Net	0.450 ± 0.144	0.443	17	37
2D/PharmFP/Net	0.382 ± 0.216	0.404	4	17
3D/SI/Net	0.024 ± 0.372	0.089	1	2

^aTable reports mean, standard deviations, and median of R^2_{test} . Top 1 is the number of cases where the model was the best. Top 2 is the number of cases where the model was the first- or second-best one.

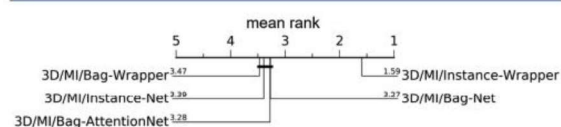


Figure 5. Comparison of MI algorithms against each other. Groups of models that are not significantly different in performance (at a confidence level of 0.05) are connected by the horizontal line. Axis plots the average ranks of models.

per molecule demonstrated poor performance (mean $R^2_{\text{test}} = 0.024$) in comparison with the other models. The poor performance of 3D SI models can be explained by the ambiguous strategy when only one lowest energy conformation is considered. The lowest energy conformation might substantially differ from the actual bioactive conformation responsible for the observed bioactivity of the molecule. However, the performance of the 3D models drastically increases as soon as all available generated conformations are considered. Mean R^2_{test} values of 0.524 and 0.468 were

obtained, respectively, for the 3D/MI/Instance-Wrapper and 3D/MI/Bag-AttentionNet models. The former even outperforms the 2D models built with Morgan fingerprints (mean $R^2_{\text{test}} = 0.464$). The 3D/MI/Instance-Wrapper models displayed the highest R^2_{test} in almost 49% of the cases (69 out of 142 data sets), and they were in the top 2 models for 105 data sets. The 2D/MorganFP/Net models were the best in 27% of the cases (39 out of 142 data sets). The other 2D models based on physicochemical or pharmacophore descriptors had poorer performance than those based on Morgan fingerprints.

The critical differences diagram of a pairwise comparison of the 2D and 3D models is shown in Figure 6. The 3D/MI/

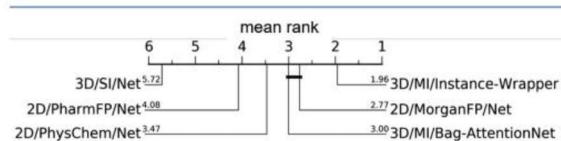


Figure 6. Comparison of 2D SI, 3D SI, and 3D MI models against each other. Similarly performed models (at a confidence level of 0.05) are connected by the horizontal line. Numbers correspond to the average ranks of models.

Instance-Wrapper models have an average rank of 1.96 and outperform the 2D/MorganFP/Net model having an average rank of 2.77. The 3D/MI/Net model showed the worst performance across almost all data sets (average rank 5.72).

We calculated the number of wins and losses to perform a pairwise comparison of models (Table 2). Wins represent the

Table 2. Pairwise Comparison of Models^a

	wins	losses	ties	inconclusive
3D/MI/Instance-Wrapper vs 2D/MorganFP/Net	90	48	1	36
3D/MI/Bag-AttentionNet vs 2D/MorganFP/Net	50	72	0	53
3D/MI/Net vs 2D/MorganFP/Net	3	99	0	73
3D/MI/Instance-Wrapper vs 3D/MI/Net	122	2	0	51
3D/MI/Bag-AttentionNet vs 3D/MI/Net	97	4	0	74

^aWins are the number of data sets for which the accuracy of predicting the biological activity of ligands by the first model is higher than that of the second (model 1 vs model 2). Losses are counted as the number of data sets where the biological activity of the ligands is more accurately predicted by the second model. Ties are the number of data sets where the accuracy of both models is equal. Inconclusive is the number of data sets where R^2_{test} of both models was less than 0.4.

number of tasks where the R^2_{test} of the first model was higher than that of the second model and at least one model had $R^2_{\text{test}} > 0.4$. For example, 3D/MI/Instance-Wrapper outperformed 3D/MI/Net in 122 out of 124 data sets (98%), and its R^2_{test} was higher than that of 2D/MorganFP/Net in 90 out of 139 data sets (65%).

The 3D/MI/Instance-Wrapper models outperformed the 3D SI models in almost all cases except for few data sets for which quite similar prediction accuracy between the two approaches was observed (see Figure 7a). The 3D/MI/Instance-Wrapper models also outperformed 2D models in many cases. However, a large variability of model performances was observed for these two approaches. Most notably, in 38 out of 142 compared data sets the 3D/MI/Instance-Wrapper

models achieved $R^2_{\text{test}} > 0.4$, while the 2D models had $R^2_{\text{test}} < 0.4$. This means that using multiple conformations in the model building may significantly improve the model performance, and if the 2D models fail one may try to apply 3D MI-QSAR approaches.

We investigated which factors can distinguish cases where 3D MI models outperformed 2D conventional ones. We analyzed the distribution of the physicochemical characteristics of data sets where the 3D/MI/Instance-Wrapper models outperform 2D models and vice versa. The data sets were divided into two groups. The first group consisted of 42 data sets, for which the 3D/MI/Instance-Wrapper models were significantly ($\Delta R^2 \geq 0.1$) better than the 2D models. The second group included 18 data sets where the 2D models outperformed ($\Delta R^2 \geq 0.1$) the 3D/MI/Instance-Wrapper models. We established that the smaller number of rotatable bonds is more favorable for the 3D/MI/Instance-Wrapper models than for the 2D (Figure 8a). This may be caused by the poorer ability of the conformer generator to generate biologically relevant conformations for more flexible compounds.⁴¹ The 3D/MI/Instance-Wrapper models were favorable in cases where the fraction of unique Murcko frameworks (the ratio of the number of unique scaffolds in the data set to the total number of molecules) was high (Figure 8b). This corresponds to data sets with higher scaffold diversity which are more difficult for the 2D models. Similar box plots were created for other characteristics of the data sets (see Supplementary Figure S3), but on average they cannot distinguish cases where the 3D/MI/Instance-Wrapper models dominate.

■ IDENTIFICATION OF BIOACTIVE CONFORMATIONS

The attention mechanism allows the 3D/MI/Bag-AttentionNet models to identify the most relevant conformations during the learning. The question arises of how accurately the attention mechanism recognizes the bioactive conformation? To answer this question, we chose the 3D/MI/Bag-AttentionNet models with $R^2_{\text{test}} > 0.4$. Then, the 3D structures of the ligands were extracted from the protein–ligand complexes retrieved from the PDB database. Since these data were sparse, four data sets having at least 10 test set molecules with available information about the bioactive conformation were chosen for the subsequent analysis. Experimental bioactive conformations were compared with the three top conformations that received the highest attention weights from the 3D/MI/Bag-AttentionNet. To measure the accuracy of identification of the bioactive conformations, we calculated the top 3 success rate as a proportion of compounds for which at least one of the three best conformations fits the experimental structure with RMSD < 2.0 Å.

To compare the accuracy of identification of relevant conformations with docking, we chose for each protein target a PDB complex with a binding site intersected with most of the binding sites of other complexes and used it for docking of the same test set compounds (ChEMBL2820-4Y8Y, ChEMBL3048-4IMS, ChEMBL335-3EAX, and ChEMBL4802-4KCQ). This was more fair than performing redocking to cognate receptor structures because in the case of machine learning we do not use information about the receptor conformation to select a relevant conformation. Docking was performed using AutoDock Vina.⁴² Three top-

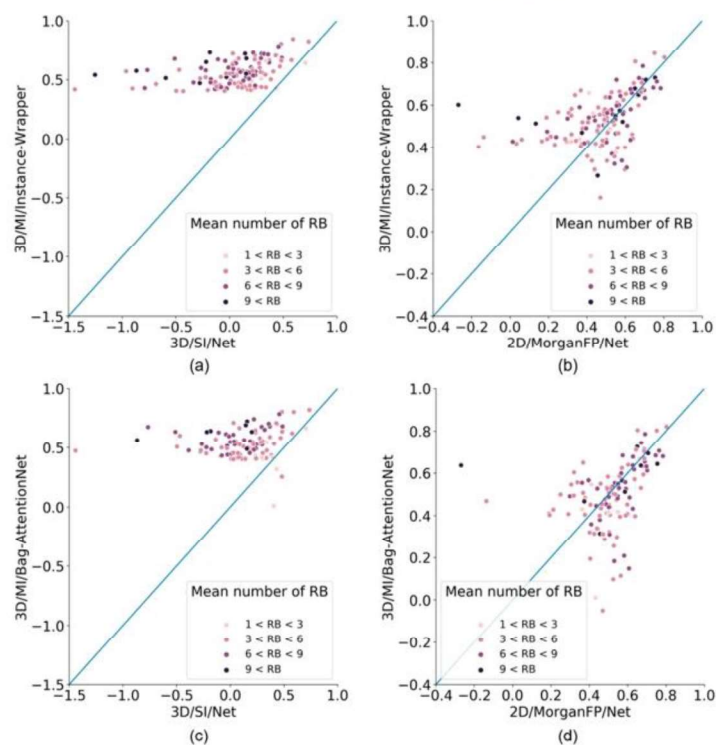


Figure 7. Correlation between the R^2_{test} values computed for the 2D and 3D models across all data sets. Each point represents a particular data set. Color code encodes the mean number of rotatable bonds (RB) of molecules in a given data set.

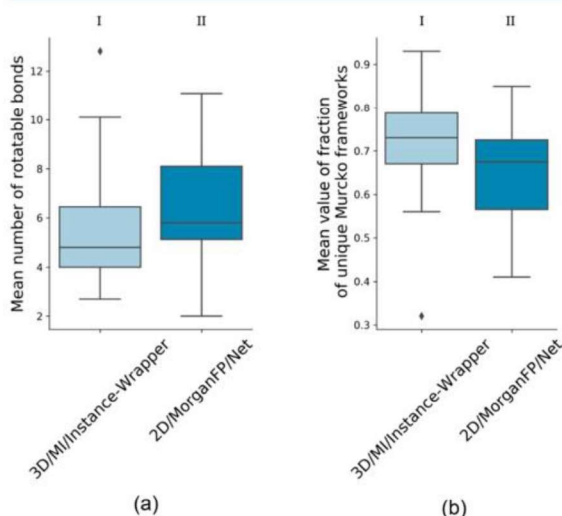


Figure 8. Distribution of the mean number of rotatable bonds (a) and mean values of the fraction of unique Murcko frameworks (b) in the ensemble of 42 data sets (I), where the 3D/MI/Instance-Wrapper models are significantly better than the 2D models ($\Delta R^2 \geq 0.1$) and in the ensemble of 18 data sets (II) where 2D models outperform 3D/MI/Instance-Wrapper ($\Delta R^2 \geq 0.1$).

scored poses were taken to calculate the top 3 statistics similarly as described above.

Since it was claimed that the RDKit conformer generator can reproduce bioactive conformations, we calculated two baseline statistics. The first one used three conformations with the lowest estimated energy. The second one corresponds to the top 3 metric value of three randomly chosen conformations for each molecule. We calculated the probability of choosing at least one conformation with the RMSD below 2 Å among the three randomly selected ones for each molecule and averaged these values across the test molecules.

The calculated random baseline statistics was relatively high (Figure 9). This indicates that the RDKit conformer generator substantially enriches the set of conformations with those which are close to the experimental ones. This also makes it challenging to improve the statistics. Selection of three conformations with the lowest energy performed comparably to random choice, and in the case of ChEMBL2820, the performance was even worse. The 3D/MI/Bag-AttentionNet models could improve the baseline accuracy in the identification of the bioactive conformations and perform comparably well or better than the random choice. The most remarkable improvement was for coagulation factor XI (ChEMBL2820). For two targets, brain and endothelial nitric-oxide synthases (ChEMBL3048 and ChEMBL4802, correspondingly), 3D/MI/Bag-AttentionNet performed comparably to the baseline. Protein-tyrosine phosphatase 1B (ChEMBL335) was the most difficult target for the identification of relevant conformations, and all approaches demonstrated low performance. This was caused by the fact that only part of those compounds binds to the protein; the remaining part was pretty flexible and exposed to a water

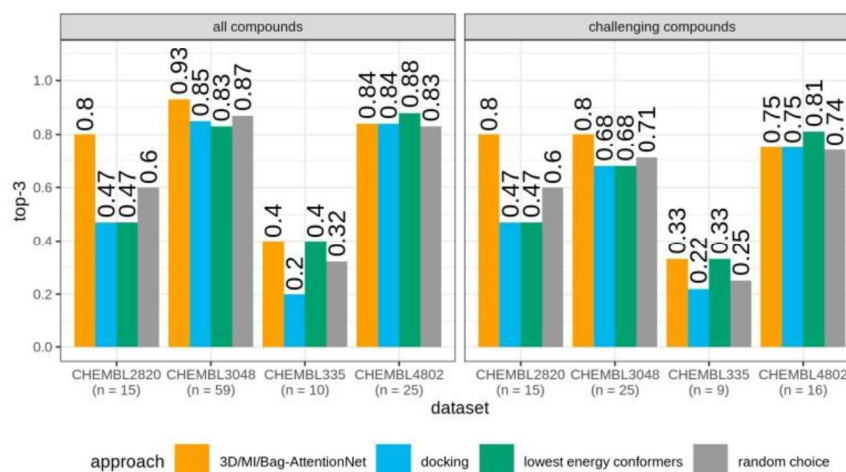


Figure 9. Identification of bioactive conformations within the test set compounds for four data sets (n is the number of compounds). Challenging compounds is a subset of test set compounds that have mean RMSDs of all generated conformations to a bioactive conformation greater than 2 Å. R^2_{test} values of the 3D/MI/Bag-AttentionNet models were 0.49, 0.52, 0.74, and 0.55 for CHEMBL2820, CHEMBL3048, CHEMBL335, and CHEMBL4802 data sets, correspondingly.

medium. Therefore, even docking could not identify true poses. In general, docking performed relatively poorly and slightly worse than the random baseline in the case of CHEMBL2820 and CHEMBL335. Examples of the lowest energy conformations and the conformations predicted by the 3D/MI/Bag-AttentionNet model in comparison with the experimental ones retrieved from the PDB are shown in Figure 10.

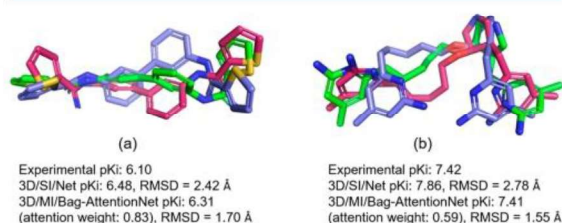


Figure 10. Examples of experimental, lowest energy, and predicted conformations: green, PDB conformation; blue, the lowest energy conformation; red, conformation predicted by the 3D/MI/Bag-AttentionNet model. RMSD to the experimental conformation is given.

In addition, we considered subsets of “challenging” compounds with a mean RMSD to the bioactive conformation greater than 2 Å. These subsets were enriched by very flexible compounds for which diverse sets of conformations were generated. As expected, the performance of key conformation identification for these compounds was lower (Figure 9), but 3D/MI/Bag-Attention had a performance comparable with or higher than the random baseline, supporting an intelligent selection of relevant conformations.

CONCLUSION

This study reports a large-scale comparison of single- and multi-instance machine learning algorithms for predicting the biological activity of chemical compounds. The molecules were represented either by the lowest energy conformation (single-instance) or by a set of generated conformations (multi-

instances). The multi-instance learning algorithms reduce the problem of ambiguous selection of a putative bioactive conformation and simultaneously consider all available conformations in the model building.

The present study is the first comprehensive comparison of MI approaches with traditional QSAR based on 2D and 3D descriptors. The results demonstrate that multi-instance models generally outperform both single-instance 3D models and traditional QSAR models built on 2D Morgan fingerprints (mean R^2_{test} = 0.524, 0.024, and 0.464, respectively). Surprisingly, on average, the application of 3D descriptors of the lowest energy conformation for QSAR modeling was only slightly better than the null model. Thus, the highest accuracy in the bioactivity predictions is achieved by the multi-instance algorithm since it considers the whole conformational space of an individual training object. This result demonstrates the importance of accounting for the dynamic nature of chemical objects for QSAR modeling.

Contrary to a previous finding,⁴³ our study shows that 3D descriptors of molecules in combination with the MI learning approach can compete with traditional 2D QSAR. Notably, there were 38 data sets where the MI learning approach showed reasonable performance while the traditional 2D QSAR model failed. This means that the MI learning approach can be applied in cases where 2D QSAR modeling fails.

Last but not least, a multi-instance neural network with an attention mechanism can correctly identify a “bioactive” conformation close to the experimental structure of a ligand retrieved from the PDB. However, it should be noted that the performance of the multi-instance models depends on the conformer generator used. The RDKit conformer generator demonstrated a good ability to generate biologically relevant conformations that was confirmed by a relatively high random choice baseline estimate.

To facilitate the community being able to apply the MI learning approach for QSAR modeling, a set of MI learning algorithms based on different MI neural network architectures as well as wrappers used in the work are available at <https://github.com/cimm-kzn/3D-MIL-QSAR>.

■ ASSOCIATED CONTENT

■ Supporting Information

The Supporting Information is available free of charge at <https://pubs.acs.org/doi/10.1021/acs.jcim.1c00692>.

Description of statistical analysis details, comparison of 2D and 3D models, and distribution of molecular characteristics for data sets that are well and poorly predicted by MIL and 2D QSAR models (PDF)

■ AUTHOR INFORMATION

Corresponding Authors

Alexandre Varnek – Laboratory of Chemoinformatics, Institute Le Bel, University of Strasbourg, 67081 Strasbourg, France; orcid.org/0000-0003-1886-925X; Email: varnek@unistra.fr

Pavel Polishchuk – Institute of Molecular and Translational Medicine, Faculty of Medicine and Dentistry, Palacký University and University Hospital in Olomouc, 77900 Olomouc, Czech Republic; orcid.org/0000-0001-5088-8149; Email: pavlo.polishchuk@upol.cz

Timur I. Madzhidov – Laboratory of Chemoinformatics and Molecular Modeling, A. M. Butlerov Institute of Chemistry, Kazan Federal University, 420111 Kazan, Russia; orcid.org/0000-0002-3834-6985; Email: timur.madzhidov@kpfu.ru

Authors

Dmitry V. Zankov – Laboratory of Chemoinformatics and Molecular Modeling, A. M. Butlerov Institute of Chemistry, Kazan Federal University, 420111 Kazan, Russia; Laboratory of Chemoinformatics, Institute Le Bel, University of Strasbourg, 67081 Strasbourg, France; orcid.org/0000-0002-6201-3347

Mariia Matveieva – Institute of Molecular and Translational Medicine, Faculty of Medicine and Dentistry, Palacký University and University Hospital in Olomouc, 77900 Olomouc, Czech Republic

Aleksandra V. Nikonenko – Institute of Molecular and Translational Medicine, Faculty of Medicine and Dentistry, Palacký University and University Hospital in Olomouc, 77900 Olomouc, Czech Republic

Ramil I. Nugmanov – Laboratory of Chemoinformatics and Molecular Modeling, A. M. Butlerov Institute of Chemistry, Kazan Federal University, 420111 Kazan, Russia; orcid.org/0000-0002-8541-9681

Igor I. Baskin – Department of Materials Science and Engineering, Technion—Israel Institute of Technology, 3200003 Haifa, Israel; orcid.org/0000-0003-0874-1148

Complete contact information is available at: <https://pubs.acs.org/doi/10.1021/acs.jcim.1c00692>

Notes

The authors declare no competing financial interest. All implemented algorithms and data sets are freely available at <https://github.com/cimm-kzn/3D-MIL-QSAR>.

■ ACKNOWLEDGMENTS

Development of 3D pharmacophore descriptors was funded by the Ministry of Education, Youth and Sports of the Czech Republic within the INTER-EXCELLENCE LTARF18013 project (agreement no. MSMT-5727/2018-2) and by the Ministry of Science and Higher Education of the Russian

Federation (agreement no. 14.587.21.0049, unique identifier RFMEFI58718X0049). Development of the neural nets architectures for the prediction of chemical object properties was supported by the Russian Science Foundation (agreement no. 19-73-10137).

■ REFERENCES

- (1) In 3D QSAR in Drug Design; Kubinyi, H., Ed.; Springer: Netherlands, 1994.
- (2) Cramer, R. D.; Patterson, D. E.; Bunce, J. D. Comparative Molecular Field Analysis (CoMFA). 1. Effect of Shape on Binding of Steroids to Carrier Proteins. *J. Am. Chem. Soc.* **1988**, *110* (18), 5959–5967.
- (3) Wendt, B.; Cramer, R. D. Challenging the Gold Standard for 3D-QSAR: Template CoMFA versus X-Ray Alignment. *J. Comput.-Aided Mol. Des.* **2014**, *28* (8), 803–824.
- (4) Silverman, B. D.; Platt, D. E. Comparative Molecular Moment Analysis (CoMMA): 3D-QSAR without Molecular Superposition. *J. Med. Chem.* **1996**, *39* (11), 2129–2140.
- (5) Todeschini, R.; Gramatica, P. The Whim Theory: New 3D Molecular Descriptors for Qsar in Environmental Modelling. *SAR QSAR Environ. Res.* **1997**, *7* (1–4), 89–115.
- (6) Hopfinger, A. J. A QSAR Investigation of Dihydrofolate Reductase Inhibition by Baker Triazines Based upon Molecular Shape Analysis. *J. Am. Chem. Soc.* **1980**, *102* (24), 7196–7206.
- (7) Pastor, M.; Cruciani, G.; Watson, K. A. A Strategy for the Incorporation of Water Molecules Present in a Ligand Binding Site into a Three-Dimensional Quantitative Structure-Activity Relationship Analysis. *J. Med. Chem.* **1997**, *40* (25), 4089–4102.
- (8) Woolfrey, J. R.; Avery, M. A.; Doweyko, A. M. Comparison of 3D Quantitative Structure-Activity Relationship Methods: Analysis of the in Vitro Antimalarial Activity of 154 Artemisinin Analogues by Hypothetical Active-Site Lattice and Comparative Molecular Field Analysis. *J. Comput.-Aided Mol. Des.* **1998**, *12* (2), 165–181.
- (9) Klebe, G.; Abraham, U.; Mietzner, T. Molecular Similarity Indices in a Comparative Analysis (CoMSIA) of Drug Molecules to Correlate and Predict Their Biological Activity. *J. Med. Chem.* **1994**, *37* (24), 4130–4146.
- (10) Hemmer, M. C.; Steinhauer, V.; Gasteiger, J. Deriving the 3D Structure of Organic Molecules from Their Infrared Spectra. *Vib. Spectrosc.* **1999**, *19* (1), 151–164.
- (11) Schuur, J. H.; Selzer, P.; Gasteiger, J. The Coding of the Three-Dimensional Structure of Molecules by Molecular Transforms and Its Application to Structure-Spectra Correlations and Studies of Biological Activity. *J. Chem. Inf. Comput. Sci.* **1996**, *36* (2), 334–344.
- (12) Daré, J. K.; Freitas, M. P. Different Approaches to Encode and Model 3D Information in a MIA-QSAR Perspective. *Chemom. Intell. Lab. Syst.* **2021**, *212*, 104286.
- (13) Hopfinger, A. J.; Wang, S.; Tokarski, J. S.; Jin, B.; Albuquerque, M.; Madhav, P. J.; Duraiswami, C. Construction of 3D-QSAR Models Using the 4D-QSAR Analysis Formalism. *J. Am. Chem. Soc.* **1997**, *119* (43), 10509–10524.
- (14) Lill, M. A. Multi-Dimensional QSAR in Drug Discovery. *Drug Discovery Today* **2007**, *12* (23–24), 1013–1017.
- (15) Dreher, J.; Scheiber, J.; Stiefl, N.; Baumann, K. XMaP: An Interpretable Alignment-Free Four-Dimensional Quantitative Structure-Activity Relationship Technique Based on Molecular Surface Properties and Conformer Ensembles. *J. Chem. Inf. Model.* **2018**, *58* (1), 165–181.
- (16) Vedani, A.; Dobler, M. Multi-Dimensional QSAR in Drug Research. In *Progress in Drug Research*; Birkhäuser Basel: Basel, 2000; pp 105–135.
- (17) Potemkin, V. A.; Arslambekov, R. M.; Bartashevich, E. V.; Grishina, M. A.; Belik, A. V.; Perspicace, S.; Guccione, S. Multiconformational Method for Analyzing the Biological Activity of Molecular Structures. *J. Struct. Chem.* **2002**, *43* (6), 1045–1049.
- (18) Kuz'min, V. E.; Artemenko, A. G.; Polishchuk, P. G.; Muratov, E. N.; Hromov, A. I.; Liahovskiy, A. V.; Andronati, S. A.; Makan, S. Y.

- Hierarchic System of QSAR Models (1D–4D) on the Base of Simplex Representation of Molecular Structure. *J. Mol. Model.* **2005**, *11* (6), 457–467.
- (19) Keeler, J. D.; Rumelhart, D. E.; Leow, W. K. Integrated Segmentation and Recognition of Hand-Printed Numerals. *Advances in Neural Information Processing Systems 3*; Lippmann, R. P., Moody, J., Touretzky, D. S., Eds.; Morgan Kaufmann: San Mateo, CA, 1991; pp 557–563.
- (20) Dietterich, T. G.; Lathrop, R. H.; Lozano-Pérez, T. Solving the Multiple Instance Problem with Axis-Parallel Rectangles. *Artif. Intell.* **1997**, *89* (1–2), 31–71.
- (21) Jain, A. N.; Dietterich, T. G.; Lathrop, R. H.; Chapman, D.; Critchlow, R. E.; Bauer, B. E.; Webster, T. A.; Lozano-Perez, T. Compass: A Shape-Based Machine Learning Tool for Drug Design. *J. Comput.-Aided Mol. Des.* **1994**, *8* (6), 635–652.
- (22) Carboneau, M. A.; Cheplygina, V.; Granger, E.; Gagnon, G. Multiple Instance Learning: A Survey of Problem Characteristics and Applications. *Pattern Recognit.* **2018**, *77*, 329–353.
- (23) Bergeron, C.; Zaretski, J.; Breneman, C.; Bennett, K. P. Multiple Instance Ranking. *Proceedings of the 25th International Conference on Machine Learning*; 2008; pp 48–55.
- (24) Fu, G.; Nan, X.; Liu, H.; Patel, R. Y.; Daga, P. R.; Chen, Y.; Wilkins, D. E.; Doerksen, R. J. Implementation of Multiple-Instance Learning in Drug Activity Prediction. *BMC Bioinf.* **2012**, *13*, S3.
- (25) Zhao, Z.; Fu, G.; Liu, S.; Elokely, K. M.; Doerksen, R. J.; Chen, Y.; Wilkins, D. E. Drug Activity Prediction Using Multiple-Instance Learning via Joint Instance and Feature Selection. *BMC Bioinf.* **2013**, *14*, S16.
- (26) Zankov, D.; Polishchuk, P.; Madzhidov, T.; Varnek, A. Multi-Instance Learning Approach to Predictive Modeling of Catalysts Enantioselectivity. *Synlett* **2021**. DOI: 10.1055/a-1553-0427
- (27) Nikonenko, A.; Zankov, D.; Baskin, I.; Madzhidov, T.; Polishchuk, P. Multiple Conformer Descriptors for QSAR Modeling. *Mol. Inf.* **2021**. DOI: 10.1002/minf.202060030.
- (28) Zankov, D. V.; Shevelev, M. D.; Nikonenko, A. V.; Polishchuk, P. G.; Rakhimbekova, A. I.; Madzhidov, T. I. Multi-Instance Learning for Structure-Activity Modeling for Molecular Properties. In *Analysis of Images, Social Networks and Texts. AIST 2019. Communications in Computer and Information Science*; van der Aalst, W. M. P., Batagelj, V., Ignatov, D. I., Khachay, M., Kuskova, V., Kutuzov, A., Kuznetsov, S. O., Lomazova, I. A., Loukachevitch, N., Napoli, A., Pardalos, P. M., Pelillo, M., Savchenko, A. V., Tutubalina, E., Eds.; AIST, 2020; pp 62–71.
- (29) Liu, G.; Wu, J.; Zhou, Z.-H. Key Instance Detection in Multi-Instance Learning. *Proceedings of the Asian Conference on Machine Learning*; 2012; pp 253–268.
- (30) Riniker, S.; Landrum, G. A. Better Informed Distance Geometry: Using What We Know To Improve Conformation Generation. *J. Chem. Inf. Model.* **2015**, *55* (12), 2562–2574.
- (31) Kutlushina, A.; Khakimova, A.; Madzhidov, T.; Polishchuk, P. Ligand-Based Pharmacophore Modeling Using Novel 3D Pharmacophore Signatures. *Molecules* **2018**, *23* (12), 3094.
- (32) Madzhidov, T. I.; Rakhimbekova, A.; Kutlushina, A.; Polishchuk, P. Probabilistic Approach for Virtual Screening Based on Multiple Pharmacophores. *Molecules* **2020**, *25* (2), 385.
- (33) Polishchuk, P.; Kutlushina, A.; Bashirova, D.; Mokshyna, O.; Madzhidov, T. Virtual Screening Using Pharmacophore Models Retrieved from Molecular Dynamic Simulations. *Int. J. Mol. Sci.* **2019**, *20* (23), 5834.
- (34) Mayr, A.; Klambauer, G.; Unterthiner, T.; Steijaert, M.; Wegner, J. K.; Ceulemans, H.; Clevert, D.-A.; Hochreiter, S. Large-Scale Comparison of Machine Learning Methods for Drug Target Prediction on ChEMBL. *Chem. Sci.* **2018**, *9* (24), 5441–5451.
- (35) Amores, J. Multiple Instance Classification: Review, Taxonomy and Comparative Study. *Artif. Intell.* **2013**, *201*, 81–105.
- (36) Wang, X.; Yan, Y.; Tang, P.; Bai, X.; Liu, W. Revisiting Multiple Instance Neural Networks. *Pattern Recognit.* **2018**, *74*, 15–24.
- (37) Ilse Maximilian, J. M. T.; Welling, M. Attention-based Deep Multiple Instance Learning <https://github.com/AMLab-Amsterdam/AttentionDeepMIL>.
- (38) Demšar, J. Statistical Comparisons of Classifiers over Multiple Data Sets. *J. Mach. Learn. Res.* **2006**, *7* (Jan), 1–30.
- (39) Benavoli, A.; Corani, G.; Mangili, F. Should We Really Use Post-Hoc Tests Based on Mean-Ranks? *J. Mach. Learn. Res.* **2016**, *17* (1), 152–161.
- (40) Ismail Fawaz, H.; Forestier, G.; Weber, J.; Idoumghar, L.; Muller, P.-A. Deep Learning for Time Series Classification: A Review. *Data Min. Knowl. Discovery* **2019**, *33* (4), 917–963.
- (41) Ebejer, J.-P.; Morris, G. M.; Deane, C. M. Freely Available Conformer Generation Methods: How Good Are They? *J. Chem. Inf. Model.* **2012**, *52* (5), 1146–1158.
- (42) Trott, O.; Olson, A. J. AutoDock Vina: Improving the Speed and Accuracy of Docking with a New Scoring Function, Efficient Optimization, and Multithreading. *J. Comput. Chem.* **2010**, *31* (2), 455–461.
- (43) Kyaw Zin, P. P.; Borrel, A.; Fourches, D. Benchmarking 2D/3D/MD-QSAR Models for Imatinib Derivatives: How Far Can We Predict? *J. Chem. Inf. Model.* **2020**, *60*, 3342–3360.

Conclusion

A new 3D modeling approach based on conformer ensembles was applied to build 3D multi-conformer models which were compared with 3D single-conformer and 2D models on the collection of 175 datasets extracted from the ChEMBL-23 database. In a pairwise comparison, 3D multi-conformer models almost for all datasets (99%) outperformed 3D single-conformer models. In total, 3D multi-conformer model demonstrated the highest performance in 63 % of datasets, while the 2D model was the best in 36% of datasets. Nevertheless, there were a few datasets in which 2D models failed to predict the bioactivity of compounds, while 3D multi-conformer models provided accurate predictions. This may indicate special cases where 3D structural information is crucial for the correct prediction of bioactivity.

It was demonstrated, that the 3D multi-conformer models, built with the attention-based multi-instance neural network, can also identify the bioactive conformers. For 3 of the 4 datasets, the 3D multi-conformer model identified more bioactive conformers than the standard docking approach. For example, for 15 experimental 3D structures from the ChEMBL2820 dataset, the lowest energy and docking conformers correctly fit “bioactive” conformers for only 7 molecules, which is even worse than the random selection (9 molecules). Meanwhile, the 3D multi-conformer model correctly identifies bioactive conformers for 12 molecules.

The developed 3D modeling approach does not require selection and alignment of conformers, which excludes manual configuration of the modeling protocol (but there are still options to improve the performance of the 3D models, such as optimization of the number of conformations, hyperparameters of machine learning algorithms, adjustment of descriptors, validation strategy, etc.). Concerning future research, there are still many other popular 2D descriptors that can be tested in the described benchmark. In the case of the 3D models, apart from the lowest-energy conformation, there are other strategies (docking or other conformer generators) to select a single conformer for modeling. Also, the benchmark analysis was designed to isolate the influence of the machine learning algorithm (as much as possible), and all 2D and 3D models were built using the standard fully-connected neural network or its multi-instance modification. However, there are many other traditional single-instance algorithms and multi-instance algorithms that can be used for building 2D and 3D models.

2.4 Modeling of catalysts enantioselectivity with conformation ensembles

Introduction

Synthesis of enantiopure compounds is a hot topic of modern organic chemistry because highly effective drugs can be chiral and enantiomers often have different biological activities. In 2021, B. List and D. McMillan were awarded the Nobel Prize for the development of asymmetric organocatalysis. In 2000 they demonstrated [105,106] that chiral organic molecules can effectively catalyze asymmetric reactions with production enantiopure compounds. Since these seminal publications, numerous chiral catalyst systems have been designed [141]. The pursuit of perspective catalysts is traditionally conducted by iterative modification of the catalyst structure aiming to increase the enantioselectivity of the considered reaction. In this process, chemists rely on their professional experience, chemical intuition, and available experimental data. This approach, albeit often culminates in the desired result, still depends on the professional background of the researcher. Despite significant progress in experimental studies of asymmetric organocatalysis, computational chemistry is an appealing technology aiming to empower experimentalists in the quest for developing new catalysts. Theoretical calculations may suggest the structure of promising catalysts before their synthesis, and experimental testing, thus, reducing the time and overheads needed to achieve their desired performance.

A perspective computational approach to the theoretical discovery of new catalysts is Quantitative Structure-Selectivity Relationship (QSSR) analysis, which applies machine learning algorithms to find the relation between experimental enantioselectivity and the catalyst structure encoded by numerical descriptors. If a correct relationship between structure and selectivity is established, the obtained model can be used for the virtual screening of candidate catalysts. The first notable example of the application of QSSR in enantioselective catalysis was published by Norrby et al. [142], where computational steric molecular descriptors (bond lengths, bond angles, and dihedral angles of metal complex) and multivariate regression were used to analyze palladium-catalyzed allylation.

Most other early studies on QSSR applied 3D modelling techniques based on the Molecular Interaction Fields (MIF) [143] approaches. MIF approaches locate molecule in the 3D grid and compute interaction energies between molecule and probe atoms/charges fixed on the grid around the molecule. The most popular MIF-based approach is a Comparative Molecular Field Analysis (CoMFA), in which 3D molecular structures (one conformer per molecule) are aligned and then placed in a 3D grid where steric and electrostatic energies with a probe are calculated in the grid nodes. Obtained steric and electrostatic descriptors are then correlated with experimental activity.

In 2003 Lipkowitz et al. [108] demonstrated the first application of CoMFA to the prediction of catalyst enantioselectivity in Diels-Alder reactions. Kozlowski et al. [109] described a QSSR approach for aldehyde alkylation with aminoalkoxide zinc catalysts, where semi-empirical methods (PM3) were used to obtain reaction transition structures, which then were aligned and processed in the calculation of interaction energies with 2s electron probe grid points.

In 2004 Melville et al. [111] published a 3D-QSSR approach based on the classic CoMFA for the glycine imine alkylation with quaternary ammonium ion catalysts in asymmetric phase-transfer catalysis (APTC). They validated the proposed approach on a library of 88 cinchona alkaloid-based catalysts and obtained accurate predictions of catalyst enantioselectivity on an external test set, which contained catalysts with a new substituent not occurring in the training set. Considering the same reaction, later in 2005 Melville and coworkers [144], focused on the conformation diversity of catalysts and applied 4D-QSAR to model the enantioselectivity of biphenyl catalysts, thereby improving the accuracy of predictions in comparison with the standard 3D-QSSR model. In the same paper, they proposed an advanced 3.5D-QSSR approach with Boltzmann-weighting of selected catalyst conformers and obtained enantioselectivity predictions even more accurately than in 4D-QSSR. Their results demonstrated the importance of molecular flexibility in enantioselectivity modelling, which was addressed in later studies [110,145]. In 2011 the asymmetric glycine imine alkylation catalyzed with a pyrrolizidine-based system was analyzed by Denmark group [146] using CoMFA-based approach. To account for conformation diversity, they generated five libraries with different combinations of scaffold conformers. This approach generates accurate predictions if a proper conformer library is selected.

The development of various methods and approaches to QSSR analysis in asymmetric synthesis culminated in the general chemoinformatics-based approach published by Denmark's group [110,145] in 2019. In this work, they explicitly state the necessity of incorporating conformation diversity into the modelling process and propose novel 3D Average Steric Occupancy (ASO) descriptors accumulating steric information from multiple catalyst conformers. They tested their approach to predicting enantioselectivity in the reaction of asymmetric addition of thiols to imines catalyzed by phosphoric acids and demonstrated that multiple conformer descriptors outperform single conformer variants.

Besides the selection of relevant conformers, the other important limitation of MIF approaches is conformers alignment. If analyzed molecules share a common scaffold, conformers alignment is a trivial process. Otherwise, if the molecules have different scaffolds, conformers alignment becomes problematic. This issue initiated the development of alignment-free 3D descriptors that are invariant to the position or orientation of the molecule in space. The first example

of the use of MIF-based alignment-independent descriptors in asymmetric catalysis was the application of GRid Independent Descriptors (GRIND)[147] demonstrated in 2005 by Sciabola and Morao [148] for examples of asymmetric reactions previously studied by Lipkowitz et al. [108], Kozlowski et al. [109] and Damen et al. [149]. GRIND uses MIF-based approaches to compute interaction fields that are encoded by alignment-independent variables with autocorrelation transform. In general, the predictive models generated with GRIND show comparable results to MIF alignment-dependent approaches [148]. Also, GRIND models are still interpretable, contrary to other models based on alignment-independent 3D descriptors, which apparently for this reason have not been widely used in the 3D-QSSR analysis. Other details on the approaches and descriptors used in QSSR can be found in the comprehensive review of Zahrt et al. [150]

Recently Asahara and Miyao [112] compared different 2D (ECFP6 and Mol2vec) and 3D descriptors (Dragon and MOE) to model the enantioselectivity of chiral Brønsted acid catalysts. The 3D descriptors were generated from the most stable conformers of reactants, products, and catalysts obtained with the force-field approach. As a result, the authors concluded that ECFP6 descriptors are found to be the best representation.

The above studies revealed three main drawbacks of existing 3D-QSSR approaches to the modelling of catalyst enantioselectivity: (i) selection of catalyst conformers, (ii) their alignment, and (iii) relevance of 3D descriptors with respect to the enantioselectivity problem. Inheriting previous conceptual progress in computational catalyst design, we have suggested a new protocol for the building of predictive models for catalyst enantioselectivity. In our approach, the catalysts are represented by an ensemble of conformers, encoded by new alignment-independent *pmapper* 3D descriptors which were successfully used in the modeling of ligands activity against 175 biological targets [12,14]. In order to consider an ensemble of catalyst conformers instead of a single selected conformer, the models were built using Multi-Instance machine Learning (MIL) algorithms. In the MIL approach, a molecule (catalyst) is presented by a bag of instances (set of conformers), and a label (experimental enantioselectivity) is associated with the bag (catalyst), but not with individual instances (conformers). In contrast to conventional single-instance learning where the object is represented by a single vector of descriptors, MIL determines a correlation between the bag descriptors and the labels. Thus, the application of MIL algorithms solves the problem of conformer selection and allows using all the generated catalyst conformers for the model building.

In this study, we demonstrate that the MIL-based 3D modelling approach can successfully be used to predict the enantioselectivity of homogeneous and phase-transfer reactions catalyzed by structurally different catalyst families. In both cases, the obtained models outperform traditional 2D models and previously reported 3D state-of-the-art approaches.

Datasets

Over the past two decades, numerous chiral organic catalysts have been designed for different types of reactions. Thus, BINOL (2,2'-Binaphthol) derivatives are popular catalysts in asymmetric synthesis because of their backbone flexibility, which enables the proper orientation of the reagents in 3D space. The *Cinchona* quaternary ammonium salts are extensively used in asymmetric phase-transfer catalysis (APTC) due to their capability to dissolve simultaneously in aqueous and organic liquids.

The catalyst enantioselectivity is often provided in enantiomeric excess (*ee* %) of the reaction which is defined as the difference between the amount of each enantiomer:

$$ee \% = \%R - \%S \text{ or } ee \% = \%S - \%R \quad (6)$$

The formula for calculating *ee* % depends on the type of the experimental datasets published in source papers. In this study, the *ee* % was converted to $\Delta\Delta G$ (kcal/mol) - a difference in free energy between competing reaction transition states leading to different enantiomers:

$$\Delta\Delta G = -RT \ln \frac{[R]}{[S]} = RT \ln \frac{100 - ee\%}{100 + ee\%} \quad (7)$$

To test our 3D modelling protocol, we selected two datasets on the chiral catalyst enantioselectivity - homogenous asymmetric nucleophilic addition and phase-transfer alkylation - used in previous modeling studies [110,111]. The phosphoric acid catalysts (PAC) dataset reported by Zahrt et al. [110] contains the enantioselectivity values for 43 catalysts used in 25 reactions of asymmetric addition of imine to thiol (Figure 16a) resulting in $43 \times 25 = 1075$ data points. Reported *ee* % (in favor of R enantiomer) ranged from -34 to 99 and for modelling were converted to $\Delta\Delta G$ (kcal/mol). A detailed description of the catalyst and reactant structures can be found in the original paper [110].

This dataset was divided into training and several test sets, as suggested by Zahrt et al. [110]. The training set consisted of 24 catalysts combined with 16 reactants resulting in $24 \times 16 = 384$ training reactions. Then, three test sets simulating different scenarios of the potential application of the models in real campaigns of catalyst design were prepared. The *reaction-out* test set containing 216 data points (24 training catalysts combined with 9 new reactions) is used to predict the

enantioselectivity of new reactions with known (presented in the training set) catalysts. The *catalyst-out* test set containing 304 data points (19 new catalysts combined with 16 training reactions) examines the potential of the model to predict the enantioselectivity of known reactions with new catalysts. The *both-out* test set represents the most challenging scenario where the model is used to predict the enantioselectivity of new reactants with new catalysts. This test set consists of 171 data points corresponding to combinations of 19 test catalysts and 9 test reactions.

Asymmetric phase transfer catalysis (APTC) enables reactions between reactants located in two immiscible phases with chiral catalysts to produce enantiopure substances. A classic example of APTC is the asymmetric synthesis of α -amino acids catalyzed by quaternary ammonium salts, particularly the alkylation of glycine-derived Schiff bases ($R^1R^2C=NR^3$) (Figure 16b).

We considered an example of asymmetric alkylation of α -amino acid derivatives catalyzed by cinchona alkaloid-based quaternary ammonium salts reported by Melville et al. [111] A catalysts library was generated by a variation of 13 substituents resulting in 88 catalysts. One substituent was presented only in a test set of 18 catalysts while the remaining 12 substituents were used to generate a training set of 70 catalysts. The reported *ee* ranged from 16 to 93 % (in favor of the S enantiomer).

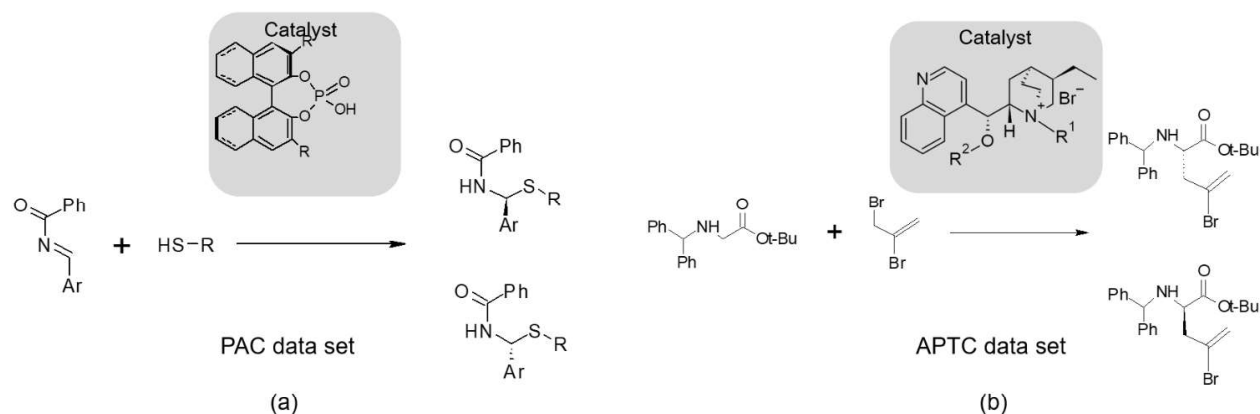


Figure 16. Examples of published reactions (datasets) considered for modeling in this study: (a) asymmetric addition of thiols to imines catalyzed by chiral phosphoric acid catalysts (PAC dataset) and (b) asymmetric alkylation of glycine-derived Schiff bases catalyzed by cinchona alkaloid-based ammonium salts (APTC dataset).

Computational details

Generation of 3D models

Reactant and catalyst descriptors

Catalyst conformers generation. Each catalyst was represented by an ensemble of conformers generated using the distance geometry algorithm implemented in the RDKit package [129]. If the RDKit algorithm failed to generate the conformers, we used a systematic conformer generator from the Open Babel package [130] and recalculated the full energies of conformers using RDKit. For each catalyst, we generated up to 50 conformers within an energy window of 50 kcal/mol.

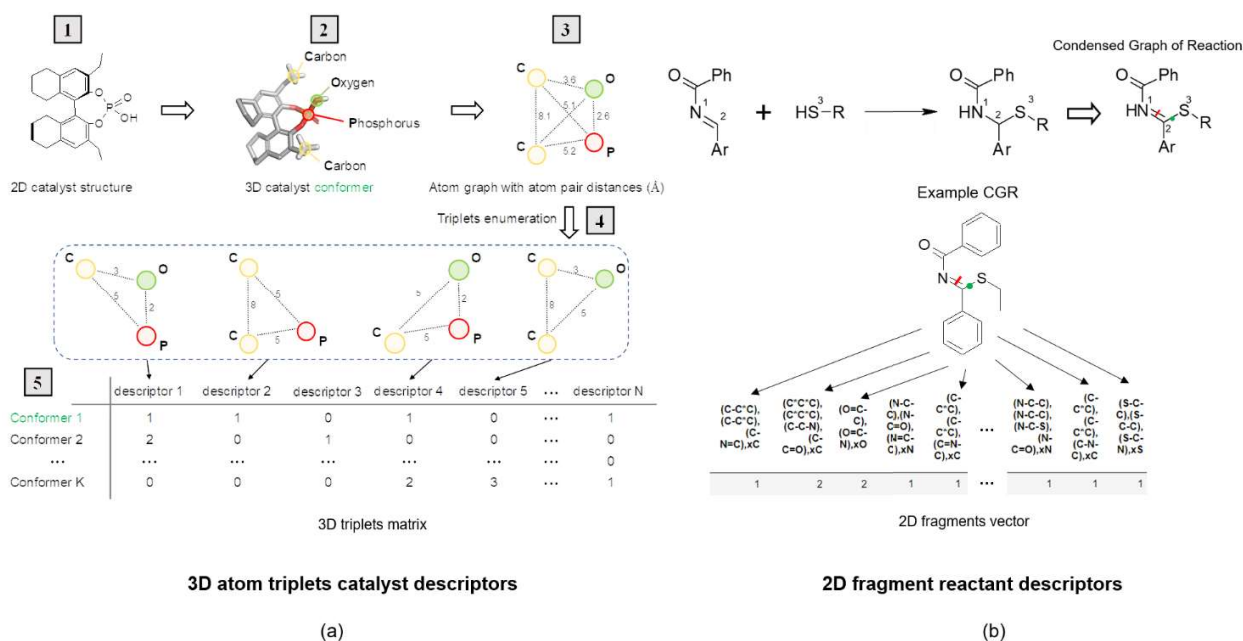


Figure 17. (a) Preparation of *pmapper* 3D descriptors for a given catalyst conformer: (1) given 2D catalyst structure; (2) generation of 3D catalyst conformer; (3) generation of a 3D fully connected graph of atoms (for demonstration, the graph of four atoms is chosen); (4) enumeration of all atom triplets; (5) counting of enumerated atom triplets in given conformer; (b) addition of thiols to imines and related Condensed Graph of Reaction (CGR). A CGR contains one created bond between the atoms S3 and C2 and one double bond transformed into a single one between the atoms N1 and C2.

Catalyst 3D descriptors. The generated catalyst conformers were encoded by *pmapper* descriptors [135], representing 3D pharmacophore quadruplets [12,14]. The pharmacophore labels (H-donor, H-acceptor, hydrophobic, positive or negative charges) are assigned to atoms, functional groups, or rings. Rings are characterized by either hydrophobic or aromatic features. The application of pharmacophore quadruplets enables encoding the stereoconfiguration of a molecule, which

guarantees that two enantiomers of a molecule have two different descriptor vectors. In our previous paper [15], we demonstrated that a combination of 3D pharmacophore quadruplets and MIL generates accurate models for the PAC dataset. However, in this work, instead of pharmacophore features, we used quadruplets and triplets of individual atoms (and centers of 5- and 6-membered aromatic rings) - atom quadruplets and atom triplets. Preliminary experiments (which will be discussed later) revealed that atom triplets significantly reduce the number of descriptors, and demonstrate even better performance than atom quadruplets. However, if a dataset contains catalysts in both R and S configurations - the application of atom quadruplets is mandatory to distinguish the two enantiomers. The atom triplets are applicable in this study because all catalysts in the considered datasets have the same stereoconfiguration.

The atom triplets are specified by (1) the list of the individual atoms (C, N, O, S, P, F, Cl, Br, I) or 5-membered and 6-membered aromatic ring and (2) the distances between atoms and/or center of rings in a triplet. The list of encoded atoms can be customized depending on the task. To enable fuzzy matching of atom triplets and identify similar ones, the distances between atoms are binned with the step of 1 Å (Figure 17a). Then the number of occurrences of each unique atom triplet is counted for each conformer, resulting in an integer descriptor matrix (Figure 17a).

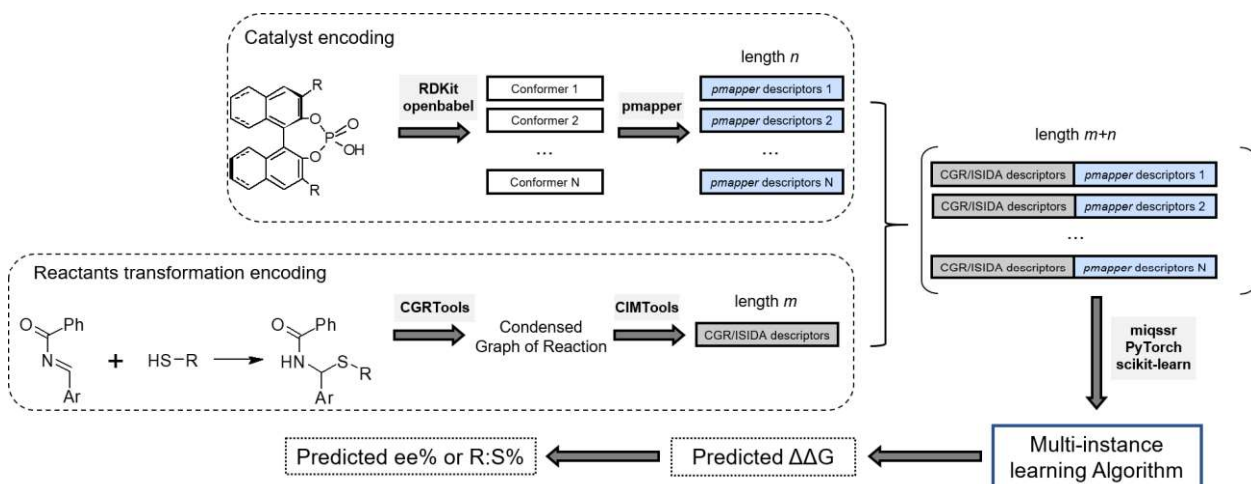


Figure 18. Preparation of descriptors encoding a combination of reactants and corresponding catalysts in a 3D modeling approach. A reactant transformation is encoded by m CGR/ISIDA fragment descriptors. A catalyst is represented by its N conformers, each encoded by n of 3D *pmapper* descriptors. Concatenation of m 2D reactant descriptors and n 3D catalyst descriptors results in the set of vectors of $(m + n)$ size. The Python 3 libraries used in the modeling workflow are indicated in bold near the arrows.

Reactant 2D descriptors calculation. Each structural transformation of reactants is transformed into a Condensed Graph of Reaction (CGR)[151] with a *CGRtools* package [152]. CGR

considers a chemical reaction as one single pseudo molecule (Figure 17b) and contains conventional chemical bonds (e.g. single, double, triple, aromatic, etc.) and so-called “dynamic” bonds describing chemical transformations, i.e. breaking or forming a bond or changing bond order. Obtained CGRs then are processed with In Silico Design and Data Analysis (ISIDA) tool to calculate 2D fragment descriptors [153]. ISIDA fragment descriptors count the occurrence of particular subgraphs (structure fragments) in given CGRs. ISIDA provides several strategies for molecule fragmentation. In this study, we used atom-centered subgraphs (atoms with first, second, etc. coordination spheres) where the radius varied from 2 to 5 atoms.

Reaction profile descriptors. Vectors of 2D fragment descriptors for reactions and 3D atom triplets for catalysts were then concatenated to form reaction profile descriptor vectors (Figure 18). If the dataset contained a single reactant transformation, there is no concatenation of catalyst and reactant descriptors. Figure 18 shows the general scheme of our 3D modelling protocol.

Multi-instance learning algorithms

For the MIL algorithms benchmark, we used a PAC dataset, which was divided into 25 subsets according to the number of reactant transformations. Each subset contained 43 catalysts with experimental $\Delta\Delta G$ measured in a given reactant transformation. Middle Absolute Error (MAE) of $\Delta\Delta G$ predictions was evaluated in a 5-fold cross-validation repeated 5 times (5 $\times$ 5-CV). The comparison results show that the *Instance-Wrapper* algorithm considerably outperforms other algorithms, including the most complex *Bag-AttentionNet* one.

The basic machine learning algorithm in *Instance-Wrapper* was a fully connected neural network with three hidden layers of 256, 128, and 64 neurons and a ReLU activation function. The optimized hyperparameters were weight decay (0.0001, 0.001, 0.01, 0.1) and learning rate (0.001 or 0.01). The maximum number of learning epochs was 1000.

Generation of 2D models

As an alternative to our 3D approach, we also considered the 2D modeling approach where the reactants and catalyst structures are encoded by different fingerprint and fragment 2D descriptors. The following fingerprints were generated using the RDKit library: Atom-Pairs (1024 bits) [154], Avalon (1024 bits)[155], and Morgan fingerprints of radius 2 (1024 bits) [156]. Fragment ISIDA [153] and CircuS [157] (**Circular Substructures**) descriptors can be calculated with different fragmentation strategies. For ISIDA, both atom-centered and linear fragments were used. CircuS are

similar to ISIDA atom-centered fragments, but explicitly consider encountered branching or cyclical structures, which makes them more efficient for catalyst structures enriched with cyclical groups and reduces the noise in the training data.

For a PAC dataset containing multiple reactant transformations, there were two encoding strategies: (a) reactant transformations were converted to CGR and then encoded by ISIDA or CircuS (fingerprints tools are unable to process CGR) fragment descriptors (Imine/Thiol CGR, Table 2) or (b) imine and thiol were encoded by fingerprints or fragment descriptors and then concatenated to a single descriptor vector (Imine/Thiol concatenation, Table 2). Then the resulting reactant transformation vectors were concatenated with fingerprint or fragment descriptor vectors of the catalysts.

Fragment-based descriptors can be calculated using different strategies and fragment lengths, generating multiple sets of descriptors. In order not to be biased towards specific descriptor sets, we applied a consensus method to calculate the final predictions. First, for each descriptor type (ISIDA, CircuS, or fingerprints), we selected models with $R^2_{\text{Train}} > 0.7$ to discard descriptor sets that poorly describe the training set. Then the predictions of the filtered models for the test set were averaged to obtain final consensus predictions of enantioselectivity. For model training, the same fully connected neural network was used as in the *Instance-Wrapper* algorithm in multi-instance models.

The following metrics were used to assess the performance of the models: Root-Mean Squared Error (RMSE), Mean Absolute Error (MAE), determination coefficient (R^2), Spearman correlation coefficient measuring the correlation between predicted and experimental catalyst ranks (ranking accuracy, RA).

Results and Discussion

Using the described datasets and modelling protocols, various 2D and 3D models for enantioselectivity prediction were generated. The 3D single-conformer model was built on the lowest-energy catalyst conformers, while 3D multi-conformer model included all the generated conformers.

Benchmarking of 2D/3D descriptors and MIL algorithms

We were interested in how effectively existing 2D and 3D descriptors encode catalyst structure isolating the influence of reactants transformation descriptors. For the comparison, we used the PAC dataset divided into 25 subsets. Each subset included a particular chemical transformation in presence of one of 43 catalysts. We choose ISIDA [153] and CircuS [157] fragment descriptors,

2D fingerprints, and 3D descriptors available in RDKit, as well as our 3D atom triplets and quadruplets descriptors. A set of 3D RDKit descriptors included Radial Distribution Function (RDF) descriptors, Molecule Representation of Structure-based on Electron diffraction (MoRSE) descriptors, Weighted Holistic Invariant Molecular (WHIM) descriptors, GETAWAY and AutoCorr3D descriptors. We compared 3D descriptors in a multi-instance setting, i.e., the considered *pmapper* and *RDKit* 3D descriptors were generated for multiple conformers. The Instance-Wraper MIL algorithm was used as a machine-learning method to build 3D models. In the case of 2D descriptors, the MIL bag contained only one instance.

The performance of 2D and 3D models (MAE of $\Delta\Delta G$ predictions, kcal/mol) was evaluated in a 5-fold cross-validation repeated 5 times (5 $\times$ 5-CV). As a result, 25 MAE values of predicted $\Delta\Delta G$ for 43 catalysts were collected for each type of descriptor for each reactant transformation (Figure 19).

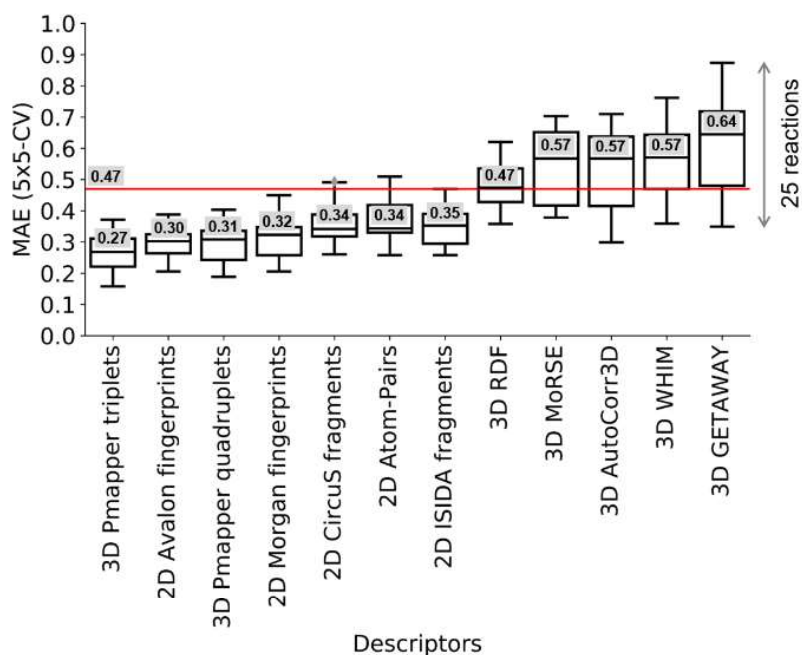


Figure 19. Comparison of different classes of 2D and 3D descriptors available online. Each catalyst was encoded by 2D fingerprint or fragment descriptors, 3D RDKit descriptors, or *pmapper* 3D descriptors. 3D descriptors were calculated for multiple catalyst conformers (i.e. there is a set of 3D multi-conformer models). Each box contains a cross-validated MAE of $\Delta\Delta G$ predictions for 43 catalysts obtained from 25 models (25 reactant transformations). The red horizontal line shows the accuracy of the default model, which constantly predicts $\Delta\Delta G$ as the average experimental $\Delta\Delta G$ across all catalysts.

The comparison results show (Figure 19) that only 2D fingerprints and fragment descriptors, as well as *pmapper* 3D quadruplets and triplets descriptors, generate predictive models (better than the baseline null model - the model that predicts enantioselectivity always as an average value of

the training experimental enantioselectivities - with MAE = 0.47 kcal/mol) for all 25 reactant transformations, while 3D *RDKit* descriptors fail to predict the catalyst enantioselectivity for the most reactant transformations (Figure 19). 3D atom triplets and quadruplets demonstrate similar performance (median MAE_{CV} = 0.27 vs. MAE_{CV} = 0.31 kcal/mol), but the use of atom triplets radically reduces the number of catalyst descriptors compared to atom quadruplets from 42824 to 2886 descriptors.

Generally, *pMapper* 3D descriptors generated from atom triplets perform slightly better (median MAE_{CV} = 0.27 kcal/mol) than all other 2D descriptors (median MAE_{CV} = 0.30-0.35 kcal/mol).

Thus, 3D *RDKit* descriptors were found unsuitable for modelling the catalyst enantioselectivity and are inferior even to 2D descriptors. The proposed 3D atom triplets demonstrated the best performance.

Comparison of multi-instance learning algorithms. We compared the five MIL algorithms[12]. The comparison was performed using the same setting as the benchmark of descriptors mentioned in Figure 19. The median MAE (in kcal/mol) over 25 reactions (5×5-CV) are as follows: *Instance-Wrapper* (0.28 kcal/mol), *Bag-Wrapper* (0.31 kcal/mol), *Instance-Net* (0.31 kcal/mol), *Bag-Net* (0.32 kcal/mol) and *BagAttention-Net* (0.35 kcal/mol). Based on the obtained results, *Instance-Wrapper* was chosen as the main algorithm for further experiments.

Asymmetric addition of thiols to imines

We compared the performance of our 2D and 3D models with the previously reported results. Sandfort et al. [113] published a structure-based machine learning platform, where reactants and catalysts were encoded by multiple fingerprint features (MFFs) resulting from the concatenation of 24 fingerprints sets calculated with *RDKit*. Zahrt's conformer-dependent 3D approach [110] is based on the ASO descriptors, accumulating steric information from an ensemble of catalyst conformers. Asahara and Miyao [112] benchmarked 2D (ECFP6 and Mol2vec) and 3D (Dragon and MOE) single-conformer descriptors.

Our models. In the *reaction-out test set*, all generated 2D and 3D models demonstrated good results. The 2D models accurately predict enantioselectivity with MAE = 0.14-0.18 kcal/mol. The 3D single-conformer model also provides accurate predictions with MAE = 0.21 kcal/mol, while the inclusion of multiple conformers in the 3D multi-conformer model considerably increases the prediction accuracy up to MAE = 0.13 kcal/mol. In contrast to the *reaction-out test set*, in the *catalyst-out test set* the 3D multi-conformer model performs significantly better (MAE = 0.22 kcal/mol) than the 3D single-conformer model (0.38 kcal/mol) and 2D models (0.26-0.36 kcal/mol). Similar to the *catalyst-out test set*, in the *both-out test set* the 3D multi-conformer model

is significantly more accurate (MAE = 0.21 kcal/mol) than the 3D single-conformer model (0.48 kcal/mol) and 2D models (0.28-0.34 kcal/mol).

Table 2. Mean Absolute Error (MAE, kcal/mol) of $\Delta\Delta G$ predictions obtained for test sets generated from the phosphoric acid catalysts (PAC) dataset. ^a 2D modelling approach published by Sandfort et al. [113], ^b 2D and 3D models published by Asahara and Miyao [112], and ^c 3D conformer-dependent approach published by Zahrt et al. [110].

Reactants representation	Model (descriptors)	Reaction-out	Catalyst-out	Both-out
Imine/Thiol concatenation	2D model (Morgan fingerprints)	0.18	0.29	0.33
	2D model (Avalon fingerprints)	0.15	0.26	0.28
	2D model (Atom-Pairs fingerprints)	0.16	0.36	0.33
	2D model (ISIDA fragments)	0.14	0.27	0.28
	2D model (CircuS fragments)	0.14	0.31	0.33
Imine/Thiol CGR	2D model (ISIDA fragments)	0.15	0.27	0.30
	2D model (CircuS fragments)	0.14	0.32	0.34
	3D single-conformer model (Atom triplets)	0.21	0.38	0.48
	3D multi-conformer model (Atom triplets)	0.13	0.22	0.21
Alternative approaches	2D Sandfort's model (MFFs fingerprints) ^a	0.14	0.25	0.28
	2D model (Mol2vec) ^b	0.13	0.34	0.40
	2D model (ECFP6) ^b	0.14	0.22	0.21
	3D single-conformer model (Dragon) ^b	0.14	0.42	0.47
	3D single-conformer model (MOE) ^b	0.15	0.48	0.55
	3D Zahrt's conformer-dependent model (ASO descriptors) ^c	0.16	0.21	0.24

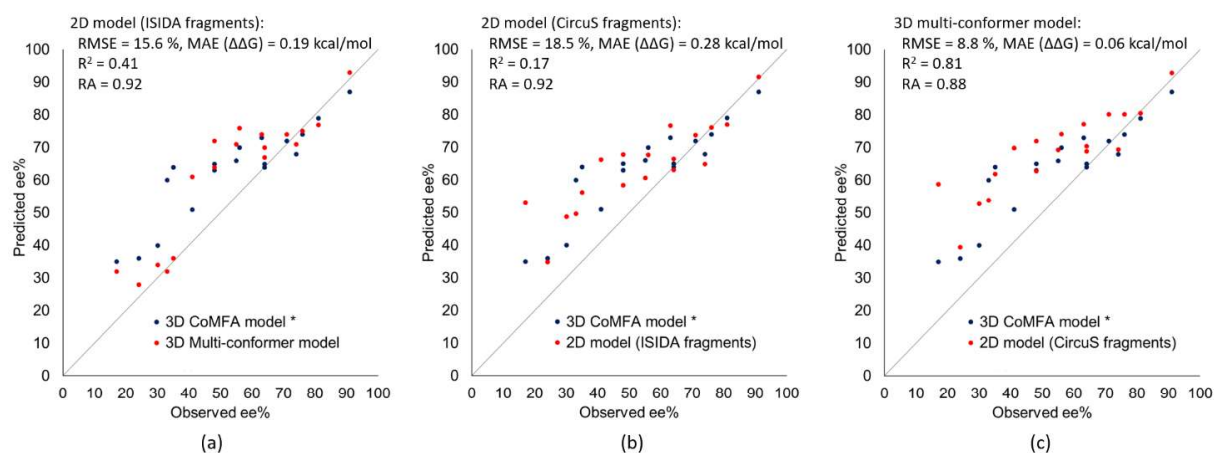


Figure 20. Observed and predicted *ee* % for 18 test catalysts from the APTC dataset comparing the performance of the 3D-CoMFA model by Melville et al [111] with: (a). 2D model (ISIDA fragments), (b) model (CircuS fragments), and (a) 3D multi-conformer model (atom triplets).

Alternative approaches. The 3D single-conformer model based on 3D atom triplets (MAE = 0.48 kcal/mol) and Miyao's 3D models based on Dragon (0.47 kcal/mol) and MOE (0.55 kcal/mol) single-conformer descriptors displayed low performance on the *both-out test set*, which

demonstrates that a single conformer was not sufficient to generate accurate models irrespective of the type of 3D descriptors (Table 2). In contrast, 3D multi-conformer model based on atom triplets was significantly more accurate (MAE = 0.21 kcal/mol) and perform slightly better than 3D conformer-dependent approach reported by Zahrt et al [110] (0.24 kcal/mol) (Table 2). Interestingly that Miyao's 2D model based on ECFP6 descriptors achieved high accuracy (MAE = 0.21 kcal/mol) similar to our 3D multi-conformer model.

To summarize, for the case of asymmetric addition of thiols to imines, the 3D multi-conformer model outperforms the 3D single-conformer models, especially in the prediction of enantioselectivity for new test catalysts, which proves the importance of accounting for conformational flexibility. We suppose that the difference in the performance of 3D single-conformer and 3D multi-conformer models will increase with the flexibility of modeled catalysts. The 3D multi-conformer model outperforms the 2D models, generated with popular fingerprints and fragment descriptors, which highlights the importance of 3D information in enantioselectivity modelling.

Asymmetric phase transfer catalysis

The dataset of asymmetric alkylation (APTC dataset) was divided into 70 training and 18 test catalysts as described by Melville et al [111]. To build the models, the original enantioselectivities were converted to $\Delta\Delta G$, then the predictions on the test set were converted to *ee* % to be compared with the predictions of the competing approach. Melville and co-workers proposed a 3D CoMFA-based approach based on minimal energy catalyst conformers and reported RMSE of *ee* predictions on 18 test catalysts as 13.4 %. Our 3D single-conformer model performed considerably worse (RMSE = 18%). Consideration of the ensemble of conformers in the 3D multi-conformer model significantly reduced RMSE to 8.8% (Figure 20). The substantial difference in the performances of 3D single-conformer and 3D multi-conformer models (RMSE of 18.0% vs. 8.8%), can be explained by the high conformation flexibility of the given catalysts – the average number of rotatable bonds in the dataset was 10. The 2D models built on ISIDA and CircuS descriptors demonstrated poor performance with RMSE of 15.6 and 18.5 %, respectively. This example demonstrated that our modelling protocol without any modifications or manual adjustment can be applied to catalysts with a new scaffold.

In a computational screening of candidate catalysts, the predictive model should effectively identify potential highly selective catalysts, i.e. the model should rank them higher than the other candidates. To quantify this characteristic of the model, we also calculated ranking accuracy (RA) which is the coefficient of correlation between predicted and experimental catalyst ranks (Spearman correlation coefficient). Figure 20 shows that despite large prediction error (RMSE) the 2D

models achieve high RA > 0.90, i.e. they well capture the general trend in enantioselectivity variation (Figure 20). The high absolute accuracy of 3D multi-conformer models in comparison to other approaches is achieved by more accurate predictions for low-selective catalysts (Figure 20).

Enantioselectivity prediction beyond the training set

A new round of catalyst screening is expected to reveal more enantioselective catalysts. In this context, it is desirable to prevent under-predictions where the predicted enantioselectivity is significantly lower than the actual value. Incorrect behavior of the model in these examples can cause underestimation of most perspective catalysts, which may not be sampled for experimental testing in the next rounds of screening. Thus, the predictive model should be specially configured to avoid under-predictions ($y^{pred} < y^{obs}$) of enantioselectivity. To increase the prediction accuracy for highly selective catalysts, we propose to train the model with a special quantile loss function:

$$L = \max[q \times (y^{pred} - y^{obs}), (q - 1) \times (y^{pred} - y^{obs})] \quad (8)$$

Quantile loss function (3) asymmetrically penalizes over-predictions ($y^{pred} > y^{obs}$) and under-predictions ($y^{pred} < y^{obs}$). For q equal to 0.5, under-predictions and over-predictions are penalized equally. The lower the value of $q < 0.5$, the more under-predictions are penalized compared to over-predictions. In this study, q was fixed at 0.1 which means that over-prediction is penalized by a factor of 0.1, and under-predictions by a factor of 0.9, and, thus, the model tries to avoid under-predictions.

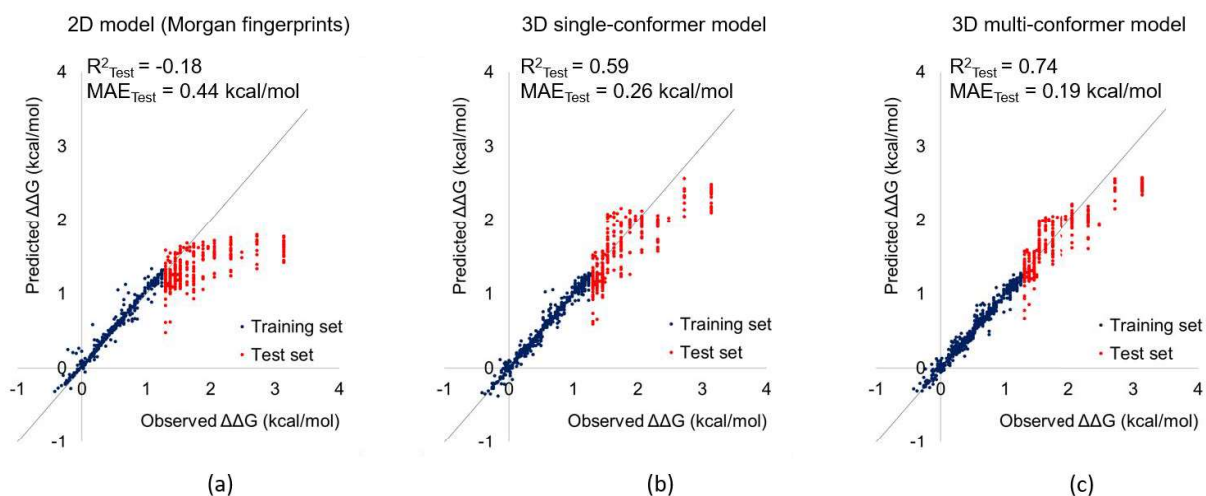


Figure 21. Predicted and observed catalyst enantioselectivity ($\Delta\Delta G$, kcal/mol) for (a) 2D model, (b) 3D single-conformer model, and (c) 3D multi-conformer model. The training set included reactions with $ee < 80\%$ and the test set with $ee \geq 80\%$. 2D and 3D models were trained with quantile loss.

To examine the potential of the models to predict enantioselectivity values beyond the training set, we followed the validation strategy proposed by Denmark's group in their original paper [110]. The PAC dataset on 1075 reactions was divided into a training set of reactions with *ee* below 80% (718 reactions) and a test set of highly selective reactions with *ee* above 80% (357 reactions). Then we built 2D and 3D models with classic mean squared error loss (MSE) and quantile loss.

All 2D models (ISIDA, CircuS, and RDKit fingerprints) built with MSE loss fails to predict enantioselectivity beyond the training set ($R^2_{\text{Test}} < 0$), while the 3D single-conformer model ($R^2_{\text{Test}} = 0.36$) and 3D multi-conformer model ($R^2_{\text{Test}} = 0.44$) performs significantly better. Model training with the quantile loss function considerably improved 3D single-conformer ($R^2_{\text{Test}} = 0.59$) and 3D multi-conformer models ($R^2_{\text{Test}} = 0.74$). The 2D models built with the quantile loss function are still worse than the null model ($R^2_{\text{Test}} < 0$) (Figure 21).

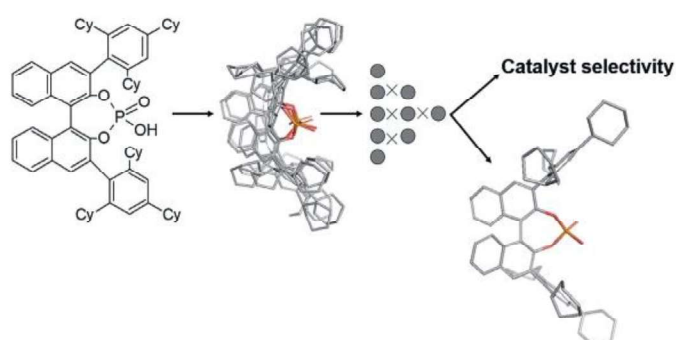
Thus, the 3D multi-conformer model better predicts catalyst enantioselectivity beyond the training set than 2D models. Furthermore, the proposed 3D multi-conformer model trained with the quantile loss is better ($\text{MAE}_{\text{Test}} = 0.19$ kcal/mol) compared to the results by Zahrt et al. approach ($\text{MAE}_{\text{Test}} = 0.33$ kcal/mol) [110].

Conclusion

In this study, multi-instance machine learning in combination with *pmapper* 3D descriptors was applied to model and predict the enantioselectivity of chiral catalysts in asymmetric addition of thiols to imines (BINOL-derived catalysts) and alkylation of glycine imine (cinchona alkaloid-based ammonium salts). The catalysts were represented either by the lowest-energy conformer (3D single-conformer model) or by multiple conformers (3D multi-conformer model). The catalyst conformers were encoded by *pmapper* 3D descriptors, which in this study are configured to count particular atom triplets and do not require alignment of the conformers. The developed 3D models were compared with traditional 2D models built with popular fingerprint and fragment descriptors and the state-of-the-art 3D approaches published in chemoinformatics papers.

In general, the inclusion of multiple catalyst conformers in the modeling process significantly increases the accuracy of enantioselectivity predictions in comparison with single-conformer modeling. The comparison analysis showed that the 3D atom triplets outperform other RDKit alignment-independent descriptors and 2D RDKit fingerprints, ISIDA, and CircuS fragment descriptors. The generated 3D multi-conformer models perform the same or better than published state-of-the-art 3D approaches. This work demonstrates that the developed 3D modelling protocol does not require the selection and alignment of conformers and applies to two different

catalyst systems (BINOL derivatives and ammonium salts), showing the best performance. The proposed *pmapper* 3D descriptors are customizable, i.e. one can manually specify the atom groups or relevant 3D patterns that are responsible for observed enantioselectivity.



Multi-Instance Learning Approach to Predictive Modeling of Catalysts Enantioselectivity

D. Zankov, P. Polishchuk, T. Madzhidov, A. Varnek

Multi-Instance Learning Approach to Predictive Modeling of Catalysts Enantioselectivity

D. Zankov^{a,c}P. Polishchuk^bT. Madzhidov^cA. Varnek^{a,d}

^a Laboratory of Chemoinformatics, University of Strasbourg, 4, B. Pascal, 67081 Strasbourg, France
varnek@unistra.fr

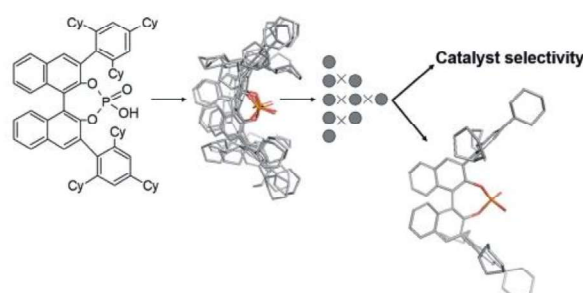
^b Institute of Molecular and Translational Medicine, Palacký University, Hnevotinska 5, 77900 Olomouc, Czech Republic

^c Laboratory of Chemoinformatics and Molecular Modeling, Kazan Federal University, Kremlyovskaya 18, 420008 Kazan, Russia

^d Institute for Chemical Reaction Design and Discovery (WPI-ICReDD), Hokkaido University, Kita 21 Nishi 10, Kita-ku, 001-0021 Sapporo, Japan

Published as part of the Cluster

Machine Learning and Artificial Intelligence in Chemical Synthesis and Catalysis



Received: 10.06.2021

Accepted after revision: 16.07.2021

Published online:

DOI: 10.1055/ja-1553-0427; Art ID: st-2021-b0229-c

Abstract Here, we report an application of the multi-instance learning approach to predictive modeling of enantioselectivity of chiral catalysts. Catalysts were represented by ensembles of conformations encoded by the *pmapper* physicochemical descriptors capturing stereoconfiguration of the molecule. Each catalyzed chemical reaction was transformed to a condensed graph of reaction for which ISIDA fragment descriptors were generated. This approach does not require any conformations' alignment and can potentially be used for a diverse set of catalysts bearing different scaffolds. Its efficiency has been demonstrated in predicting the selectivity of BINOL-derived phosphoric acid catalysts in asymmetric thiol addition to *N* acylimines and benchmarked with previously reported models.

Key words asymmetric catalysis, chemoinformatics, machine learning, QSSR

Enantioselective catalysis is widely used for the synthesis of enantiomerically pure compounds. Design of perspective catalysts is traditionally conducted by iterative modification of the molecular structure aiming to increase the enantioselectivity of a reaction product. Predictive chemoinformatics models may guide chemists toward the most promising catalysts before their synthesis and experimental testing, reducing in such a way both human and material resources.^{1,2} Such models built on molecular descriptors encoding catalyst and reactants structures in combination with machine learning methods are used for hunting potent catalysts in virtual screening experiments.

The early models of enantioselectivity by Kozłowski³ and by Lipkowitz⁴ considered only one conformation per catalyst. Later on, Melville et al.⁵ suggested considering several conformations along the molecular dynamics trajectory

within the CoMFA approach. Another strategy to account for multiple conformations was suggested by the Denmark group who invented average steric occupancy (ASO)⁶ and average electronic indicator field (AEIF)⁷ descriptors. The aligned conformations were placed in the rectangular box followed by calculations of either relative occupancy by the catalyst atoms of each node of the rectangular grid (ASO) or normalized atomic charge of atoms overlapping with the grid nodes (AEIF). These descriptors provided multidimensional information for the CoMFA approach which significantly improved the performance of the enantioselectivity models.^{6,7} Yamaguchi and Sodeoka successfully applied molecular field analysis to some catalyst–substrate complexes in order to design new highly effective catalysts.⁸ Recently, Xu et al. reported spherical projection descriptors of molecular stereostructure (SPMS),⁹ which allows precise representation of the molecular van der Waals surface. These descriptors were calculated for each conformation of catalyst and substrate and then used in a convolutional neural network to train an enantioselectivity model on the dataset reported in reference.⁶

Here, we report an alternative approach – multi-instance machine learning¹⁰ algorithm (Figure 1) accounting for multiple molecular conformations in structure–activity tasks. This method in combination with original molecular and reaction descriptors has been applied to the dataset on the selectivity of phosphoric acid catalysts in asymmetric addition of thiols to imines (Figure 2) reported by Zahrt et al.⁶ This dataset contains selectivity of 43 catalysts systematically measured in 25 reactions, which results in 1075 data points. The catalyst selectivities were estimated by enantiomeric excess (*ee* %) ranged from –43 to 99. For the model development, the *ee* % values were converted into $\Delta\Delta G$ (kcal/mol), a free-energy difference between compet-

ing transition states leading to different enantiomers. A detailed description of the catalyst and reactant structures is given in the original paper.⁶

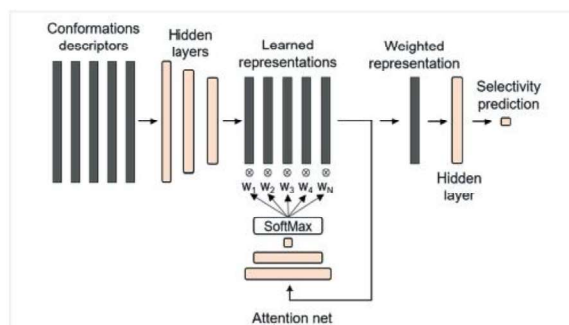


Figure 1 Multi-instance neural network with the attention mechanism. The network receives an ensemble of descriptor vectors of all considered conformations of a given molecule as an input in order to predict both enantiomeric selectivity and the weights w_i measuring the importance of each conformation.

The multi-instance learning algorithm represents a neural network with an attention mechanism, which prioritizes few conformations responsible for the observed activity and ignores the irrelevant conformations introducing noise in the modeling process (Figure 1). Namely, the attention mechanism assigns to each conformation a weight from 0 to 1, determining its importance in terms of predicting catalyst selectivity. The sum of all attention weights equals 1. In the learning process, each instance (conformation descriptor vector) runs through three fully connected layers with 256, 128, and 64 hidden neurons, respectively. Then the learned instance representations inputs to the attention network with 64 hidden neurons, in which the number of output neurons is equal to the number of input instances. The output neurons are followed by a *Softmax* unit which calculates attention weights for each instance. The weighted averaging of learned instance representations results in the embedding vector, which, in turn, is used to predict reaction selectivity (Figure 1). In the multi-instance approach, a catalyst molecule is represented by a bag of instances (*i.e.*, a set of conformations) to which a label (a selectivity value) is assigned. The multi-conformation models were compared with single-conformation models constructed for the lowest-energy catalyst conformation. Each catalyst was represented by a set of its conformations, which were encoded by *pmapper* descriptors.¹¹ These descriptors were developed in our group and probed in predicting the biological activity of molecules.¹² They do not require alignment of conformations and can potentially be applied to model catalysts with diverse scaffolds. Also, *pmapper* descriptors are sensitive to the stereoconfiguration of the molecule, *i.e.*, enantiomers are described by different descriptor vectors.

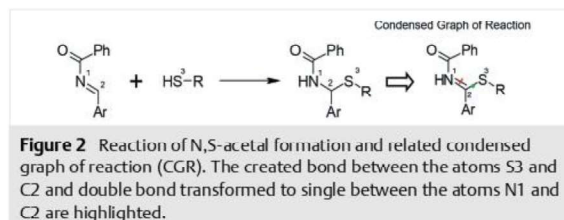


Figure 2 Reaction of N,S-acetal formation and related condensed graph of reaction (CGR). The created bond between the atoms S3 and C2 and double bond transformed to single between the atoms N1 and C2 are highlighted.

Each reaction was transformed to a condensed graph of reaction (CGR)¹³ with a *CGRtools* package.¹⁴ CGR is a single graph, which encodes an ensemble of reactants and products as is shown in Figure 2. CGR results from the superposition of the atoms of products and reactants having the same numbers. It contains both conventional chemical bonds (single, double, triple, aromatic, etc.) and so-called 'dynamic' bonds describing chemical transformations, *i.e.*, breaking or forming a bond or changing bond order. Given CGRs were encoded by (in silico design and data analysis (ISIDA) fragment descriptors,¹⁵ counting the occurrence of particular subgraphs (structural fragments) of different topologies and sizes. In this study, atom-centered subgraphs containing a given atom with the atoms and bonds of its *n* coordination spheres ($n = 1-4$) were used.

For each catalyst, up to 50 conformations within a 10 kcal/mol energy window have been generated using the distance geometry algorithm implemented in RDKit.¹⁶ The conformations with RMSD values below 0.5 Å with respect to selected conformations were removed in order to reduce redundancy. Then, selected conformations were encoded by a vector of *pmapper*¹¹ descriptors. Each conformation is represented by an ensemble of physicochemical features assigned to atoms, functional groups, or rings: H-donor, H-acceptor, or hydrophobic, or positively or negatively charged. Rings are characterized by either hydrophobic or aromatic features. All possible combinations of features quadruplets are enumerated. Each quadruplet is encoded by a canonical signature, which contains information about comprising features, the distance between them, and stereoconfiguration. To enable fuzzy matching of quadruplets to identify similar ones, the distances between features are binned with the step of 1 Å. Each unique quadruplet is considered as a descriptor whereas its count is a descriptor value (Figure 3) ■■sentence OK?■■. More details about descriptor generation are reported in our previous paper.¹¹ Vectors of 2D fragment reaction descriptors and 3D physicochemical quadruplets were then concatenated to form a combined reaction/catalyst descriptor vector (Figure 4). For the sake of comparison, some models were built using RD-Kit 3D descriptors, which were used in reaction/catalyst descriptor vector instead of *pmapper* descriptors.

The single- and multi-conformation models (SCM and MCM, respectively) were built on the training set of 384 data points resulted from a combination of 24 catalysts combined with 16 reactions. The models were validated on

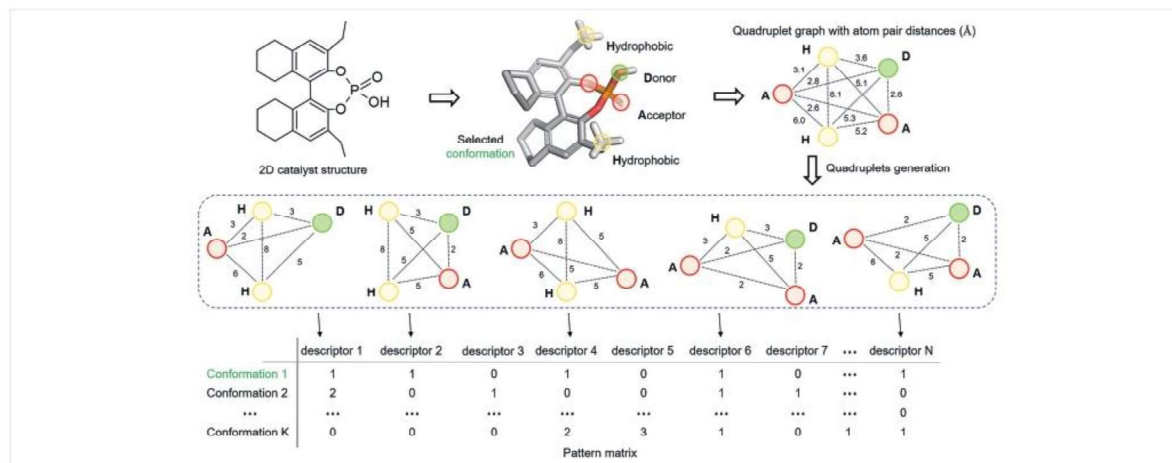


Figure 3 Workflow of preparation of *pmapper* descriptors for a given conformation. The physicochemical labels are assigned to particular atoms or functional groups followed by the preparation of a 3D fully connected graph for which an ensemble of quadruplets is generated.

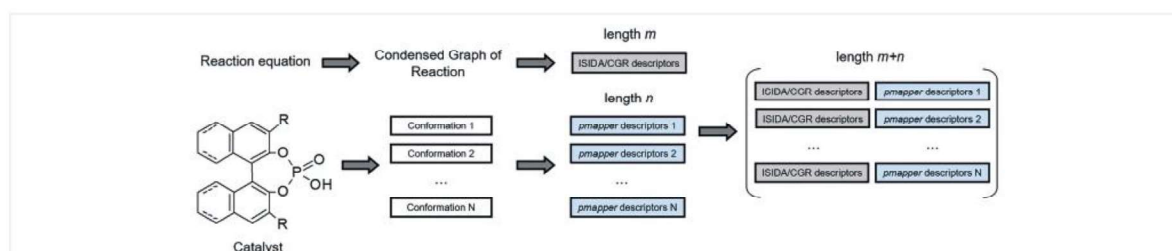


Figure 4 Preparation of descriptors encoding reaction/catalyst combinations. A chemical reaction is encoded by *mISIDA/CGR* descriptors calculated for the condensed graph of reaction. A catalyst is represented by its N conformations, each encoded by n *pmapper* descriptors. Concatenation of reaction and catalyst descriptors results in the vector of $(m+n)$ size.

three test sets selected according to different scenarios: (a) new reactions with known catalysts, (b) known reactions with new catalysts, and (c) new reactions with new catalysts. Thus, Test set 1 contained 216 instances resulted from a combination of 24 catalysts from the training set with 9 new reactions, Test set 2 included 314 instances (19 new catalysts/16 training reactions), and Test set 3 contained 171 instances (19 new catalysts/9 new reactions).

Performances of single-conformation and multi-conformation models (mean absolute error, MAE) in comparison with those of the model by Zahrt et al.⁶ are given in Figure 5. One may see that for Test set 1, both SCM and MCM perform similarly to Zahrt's model, whereas for Test sets 2 and 3, performances of MCM and Zahrt's models are similar whereas SCM performs much worse.

These results demonstrate the importance of accounting for all representative catalyst conformations in predictive modeling. We expect that the difference in the performance of single- and multi-conformation models would increase when more flexible catalysts are considered. This

emphasizes the importance of the choice of 3D descriptors able to capture relationships between the structure of molecules and their catalytic activity. It seems that our *pmapper* descriptors are well suited for this task. Our benchmark-

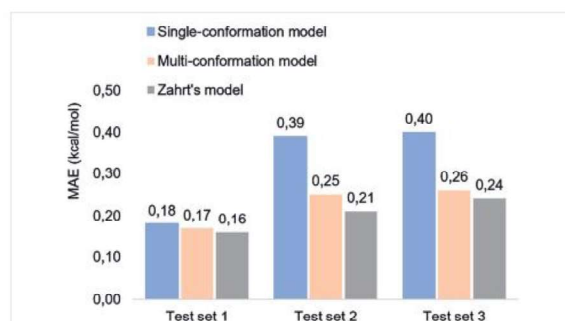


Figure 5 Mean absolute error (MAE, kcal/mol) obtained for Test sets 1–3.

ing studies demonstrated that multi-conformation models built on *pMapper* descriptors outperformed those based on 3D descriptors calculated with RDKit (Figure 6).

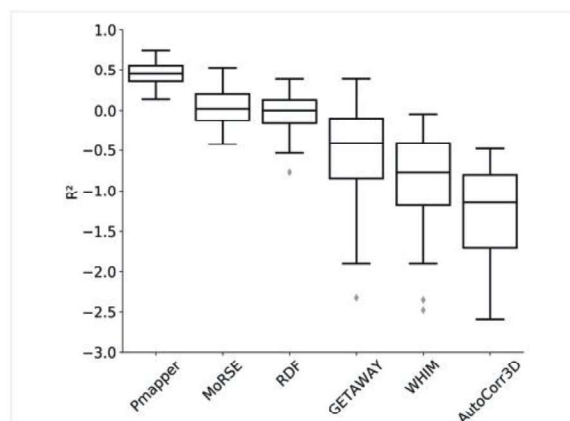


Figure 6 Performance of models based on different classes of 3D descriptors in predicting BINOL-derived catalysts selectivity in 25 reactions. Each box contains a cross-validated determination coefficient R^2 for 25 models (one model per reaction).

To summarize, our approach combining original ISI-DA/CGR descriptors for chemical reactions, 3D physico-chemical *pMapper* descriptors for catalysts, and multi-instance machine learning method performs similarly to the state-of-the-art model recently reported by Denmark's group. Unlike conventional 3D or 4D QSAR techniques, our approach is more general because it doesn't require any conformations or atomic alignment and potentially allows training a model on structurally diverse datasets combining catalysts with different scaffolds.¹⁷

Conflict of Interest

The authors declare no conflict of interest.

Funding Information

DZ thanks the French Embassy in Russia for the PhD fellowship. TM thanks Russian Science Foundation (Grant No. 19-73-10137) for the support.

References

- (1) Yang, W.; Fidelis, T. T.; Sun, W.-H. *ACS Omega* **2019**, *5*, 83.
- (2) Yada, A.; Nagata, K.; Ando, Y.; Matsumura, T.; Ichinoseki, S.; Sato, K. *Chem. Lett.* **2018**, *47*, 284.
- (3) Kozłowski, M. C.; Dixon, S. L.; Panda, M.; Lauri, G. *J. Am. Chem. Soc.* **2003**, *125*, 6614.
- (4) Lipkowitz, K. B.; Pradhan, M. *J. Org. Chem.* **2003**, *68*, 4648.
- (5) Melville, J. L.; Lovelock, K. R. J.; Wilson, C.; Allbutt, B.; Burke, E. K.; Lygo, B.; Hirst, J. D. *J. Chem. Inf. Model.* **2005**, *45*, 971.
- (6) Zahrt, A. F.; Henle, J. J.; Rose, B. T.; Wang, Y.; Darrow, W. T.; Denmark, S. E. *Science* **2019**, *363*, eaau5631; ■■inserted page no. ok?■■.
- (7) Henle, J. J.; Zahrt, A. F.; Rose, B. T.; Darrow, W. T.; Wang, Y.; Denmark, S. E. *J. Am. Chem. Soc.* **2020**, *142*, 11578.
- (8) Yamaguchi, S.; Sodeoka, M. *Bull. Chem. Soc. Jpn.* **2019**, *92*, 1701.
- (9) Xu, L.-C.; Li, X.; Tang, M.-J.; Yuan, L.-T.; Zheng, J.-Y.; Zhang, S.-Q.; Hong, X. *Synlett* **2020**, *31*, in press; DOI: 10.1055/s-0040-1705977.
- (10) Dietterich, T. G.; Lathrop, R. H.; Lozano-Pérez, T. *Artif. Intell.* **1997**, *89*, 31.
- (11) Kutlushina, A.; Khakimova, A.; Madzhidov, T.; Polishchuk, P. *Molecules* **2018**, *23*, 3094.
- (12) Zankov, D. V.; Matveieva, M.; Nikonenko, A.; Nugmanov, R.; Varnek, A.; Polishchuk, P.; Madzhidov, T. *ChemRxiv* **2020**, reprint; DOI: 10.26434/chemrxiv.13456277.v1.
- (13) Hoonakker, F.; Lachiche, N.; Varnek, A.; Wagner, A. *Int. J. Artif. Intell. Tools* **2011**, *20*, 253.
- (14) Nugmanov, R. I.; Mukhametgaleev, R. N.; Akhmetshin, T.; Gimadiev, T. R.; Afonina, V. A.; Madzhidov, T. I.; Varnek, A. *J. Chem. Inf. Model.* **2019**, *59*, 2516.
- (15) Varnek, A.; Fourches, D.; Hoonakker, F.; Solov'ev, V. P. *J. Comput. Aided. Mol. Des.* **2005**, *19*, 693.
- (16) Riniker, S.; Landrum, G. A. *J. Chem. Inf. Model.* **2015**, *55*, 2562.
- (17) Implemented modeling protocol is available at (accessed June 7, 2021): <https://github.com/dzankov/3D-MIL-QSSR>

Part 3. Modeling reaction characteristics with conjugated machine learning

Conjugated machine learning is a new concept in reaction QSPR modeling that integrates fundamental thermodynamic and kinetic laws with machine learning algorithms. Conjugated models can be built using ridge regression or artificial neural networks. This part demonstrates how fundamental chemical equations can be integrated with a learning algorithm to model the characteristics of binary tautomerism reactions, cycloaddition reactions, and competing E2/S_N2 reactions.

3.1 Methodological developments

1) Design of conjugated learning algorithms. Fundamental thermodynamic and kinetic equations can be integrated with machine learning algorithms by designing special loss functions. This process can be divided into several steps:

1. Design an equation-based loss function in which the main characteristic A is calculated using an integrated equation and the related characteristics B and C .

Define equation F relating main characteristic A with characteristics B and C :

$$A = F(B, C) \quad (9)$$

Design equation-based quadratic loss function for A :

$$A^{pred} = F(B^{pred}, C^{pred}) \quad (10)$$

$$E_A = \|A^{exp} - A^{pred}\|^2 = \|A^{exp} - F(B^{pred}, C^{pred})\|^2 \quad (11)$$

2. Combine equation-based loss function with individual loss functions of related characteristics B and C .

Individual B model:

$$E_B = \|B^{exp} - B^{pred}(\beta_B)\|^2 \quad (12)$$

Individual C model:

$$E_C = \|C^{exp} - C^{pred}(\beta_C)\|^2 \quad (13)$$

Conjugated model:

$$E = aE_A + bE_B + cE_C$$

$$E = a\|A^{exp} - F(B^{pred}, C^{pred})\|^2 + b\|B^{exp} - B^{pred}\|^2 + c\|C^{exp} - C^{pred}\|^2 \quad (14)$$

where a , b , and c are trade-off coefficients that control the contribution of each loss function to the conjugated loss function.

3) Estimate regression weights (parameters) β_B and β_C of the conjugated model:

$$\beta_B, \beta_C = \operatorname{argmin}(\partial\beta_B, \partial\beta_C) \quad (15)$$

Regression weights can be estimated either analytically by calculation of the analytic derivative of E and setting it equal to 0, or the solution can be found numerically by gradient decent approach. The obtained optimal parameters β_B and β_C can be used to generate predictions that satisfy the equation embedded in the conjugated model.

2) Contribution coefficients optimization. The conjugated machine learning algorithms (ridge regression and neural networks) are based on specially designed multi-objective loss functions. In the optimization process, multiple objectives optimization is balanced by adjusting the contribution coefficients (trade-off coefficients) in equation (14). In this research, several approaches were applied to adjust the contribution coefficients.

Grid search. Grid search is a standard method for the optimization of hyperparameters of machine learning methods. In grid search, all available combinations of hyperparameters are tested and the best combination is selected according to a prediction accuracy metric. Grid search can be adapted to find optimal contribution coefficients, but this method can be computationally expensive because of the large number of tested combinations. In this study, the grid search method was used to build conjugated models for predicting the tautomeric constant in Section 3.2, where the conjugated model had a single contribution coefficient α , which can be optimized using a «nested» grid search technique. This type of grid search is based on several consecutive sessions of scanning possible values of the optimized parameter (Figure 22).

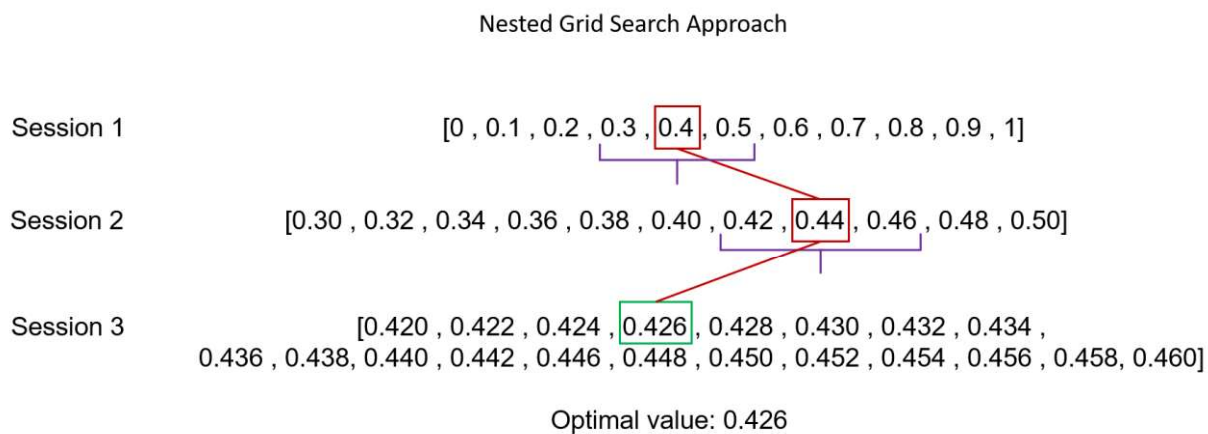


Figure 22. An example of optimization of a single continuous parameter using nested grid search. The range of possible parameter values is iteratively specified until the optimal value is found.

The conjugated models for the tautomeric constant have a single contribution coefficient α , which ranged from 0 to 1. It was observed that the possible optimal value of α was between 0.9 and 1 and to precise it, new values ([0.95, 0.975, 0.9875, 0.99375, 0.996875, 0.9984375, 0.99921875, 0.999609375, 0.9998046875, 0.99990234375]) were scanned, calculated using the following equation $\alpha^{next} \pm 0.1 \times 0.5^n$, where $\alpha^{next} = 1$ is α value and n ranged from 1 to 11. Thus, the grid search is a suitable method for optimizing the small number (1 or 2) of contribution coefficients.

Bayesian optimization. A bayesian optimization is an efficient approach for optimizing the objective function when traditional optimization methods such as gradient descent are not applicable, due to time and computational cost. The idea of bayesian optimization is to build a probability model of the objective function and use it to select the most promising hyperparameters to evaluate in the true objective function. Optimization of hyperparameters of machine learning algorithms is a suitable task for bayesian optimization approaches because to test each combination of hyperparameters one needs to train and validate the model, which can be a time-consuming process, especially for deep learning algorithms. In addition, hyperparameters can be real-valued, discrete, or conditional variables and the simultaneous optimization of which is impossible in traditional optimization methods but is feasible in bayesian optimization. *Hyperopt* [158] is a Python package for the bayesian optimization of ML hyperparameters, based on the Tree-of-Parzen-Estimators (TPE) algorithm [159].

In this research, *hyperopt* was used to optimize the hyperparameters of the ridge regression *conjugated* models for predicting Arrhenius equation parameters. The values of contribution coefficients were sampled from a continuous space defined between 0 to 1, and the regularization

coefficients took discrete values between 10^{-10} to 10^5 . The *hyperopt* algorithm adjusts the hyperparameters by maximizing the validation accuracy of the model.

Genetic algorithm. Evolutionary algorithms are stochastic search methods that seek to improve search performance by exploring a set of promising areas in the solution space [160]. They are based on the mechanisms of evolution of biological organisms. A genetic algorithm is a type of evolutionary computation. A distinctive feature of the genetic algorithm is the emphasis on the use of the crossover operator, which operates by recombining candidate solutions. Genetic algorithm manipulates several solutions simultaneously, which reduces the probability of getting trapped in local optima compared with optimization methods that proceed from point to point in the solution space. Also, genetic algorithms can work with almost any type of optimized function, because it does not require the differentiability of the function. In this research project, the basic implementation of the genetic algorithm (<https://github.com/dzankov/GenOpt>) was adapted to optimize the hyperparameters of machine learning algorithms, including the contribution coefficients in the conjugated models. Preliminary experiments indicated that the developed genetic algorithm approach for optimization of hyperparameters of machine learning algorithms performs similarly to the *hyperopt* approach.

Optimization of contribution coefficients with gradient decent. Contribution coefficients in conjugated neural network algorithms can be automatically adjusted during neural network training using gradient descent. In this approach, contribution coefficients are not fixed before training the neural network as hyperparameters but are internal global parameters of the neural network, which are optimized along with neural network weights. As a result, a single training of the conjugated neural network is enough to obtain optimal values of the contribution coefficients.

3) Descriptors. Each reaction was transformed into the Condensed Graph of Reaction (CGR) [153] generated with the CGRtools module [152]. CGR is derived from the superposition of products and reactants and contains both conventional chemical bonds (single, double, triple, aromatic, etc.) and so-called “dynamic” bonds describing chemical transformations, i.e. breaking or forming a bond or changing bond order. Generated CGRs were processed by the ISIDA tool [161,162] to calculate fragment descriptors by counting the occurrence of particular subgraphs (structural fragments) of different topologies and sizes.

The vector of fragment descriptors for each reaction was concatenated with the vector of solvent descriptors, which included 14 descriptors, describing such properties of solvent as polarity, polarizability, Catalan constants SPP, SA, SB, Kamlet-Taft constants α , β , π^* , dielectric constants, function of the refractive index. These descriptors were successfully applied in previous publications [163–166].

4) Software. The conjugated ridge regression and neural network algorithms are implemented using the PyTorch package [167]. Ridge regression algorithms are implemented using PyTorch tensor objects, which perform matrix calculations using the graphics processing unit (GPU). Neural network algorithms were implemented using standard PyTorch modules. CGR/ISIDA descriptors were generated using CGRTools [152] and CIMTools (<https://github.com/cimm-kzn/CIMtools>) packages. The open-source code of the implemented conjugated ridge regression and neural networks algorithms is available at (<https://github.com/dzankov/CoLearn>).

3.2 Modeling of tautomeric constant

If two tautomeric forms share a common anion, the tautomeric equilibrium constant can be expressed as the difference between the acidity constants of the corresponding tautomers. The tautomeric equation is used in calculating the tautomeric equilibrium constant in commercially available tools for predicting the population of tautomeric forms in water [168,169] (equation-based models). But, in previous works [170,171] it was demonstrated that direct prediction of the tautomeric equilibrium constants often is more accurate. The poor performance of equation-based models in predicting the tautomeric equilibrium constant stems from the fact that it is extremely difficult to measure the acidity of all tautomeric forms which leads to the lack of training data on minor tautomers.

In this study, a tautomeric equation relating the tautomeric equilibrium constant and the acidity of the corresponding tautomers was integrated with ridge regression and neural network algorithms. Three models for predicting the $\log K_T$ tautomeric constant was compared:

- 1) The individual $\log K_T$ model, which is trained with the $\log K_T$ data on 639 tautomeric reactions. The individual $\log K_T$ the model directly predicts the $\log K_T$ for a given reaction.
- 2) The equation-based model, which calculates the prediction of $\log K_T$ using the tautomeric equation and the pK_a of tautomers predicted by the individual pK_a model trained with pK_a data on 2371 organic compounds.
- 3) The conjugated model, which is trained on both $\log K_T$ and pK_a datasets.

Conjugated Quantitative Structure–Property Relationship Models: Application to Simultaneous Prediction of Tautomeric Equilibrium Constants and Acidity of Molecules

Dmitry V. Zankov,[†] Timur I. Madzhidov,^{*,†} Assima Rakhimbekova,[†] Timur R. Gimadiev,[†] Ramil I. Nugmanov,[†] Marina A. Kazymova,[†] Igor I. Baskin,^{†,‡} and Alexandre Varnek^{§,||}

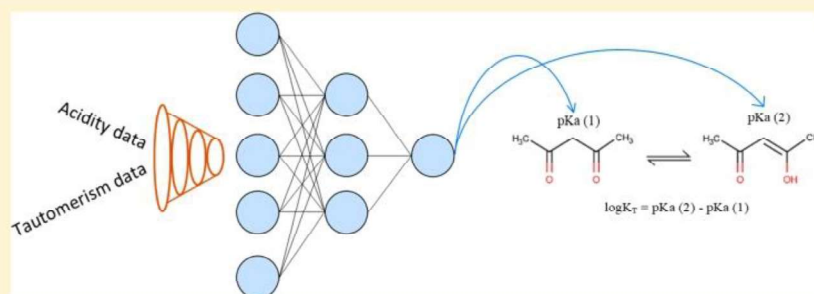
[†]Laboratory of Chemoinformatics and Molecular Modeling, Butlerov Institute of Chemistry, Kazan Federal University, Kremlyovskaya str. 18, 420008 Kazan, Russia

[‡]Faculty of Physics, Moscow State University, Vorob'evy gory 1, 119234 Moscow, Russia

[§]Laboratory of Chemoinformatics, UMR 7140 CNRS, University of Strasbourg, 1, rue Blaise Pascal, 67000 Strasbourg, France

^{||}Institute for Chemical Reaction Design and Discovery (WPI-ICReDD), Hokkaido University, Kita 21 Nishi 10, Kita-ku, 001-0021 Sapporo, Japan

Supporting Information



ABSTRACT: Here, we describe a concept of conjugated models for several properties (activities) linked by a strict mathematical relationship. This relationship can be directly integrated analytically into the ridge regression (RR) algorithm or accounted for in a special case of “twin” neural networks (NN). Developed approaches were applied to the modeling of the logarithm of the prototropic tautomeric constant ($\log K_T$) which can be expressed as the difference between the acidity constants (pK_a) of two related tautomers. Both conjugated and individual RR and NN models for $\log K_T$ and pK_a were developed. The modeling set included 639 tautomeric constants and 2371 acidity constants of organic molecules in various solvents. A descriptor vector for each reaction resulted from the concatenation of structural descriptors and some parameters for reaction conditions. For the former, atom-centered substructural fragments describing acid sites in tautomer molecules were used. The latter were automatically identified using the condensed graph of reaction approach. Conjugated models performed similarly to the best individual models for $\log K_T$ and pK_a . At the same time, the physically grounded relationship between $\log K_T$ and pK_a was respected only for conjugated but not individual models.

1. INTRODUCTION

Physicochemical properties of chemical compounds are often closely related by physically meaningful mathematical relations. In this context, it is important to ensure the validity of such a relationship for the properties predicted by corresponding individual quantitative structure–property relationship (QSPR) models. However, due to the statistical nature of QSPR models and the impossibility to reduce prediction errors to zero, the achievement of the goal is pretty improbable even if each related property is predicted with reasonable accuracy. In order to solve this problem, we introduce the concept of *conjugated* QSPR models, in which relationships between the properties are explicitly embedded in the modeling method-

ology. Here, the conjugated QSPR technique is demonstrated for the case of tautomeric equilibria.

Tautomerism is one of the most important phenomena in organic and bioorganic chemistry. It is a key factor influencing spontaneous mutagenesis, the functioning of nucleic acids, proteins and sugars, and protein–ligand interactions, along with other important natural processes in biology. Considering tautomerism is also of prime importance for computer-aided drug discovery. In the field of chemoinformatics, this phenomenon leads to uncertainty in representing chemical structures, which can cause problems when storing and

Received: August 30, 2019

Published: October 22, 2019

Table 1. Predictive Performance of Individual and Conjugated Models Estimated in 5 × 10-fold Cross Validation^a

data set used for training	model's type	method	hyperparameters	logK _T		pKa	
				RMSE	Q ²	RMSE	Q ²
logK _T	individual	RR	$\alpha = 1 \lambda = 0.3$	0.92 (0.01)	0.67 (0.01)	10.73	−4.87
		NN	$\alpha = 1$	0.85 (0.004)	0.73 (0.003)	10.9	−5.06
logK _T and pKa	conjugated	RR	$\alpha = 0.95 \lambda = 0.1$	0.92 (0.01)	0.67 (0.01)	1.56 (0.01)	0.88 (0.01)
		NN	$\alpha = 0.90$	0.88 (0.05)	0.70 (0.01)	1.52 (0.05)	0.88 (0.01)
pKa	individual	RR	$\alpha = 0 \lambda = 1$	4.90	−8.31	1.56 (0.04)	0.88 (0.02)
		NN	$\alpha = 0$	7.38	−20.14	1.49 (0.04)	0.89 (0.01)

^aMean values for performance metrics are presented, and standard deviations are given in parentheses.

processing chemical data, as well as when building QSAR/QSPR models. For this reason, the importance of taking into account tautomeric transformations when registering compounds, computer design of new drugs, and searching for molecules with desired properties has been repeatedly emphasized.^{1–6} In turn, this led to the development of computational approaches to enumerate possible tautomers of chemical compounds,^{7–15} as well as to evaluate the population of different equilibrium tautomeric forms in solution.^{2,16–18} In the case of prototropic tautomerism, the logarithm of tautomeric constant, logK_T, is equal to the difference between the acidity constants, pKa, of two tautomers sharing common anion after deprotonation:¹⁹

$$\log K_T = \text{pKa}(2) - \text{pKa}(1) \quad (1)$$

Similarly, the tautomeric constant can also be expressed in terms of the basicity constants of the compounds, if both tautomers have a common protonated form. Equation 1 is used in several commercially available tools to estimate logK_T from the values of pKa predicted for two tautomers using QSPR models for acidity, as an intermediate stage for assessing the population of different tautomeric forms in water.^{20,21} The disadvantage of this approach stems from the difficulties to measure the acidity constant of little populated minor tautomers. Moreover, logK_T assessment as the difference of two pKa values according to eq 1 leads to the accumulation of uncertainties of the predictions. These two reasons may significantly reduce the performance of QSPR models for logK_T. Alternatively,^{22,23} logK_T can be modeled directly without any need to use eq 1. However, despite the high predictive performance of these models, the exact agreement of calculated logK_T and pKa values with “fundamental” eq 1 is no more guaranteed.

In order to solve this problem, we suggest developing *conjugated QSPR models*, which output logK_T and pKa always complying to eq 1. For linear conjugated models, an analytical expression extending the popular ridge regression (RR) method was developed. For nonlinear conjugated models, special neural network (NN) architecture was proposed.

2. METHODOLOGY

In this section, we describe the methodology of the preparation of linear and nonlinear conjugated models.

2.1. Conjugated Ridge Regression Model. Let us consider a linear regression equation

$$y_A^{\text{pred}} = Xw \quad (2)$$

where X is a matrix of molecular descriptors, w is a row vector of regression coefficients, and y_A^{pred} is a column vector of predicted pKa values. Regression coefficients w can be found

by minimizing the sum of squared differences between predicted and experimental acidity values y_A^{pred} for a training set:

$$E_A(w) = \sum_i (y_{A,i}^{\text{exp}} - y_{A,i}^{\text{pred}})^2 = (y_A^{\text{exp}} - Xw)^T (y_A^{\text{exp}} - Xw) \quad (3)$$

A combination of eqs 1 and 2 results in an equation for predicted tautomeric equilibrium constants y_T^{pred}

$$y_T^{\text{pred}} = X_2w - X_1w = (X_2 - X_1)w \quad (4)$$

where X_2 and X_1 are descriptor matrix for the tautomers in equilibrium. Similarly to eq 3, regression coefficients w are determined by minimizing the error of logK_T predictions:

$$E_T(w) = (y_T^{\text{exp}} - (X_2 - X_1)w)^T (y_T^{\text{exp}} - (X_2 - X_1)w) \rightarrow \min \quad (5)$$

It should be noted that eqs 2–3 for pKa and eqs 4–5 for logK_T involve exactly the same vector of regression coefficients w . In order to determine optimal w values, three different objective functions should be simultaneously minimized: $E_T(w)$ and $E_A(w)$ calculated according to eqs 3 and 5, respectively, and the model complexity, expressed by the term $w^T w$. A common way of minimizing several objectives simultaneously is to minimize their linear combination with adjustable mixing coefficients:

$$E(w) = \alpha E_T(w) + (1 - \alpha) E_A(w) + \lambda w^T w \rightarrow \min \quad (6)$$

where λ is a regularization coefficient, while α takes values from 0 to 1 and controls the trade-off between minimizing prediction errors of tautomeric constants vs acidity constants. Thus, $\alpha = 1$ (or $\alpha = 0$) correspond to minimizing prediction errors for logK_T (or pKa) according to eq 5 (or eq 3). Values of α between 0 and 1 correspond to the models trained simultaneously on two different data sets: one for logK_T (tautomers data set) and another one for pKa (acidity constants data set).

Differentiation of $E(w)$ with respect to w and equating the derivative to zero leads to an analytical expression for weights w corresponding to the minimum of $E(w)$:

$$w = [\alpha(X_2 - X_1)^T(X_2 - X_1) + (1 - \alpha)X^T X + \lambda I]^{-1} [\alpha(X_2 - X_1)^T y_T^{\text{exp}} + (1 - \alpha)X^T y_A^{\text{exp}}] \quad (7)$$

The vector of regression coefficients w estimated by eq 7 can be used to simultaneously predict pKa and logK_T according to eqs 2 and 4, respectively. Notice that at $\alpha = 0$ and $\alpha = 1$, eq 7 becomes identical to classical RR method²⁴ built solely on the acidity or tautomerism data correspondingly. Indeed, at $\alpha = 0$

$$\mathbf{w} = (\mathbf{X}^T \mathbf{X} + \lambda \mathbf{I})^{-1} \mathbf{X}^T \mathbf{y}_A^{\text{exp}} \quad (8)$$

and at $\alpha = 1$

$$\mathbf{w} = [(\mathbf{X}_2 - \mathbf{X}_1)^T (\mathbf{X}_2 - \mathbf{X}_1) + \lambda \mathbf{I}]^{-1} (\mathbf{X}_2 - \mathbf{X}_1)^T \mathbf{y}_T^{\text{exp}} \quad (9)$$

Thus, the models built with $\alpha = 0$ or $\alpha = 1$ are *individual models*, while the models with $0 < \alpha < 1$ are *conjugated models*.

It should be noted that for two tautomers in equilibrium, solvent and temperature descriptors are identical. Therefore, subtraction $\mathbf{X}_2 - \mathbf{X}_1$ in eqs 4 and 5 results in the deletion of any information about the experimental conditions. Hence, the models for $\log K_T$ resulting from eq 5 are not able to describe the dependence of $\log K_T$ on solvent and temperature. This means that conjugated RR models can be developed only for tautomeric equilibria measured strictly at the same conditions. This limitation is a consequence of the hypothesis of the linear dependence of $\log K_T$ from solvent and temperature descriptors.

2.2. Conjugated Artificial Neural Network Model. To overcome the above limitation by introducing nonlinearity between descriptors and predicted properties, we have developed a special architecture of “twin” neural networks (NN) based on fully connected feed-forward multilayer NN with shared values of connection weights $\mathbf{w}$.

The entire network consists of three “twin” subnetworks I–III (Figure 1). Each subnetwork is a “shallow” multilayer

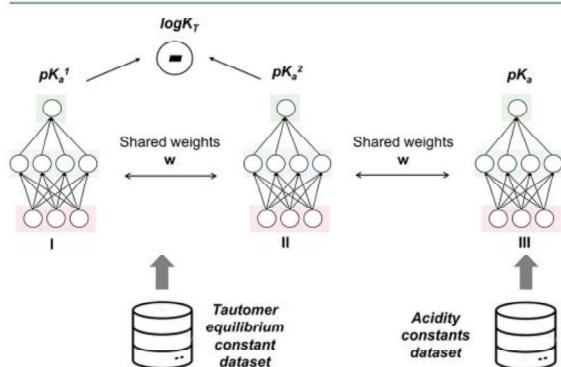


Figure 1. Architecture of the “twin” neural network for simultaneous prediction of tautomeric and acidity constants.

perceptron with a single hidden layer containing several rectified linear units. Tautomer data set feeds to subnetworks I

and II, whereas acidity constants data set feeds to subnetwork III. The outputs of these subnetworks are acidity constants $pK_a(1)$, $pK_a(2)$ (acidities of two tautomeric forms in equilibrium), and pK_a , respectively. The outputs of the subnetworks I and II feed to special unit computing $\log K_T$ according to eq 1. It should be noted that the same values of network parameters $\mathbf{w}$ are used in all three subnetworks, thus subnetworks are identical. These weights are determined by minimizing the functional $E(\mathbf{w})$:

$$E(\mathbf{w}) = \alpha \sum_i (y_{T,i}^{\text{exp}} - y_{T,i}^{\text{pred}})^2 + (1 - \alpha) \sum_j (y_{A,j}^{\text{exp}} - y_{A,j}^{\text{pred}})^2 \quad (10)$$

This NN architecture has been realized with the *PyTorch* package²⁵ which allows usage of the same network in different applications. Since *PyTorch* dynamically constructs a computational graph, it can determine the gradient of the error functional $E(\mathbf{w})$ with respect to weights $\mathbf{w}$ of subnetworks followed by their updating in order to reduce prediction errors for $\log K_T$ and pK_a .

Thus, similarly to conjugated RR models, the NN parameters $\mathbf{w}$ were optimized in such a way that the $\log K_T$ and pK_a were predicted simultaneously. However, unlike the linear ridge regression method, neural networks establish nonlinear relationships between experimental condition descriptors and equilibrium constants.

Thus, the trained network is able to predict both $\log K_T$ and pK_a . It can also be used to build individual models for $\log K_T$ and pK_a . Thus, at $\alpha = 1$, the neural network is trained only on $\log K_T$ because the error of pK_a prediction does not affect the updating of the weights. Similarly, at $\alpha = 0$, the model learns the only pK_a .

3. COMPUTATIONAL DETAILS

3.1. Data. Two data sets were used in the modeling: tautomers data set and acidity constants data set. A data set for tautomeric equilibrium constants (*tautomers data set*) consisted of 575 reactions from reference²² and 64 equilibria collected by M. Nicklaus' group.²⁶ Only binary equilibria with known temperature, solvent, and tautomers ratio were selected. Ring-chain tautomeric equilibria were excluded since eq 1 could not be applied for that case. Thus, the resulting tautomers data set consisted of 639 reactions studied in 24 different solvents and corresponding to 10 different types of prototropic tautomerism. In addition, we used *acidity constants data set* contained

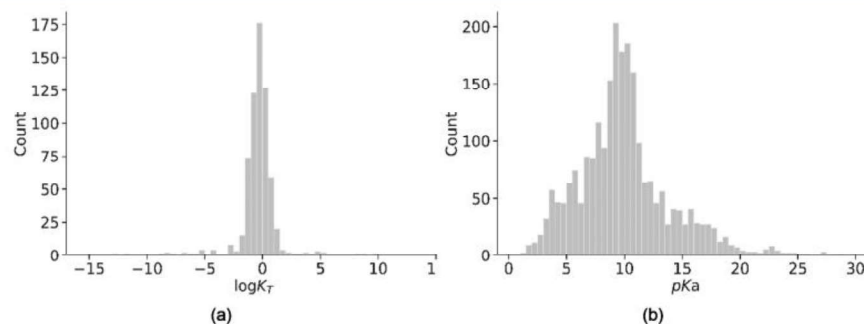


Figure 2. Distribution of (a) the logarithm of tautomeric constant and (b) the acidity constant over the corresponding training sets.

manually collected pKa values of 2371 organic molecules in 13 different solvents, manually extracted from the Palm's handbook.²⁷ Distribution of the logarithm of tautomeric equilibrium constant ($\log K_T$) and pKa for the data sets is given in Figure 2. Both data sets are described in the Supporting Information and a link to download them is given.

Chemical structures were standardized using the ChemAxon Standardizer tool:²⁸ functional groups (nitro, sulfo, and others) were reduced to a standard form, Kekule structures were transferred to aromatic structures if they were in accordance with the Hückel's rule. The data sets were also visually inspected in order to avoid the errors in the data.

The atom-to-atom mapping was determined in a consensus manner²⁹ using the ChemAxon Standardizer²⁸ and GGA Indigo programs³⁰ followed by visual inspection.

The predictive performance of the models was assessed on two external test sets taken from paper.²² The first one (TEST1) consisted of tautomeric equilibria present in the training data set but studied under different experimental conditions. The second test set (TEST2) contained unique transformations that were absent in the training data set. Notice that ring-chain tautomerization reactions were excluded from these test sets.

3.2. Descriptors. Descriptors vector for each molecule resulted from the concatenation of structural and condition descriptors. The ISIDA fragment descriptors²⁴ were used to encode molecular structure. At the first stage, the atoms representing acid sites in molecules were labeled. For the acidity constants data set, the labels were assigned manually when the corresponding molecule entered the database. For tautomeric rearrangements, the assignment of the label was performed automatically. For this purpose, each tautomeric equilibrium was encoded by condensed graph of reaction (CGR).²⁴ Hydrogen atoms in CGR were explicitly accounted for (Figure 3). Obtained CGRs allow identification of breaking or forming bonds and, hence, the atoms adjacent to these bonds. In prototropic tautomeric rearrangement, a hydrogen atom moves from atom A in tautomer 1 to atom B in tautomer 2. Thus, both atoms A and B were marked as acid sites in the corresponding tautomeric forms (Figure 3).

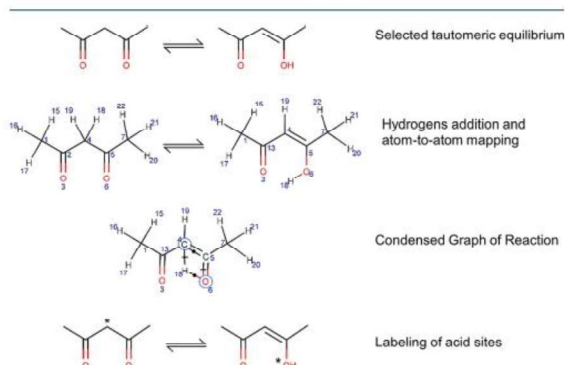


Figure 3. Automatized labeling acid sites in tautomeric rearrangement reactions with the help of condensed graph of reaction (CGR). The latter results from the superposition of atoms of two tautomers having the same number. Black dots and perpendicular dashes in CGR depict, respectively, the formed and broken bonds. The asterisk (*) marks the atoms of the acid center in tautomers.

Fragment descriptors were calculated using the ISIDA Fragmentor program.²⁴ This program computes fragment descriptors by enumerating fragments (subgraphs of molecular graphs) belonging to some topology type (for example, chains) and evaluating their values by counting the number of times they occur in each molecule from the data set. In this study, atom-centered fragments including from 1 to 3 atoms were used as descriptors. Upon fragment generation marked atom approach^{31–33} was used, i.e. fragments that include an acid center label (so-called marked atom) are distinguished from those without such labels. Both fragments with and without marked atoms were included in the pool as different descriptors (so-called MA3 approach in papers^{31,33}). This allowed us to distinguish acid sites from other atoms.

A vector of fragment descriptors was concatenated with a vector of descriptors characterizing the solvent and the temperature. A set of 14 solvent descriptors including Catalan constants SPP,³⁴ SA,³⁵ SB,³⁶ Kamlet–Taft constants α ,³⁷ β ,³⁸ π^* ,³⁹ four functions of dielectric constant ϵ (Born function $f_B = \frac{\epsilon-1}{\epsilon}$ and Kirkwood function $f_K = \frac{\epsilon-1}{2\epsilon+1}$, $f_1 = \frac{\epsilon-1}{\epsilon+1}$, and $f_2 = \frac{\epsilon-1}{\epsilon+2}$), three functions of the refractive index n_D^{20} ($g_1 = \frac{n^2-1}{n^2+1}$, $g_2 = \frac{n^2-1}{2n^2+1}$, $h = \frac{(n^2-1)(\epsilon-1)}{(2n^2+1)(2\epsilon+1)}$) were used. Solvent parameters were taken from original literature sources (SPP, SA, SB, α , β , π^*) or were calculated on the basis of dielectric constants (f_B, f_K, f_1, f_2) and refractive indices (g_1, g_2, h). Since in some cases aqueous–organic mixtures were used as solvents, the mole fraction of the organic solvent in the mixture (for a pure solvent, 100%) was also used as descriptor. Besides, the inverse temperature, $1/T$, was also used. Such reaction condition descriptions have shown good results in our previous papers.^{22,40,41}

3.3. Building and Validation of QSPR Models.

3.3.1. Descriptor Preparation. Molecular descriptors resulting from the concatenation of structural and condition descriptors were computed according to the procedure described in section 3.2. Descriptors matrix X was prepared for all molecules from the acidity constants data set, whereas X_1 and X_2 matrices were prepared for the tautomers data set. Notice that the matrices X , X_1 , and X_2 have the same set of descriptors in the columns. The number of rows in the matrix X coincides with the number of compounds in the acidity data set, whereas in the matrices X_1 and X_2 —with the number of tautomeric equilibria in the tautomers data set.

3.3.2. Ridge Regression Modeling. Equation 7 was used to identify optimal regression coefficients w in RR models. Optimal hyperparameters α and λ were found in a grid search of possible combinations of α and λ . Two ways of varying α were considered: (i) linearly from 0 to 1 with step 0.1 and (ii) nonlinearly as $\alpha = 1 - 0.1 \times \left(\frac{1}{2}\right)^n$ for n varying from 0 to 10 with step 1. The second approach was found particularly useful to find the optimal value of α since it is often close to 1. The following values of the regularization coefficient λ were used in the grid search: 0.0001, 0.0003, 0.001, 0.003, 0.01, 0.03, 0.1, 0.3, 1, 3, 10, 30, 100, 300.

The optimal values of the hyperparameters (α and λ) corresponded to the maximal value of the coefficient of determination Q^2 (or, equivalently, the minimal value of the root-mean-square error of prediction, RMSE) found in 10-fold cross-validation procedure repeated 5 times after random structures reshufflings (5×10 CV):

$$Q^2(R^2) = 1 - \frac{\sum_{i=1}^N (y_i^{\text{exp}} - y_i^{\text{pred}})^2}{\sum_{i=1}^N (y_i^{\text{exp}} - \bar{y}^{\text{exp}})^2} \quad (11)$$

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^N (y_i^{\text{exp}} - y_i^{\text{pred}})^2}{N}} \quad (12)$$

3.3.3. Neural Network Modeling. The NN described in section 2 was implemented using the *PyTorch* framework.²⁵ The NN hyperparameters such as the number of neurons in the hidden layer and the number of learning epochs were optimized. The weights ω were updated using the Adam optimizer.⁴² In one batch 64 tautomeric equilibrium reactions and 216 acidity data were used. The batch size was selected in a way that all tautomeric and acidity data is fed to the network in one epoch. Feed-forward networks with one hidden layer and 2^n ($n = 6 \dots 11$) neurons in layer with the ReLU activation function was considered. Our tests have shown that the multilayer architecture does not provide any benefit to the model performance. Based on cross-validation results, a network containing one hidden layer with 512 neurons trained on 300 epochs with a learning rate of 0.001 was selected.

It has been found that the value of L2 weight regularization did not affect the accuracy of $\log K_T$ and pKa predictions, therefore this hyperparameter was not optimized and set to 0. The coefficient α varied in the same range as in RR modeling.

The source code for RR and NN modeling is provided as part of the Supporting Information.

3.3.4. Model Validation. Each individual (RR or NN) models for $\log K_T$ or pKa were validated in 5×10 CV both on tautomers and acidity constants data sets. The hyperparameters leading to the minimum average RMSE of $\log K_T$ and pKa were selected. A model for one property (e.g., $\log K_T$) was applied to predict both $\log K_T$ and pKa in two different data sets. The individual and conjugated model performances in cross-validation are reported in Table 1, whereas an example of $\log K_T$ and pKa predictions for one selected tautomeric equilibrium is given in Table 2.

Particular attention was paid to the prediction of acidity constants of minor tautomers. Table 3 reports the results of the application of individual and conjugated models to a subset of 18 tautomeric keto–enol equilibria.

Table 2. Predicted and Experimental Values of Tautomer Equilibrium Constant and Tautomer Acidity^a

				
	method	$\log K_T$	pKa(1)	pKa(2)
experiment		−0.21	9.93	9.71 ^b
individual models ($\log K_T$)	RR	−0.27	0.66	0.39
	NN	−0.25	0.41	0.16
individual models (pKa)	RR	0.93	10.22	11.16
	NN	0.83	10.29	11.12
conjugated models	RR	−0.26	10.26	10.00
	NN	−0.22	10.16	9.94

^aConditions: 100% dioxane, 25 °C. ^bExperimental value for pKa(2) was calculated according to eq 1 based on measured acidity of ketone form and tautomeric equilibrium constant.

Table 3. Performance of the Models for Predicting the Acidity Constant of Major (Ketone) and Minor (Enol) Tautomers for a Subset of 18 Selected Keto–Enol Equilibria

model's type	method	RMSE	R ²
Tautomer 1 (Ketone)			
individual (pKa)	RR	0.94	0.75
	NN	0.92	0.76
conjugated	RR	0.97	0.74
	NN	0.98	0.74
Tautomer 2 (Enol)			
individual (pKa)	RR	2.31	−0.33
	NN	2.23	−0.28
conjugated	RR	0.93	0.77
	NN	0.95	0.78

Examples of prediction outliers are drawn in Table 4. Performances of the models on external sets TEST1 and TEST2 are given in Table 5.

Table 4. Some of Outliers of the Conjugated NN Model

No	Equilibrium	Predicted $\log K_T$	Experimental $\log K_T$	Conditions
1		3.02	8.3	H ₂ O, 298 K
2		2.78	9.6	H ₂ O, 298 K
3		−4.28	−1.28	Acetone (100%), 303 K
4		1.41	4.69	H ₂ O, 293 K
5		−8.81	−1.19	Acetone (100%), 303 K
6		−1.25	−8.7	H ₂ O (100%), 294 K

4. RESULTS AND DISCUSSION

4.1. Individual and Conjugated QSPR Models.

4.1.1. Individual Models. Individual RR models for $\log K_T$ and pKa were obtained using eq 7 with $\alpha = 1$ and $\alpha = 0$, respectively. The models for pKa were also used to predict tautomeric constants according to eq 1. Individual NN models for these properties were built by minimizing the error function (eq 10) with $\alpha = 1$ (for individual $\log K_T$ models) and $\alpha = 0$ (for individual pKa models).

Variation of regularization coefficient λ from 0.0001 to 1 in RR modeling did not change much the accuracy of $\log K_T$ prediction ($Q^2 = 0.67$). However, starting from $\lambda = 3$ the model performance started to decrease. The predictive performance of the individual NN model for $\log K_T$ is a bit higher ($Q^2 = 0.73$).

The predictive performance of the RR individual model for pKa ($\alpha = 0$) increases with the regularization coefficient λ and reaches its maximum $Q^2 = 0.88$ at $\lambda = 3.0$. The NN model for pKa performed similarly ($Q^2 = 0.89$).

As one can see from Table 1, both RR and NN individual models for predicting pKa ($\alpha = 0$) were not able to assess $\log K_T$ correctly with the help of eq 1. The NN and RR models trained on $\log K_T$ data only (at $\alpha = 1$) can be used to predict "pseudo-acidities" whose difference is $\log K_T$. However, as

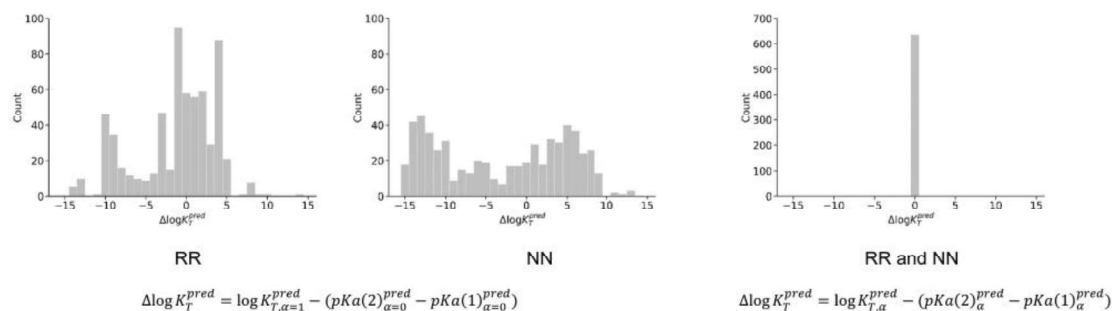


Figure 4. Comparison of the difference in direct prediction of the tautomeric equilibrium constant and its indirect computation based on eq 1 based on predictions made with individual RR (left) and NN (center) models, as well as by the conjugated model (right). The best individual and conjugated models given in Table 1 are used here (α corresponds to the best conjugated model; see Table 1).

expected, their values do not generally correspond to the actual values of acidity constants of tautomers (see Table 1).

4.1.2. Conjugated QSPR Models. When α varies from 0 to 1 (excluding the bounds), the models are built on both tautomers and acidity constants data sets. Such models can predict $\log K_T$ and pKa of the corresponding tautomers simultaneously or acidity of any molecule. Hyperparameters of conjugated models discussed below were optimized to achieve the optimal performance of both properties (largest sum of Q^2 on $\log K_T$ and pKa data). Conjugated models can also be optimized to achieve the best performance on $\log K_T$ or pKa, but anyway, the performance of models obtained does not exceed the one for individual models according to statistical tests.

In cross-validation, both RR and NN models perform similarly to individual models for $\log K_T$ (applied to tautomeric equilibrium constant predictions) and pKa (applied to acidity constant predictions); see Table 1.

Results reported for cross-validation in Table 1 are in agreement with $\log K_T$ and pKa predictions for the selected molecule (see Table 2). One may see that predictions of tautomers acidity constants with the individual model for $\log K_T$ and, vice versa, predictions of $\log K_T$ with the individual model for pKa fail. On the other hand, conjugated models predict both properties with reasonable accuracy.

Conjugated modeling represents a special case of multitask learning when an exact equation uniting two modeled properties is known. The link between properties predicted with the conjugated model is assured by the model building algorithm.

In order to illustrate this for the tautomers case, the distributions of difference between predicted directly tautomeric equilibrium constant and that found as the difference of acidities of two tautomeric forms $\Delta \log K_T^{\text{pred}} = \log K_T^{\text{pred}} - (pK_a(2)^{\text{pred}} - pK_a(1)^{\text{pred}})$ were prepared for $\log K_T$ and pKa predicted with individual or conjugated models. One may see that in the case of conjugated model $\Delta \log K_T^{\text{pred}} = 0$ which strictly follows eq 1; see Figure 4 (right). For $\log K_T$ and pKa predicted with individual models, $\Delta \log K_T^{\text{pred}}$ varies from -15 to 13 , see Figure 4 (left and center). It means that eq 1 is not respected when $\log K_T$ and pKa are modeled individually.

In general, individual pKa models predict acidity constants of organic molecules with reasonable accuracy. However, pKa prediction of minor tautomers is problematic because these compounds are underrepresented in the acidity constants data set. This is illustrated for a subset of 18 keto–enol equilibria

extracted from the tautomers data set. The pKa values for major tautomers ($pK_a(1)$) measured at the same experimental conditions as $\log K_T$ were extracted from the acidity constants data set. Since experimental pKa values for minor tautomers ($pK_a(2)$) were not available, they were estimated from $\log K_T$ and $pK_a(1)$ using eq 1.

To estimate the ability to predict the acidity constant of minor tautomers, we used the leave-one-out procedure in which the model was built on the entire tautomers set excluding one (out of 18) keto–enol equilibria.

One can see from the results presented in Table 3 that only the conjugated models were able to predict accurately the acidity constants of both, major (ketone) and minor (enol), tautomers, while the individual models provided reasonably accurate predictions only for the major tautomer.

Equilibria for which deviations of $\log K_T$ values predicted with the NN model from the experimental ones exceeded 3 RMSE were considered as outliers; six of them are given in Table 4 and discussed below. They could be interpreted by either erroneous experimental data used in the model building or specificity of particular equilibria. Thus, $\log K_T$ values for equilibria 1 and 2 are pretty large. When such examples are included in a test set at a given cross-validation fold, their equilibrium constants are outside of the $\log K_T$ range for the corresponding training set. This may lead to big errors because of data extrapolation. Notice 1 and 2 were also detected as outliers in our previous study²² in which the same tautomers data set was used. Reactions 3 and 4 represent the same equilibrium with permuted reactant and product. If reaction conditions are similar, equilibrium constants of forward and backward reactions (by absolute value) must also be similar and have opposite sign. A drastic difference in absolute values of experimental $\log K_T$ (1.28 for 3 and 4.68 for 4) seems to be erroneous and can hardly be explained by solvent and temperature effects. For conjugated model, based on eq 1, $\log K_T$ for forward and backward reactions are predicted with different signs and thus equilibrium constants for reaction 3 and 4 are predicted with large errors. Drastic difference between experimental equilibrium constant of reactions 5 and 6 is likely to be erroneous too.

4.2. External Validation. The performance of the conjugated model to predict $\log K_T$ was assessed on two external test sets TEST1 and TEST2 described in section 3.1. It was revealed by the fragment control applicability domain⁴³ that 13 out of 21 instances in TEST2 included new structural moieties absent in training set. Such molecules are potentially

subjected to large extrapolation errors and were excluded from consideration as out of the model's applicability domain. TEST1 molecules cannot be outside of the fragment control applicability domain since the data set included only those tautomers pairs that exist in the training set.

As one may see from Table 5, reasonable RMSE values in the range of 0.82–0.89 were obtained for both sets. They are

Table 5. Validation of Conjugated Models for $\log K_T$ on 17 Equilibria from the TEST1 Set and 8 Equilibria from the TEST2 Set Retained by the Models' Applicability Domains

method	RMSE
TEST1	
RR	0.89
NN	0.82
TEST2	
RR	0.85
NN	0.84

also similar to RMSE values (0.92 for RR and 0.88 for NN) obtained for the conjugated model in cross-validation, see Table 1. Despite this, we cannot directly compare these results with paper²² since data sets are different (ring–chain tautomers were excluded, and the fragment control applicability domain was not applied in ref 22), but results of external validation are similar to corresponding values in it (0.66 for TEST1 and 1.63 for TEST2).

5. CONCLUSIONS

In this paper, we introduce the concept of *conjugated* QSPR models for the simultaneous prediction of several mutually related properties. Mathematical relations between property values are ensured by specially constructed machine learning algorithms. Here, conjugated RR and NN models were built for the prototropic tautomeric equilibrium constant ($\log K_T$) and acidity constant (pKa) related by eq 1. For this purpose, we have derived an analytical expression for calculating regression coefficients in the RR and developed a special architecture of NN strictly accounting for eq 1.

The predictive performance of conjugated models was compared with that of individual models for $\log K_T$ and pKa built independently using classical QSPR workflow. It has been demonstrated that individual models for pKa are not able to predict accurately $\log K_T$ using eq 1. Moreover, individual models for pKa fail to predict acidity values for minor tautomers because of the lack of experimental data for the models training. This problem can also be solved with the help of the conjugated models in which eq 1 is strictly respected, and hence, accurate predictions of $\log K_T$ and pKa for the major tautomer leads to a good prediction of acidity of the minor tautomer as well.

■ ASSOCIATED CONTENT

● Supporting Information

The Supporting Information is available free of charge on the ACS Publications website at DOI: 10.1021/acs.jcim.9b00722.

Source code for the RR and NN models, description of tautomeric equilibrium and acidity data sets, and a link to download them (PDF)

■ AUTHOR INFORMATION

Corresponding Author

*Email: Timur.Madzhidov@kpfu.ru.

ORCID

Dmitry V. Zankov: 0000-0002-6201-3347
 Timur I. Madzhidov: 0000-0002-3834-6985
 Assima Rakhimbekova: 0000-0002-6820-6385
 Timur R. Gimadiev: 0000-0001-5012-0308
 Ramil I. Nugmanov: 0000-0002-8541-9681
 Marina A. Kazymova: 0000-0002-4111-8895
 Igor I. Baskin: 0000-0003-0874-1148
 Alexandre Varnek: 0000-0003-1886-925X

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

The work was funded by the Russian Science Foundation (project No 19-73-10137).

■ ABBREVIATIONS

RR, ridge regression; NN, artificial neural network; QSPR, quantitative structure–property relationship; CGR, condensed graph of reaction

■ REFERENCES

- (1) Pospisil, P.; Ballmer, P.; Scapozza, L.; Folkers, G. Tautomerism in Computer-Aided Drug Design. *J. Recept. Signal Transduction Res.* **2003**, *23* (4), 361–371.
- (2) Clark, T. Tautomers and Reference 3D-Structures: The Orphans of in Silico Drug Design. *J. Comput.-Aided Mol. Des.* **2010**, *24* (6–7), 605–611.
- (3) Warr, W. Tautomerism in Chemical Information Management Systems. *J. Comput.-Aided Mol. Des.* **2010**, *24* (6–7), 497–520.
- (4) Sayle, R. A. So You Think You Understand Tautomerism? *J. Comput.-Aided Mol. Des.* **2010**, *24* (6–7), 485–496.
- (5) Katritzky, A. R.; Hall, C. D.; El-Gendy, B. E.-D. M.; Draghici, B. Tautomerism in Drug Discovery. *J. Comput.-Aided Mol. Des.* **2010**, *24* (6–7), 475–484.
- (6) Sitzmann, M.; Ihlenfeldt, W.-D.; Nicklaus, M. C. Tautomerism in Large Databases. *J. Comput.-Aided Mol. Des.* **2010**, *24* (6–7), 521–551.
- (7) ACD/Tautomers; Advanced Chemistry Development, Inc., Toronto, ON, Canada, 2015; www.acdlabs.com (accessed 14/10/2019).
- (8) JChem. Calculator Plugins 15.8.3; ChemAxon Kft., Budapest, Hungary, 2015; <http://www.chemaxon.com> (accessed 14/10/2019).
- (9) MN Tautomer; Molecular Networks GmbH, Germany and Altamira, LLC, USA, 2015.
- (10) LigPrep Tautomeriser; Schrödinger, LLC, 2019; <https://www.schrodinger.com/ligprep> (access date 14/10/2019).
- (11) CACTVS; Xemistry GmbH, 2019; <http://www.xemistry.com> (accessed 14/10/2019).
- (12) QUACPAC; OpenEye Scientific Software, 2019; <https://www.eyesopen.com/quacpac> (accessed 14/10/2019).
- (13) BIOVIA Pipeline Pilot; BIOVIA, USA, 2019; <https://www.3dsbiovia.com/products/collaborative-science/biovia-pipeline-pilot/> (accessed 14/10/2019).
- (14) Harańczyk, M.; Gutowski, M. Quantum Mechanical Energy-Based Screening of Combinatorially Generated Library of Tautomers. TauTGen: A Tautomer Generator Program. *J. Chem. Inf. Model.* **2007**, *47* (2), 686–694.
- (15) Kochev, N. T.; Paskaleva, V. H.; Jeliakova, N. Ambit-Tautomer: An Open Source Tool for Tautomer Generation. *Mol. Inf.* **2013**, *32* (5–6), 481–504.

- (16) Greenwood, J. R.; Calkins, D.; Sullivan, A. P.; Shelley, J. C. Towards the Comprehensive, Rapid, and Accurate Prediction of the Favorable Tautomeric States of Drug-like Molecules in Aqueous Solution. *J. Comput.-Aided Mol. Des.* **2010**, *24* (6–7), 591–604.
- (17) Pospisil, P.; Ballmer, P.; Scapozza, L.; Folkers, G. Tautomerism in Computer-Aided Drug Design. *J. Recept. Signal Transduction Res.* **2003**, *23* (4), 361–371.
- (18) Oellien, F.; Cramer, J.; Beyer, C.; Ihlenfeldt, W.-D.; Selzer, P. M. The Impact of Tautomer Forms on Pharmacophore-Based Virtual Screening. *J. Chem. Inf. Model.* **2006**, *46* (6), 2342–2354.
- (19) Angyl, S. J.; Angyal, C. L. 268. The Tautomerism of N-Hetero-Aromatic Amines. Part I. *J. Chem. Soc.* **1952**, 1461.
- (20) Szegezdi, J.; Csizmadia, F. Tautomer Generation. *PKa Based Dominance Conditions for Generating Dominant Tautomers*; In American Chemical Society Fall Meeting, Boston, MA, August 19–23; 2007.
- (21) Milletti, F.; Storch, L.; Sforna, G.; Cross, S.; Cruciani, G. Tautomer Enumeration and Stability Prediction for Virtual Screening on Large Chemical Databases. *J. Chem. Inf. Model.* **2009**, *49* (1), 68–75.
- (22) Gimadiev, T. R.; Madzhidov, T. I.; Nugmanov, R. I.; Baskin, I. I.; Antipin, I. S.; Varnek, A. Assessment of Tautomer Distribution Using the Condensed Reaction Graph Approach. *J. Comput.-Aided Mol. Des.* **2018**, *32* (3), 401–414.
- (23) Glavatskikh, M.; Madzhidov, T.; Baskin, I. I.; Horvath, D.; Nugmanov, R.; Gimadiev, T.; Marcou, G.; Varnek, A. Visualization and Analysis of Complex Reaction Data: The Case of Tautomeric Equilibria. *Mol. Inf.* **2018**, *37* (9–10), 1800056.
- (24) Varnek, A.; Fourches, D.; Hoonakker, F.; Solov'ev, V. P. Substructural Fragments: An Universal Language to Encode Reactions, Molecular and Supramolecular Structures. *J. Comput.-Aided Mol. Des.* **2005**, *19* (9–10), 693–703.
- (25) Paszke, A.; Gross, S.; Chintala, S.; Chanan, G.; Yang, E.; DeVito, Z.; Lin, Z.; Desmaison, A.; Antiga, L.; Lerer, A. Automatic Differentiation in PyTorch. In *NIPS 2017 Workshop Autodiff Submission*; OpenReview.net: Long Beach, CA, USA, 2017; pp 1–4.
- (26) Nicklaus, M. C. and team. Tautomer Structures Extracted from Experimental Literature, Release 1. <https://cactus.nci.nih.gov/download/tautomer/> (accessed 14/10/2019).
- (27) Palm, V. A. *Tables of Rate and Equilibrium Constants of Heterolytic Organic Reactions*; VINITI: Moscow, 1978.
- (28) ChemAxon Standardizer, version 19.12; <http://www.chemaxon.com> (accessed 14/10/2019).
- (29) Madzhidov, T. I.; Nugmanov, R. I.; Gimadiev, T. R.; Lin, A. I.; Antipin, I. S.; Varnek, A. Consensus Approach to Atom-to-Atom Mapping in Chemical Reactions. *Butlerov Commun.* **2015**, *44* (12), 170–176.
- (30) Indigo Toolkit; EPAM Systems, Inc., 2014; <http://lifescience.opensource.epam.com/indigo/> (accessed 14/10/2019).
- (31) Glavatskikh, M.; Madzhidov, T.; Solov'ev, V.; Marcou, G.; Horvath, D.; Graton, J.; Le Questel, J.-Y.; Varnek, A. Predictive Models for Halogen-Bond Basicity of Binding Sites of Polyfunctional Molecules. *Mol. Inf.* **2016**, *35* (2), 70–80.
- (32) Glavatskikh, M.; Madzhidov, T.; Solov'ev, V.; Marcou, G.; Horvath, D.; Varnek, A. Predictive Models for the Free Energy of Hydrogen Bonded Complexes with Single and Cooperative Hydrogen Bonds. *Mol. Inf.* **2016**, *35* (11–12), 629–638.
- (33) Ruggiu, F.; Solov'ev, V.; Marcou, G.; Horvath, D.; Graton, J.; Le Questel, J.-Y.; Varnek, A. Individual Hydrogen-Bond Strength QSPR Modelling with ISIDA Local Descriptors: A Step Towards Polyfunctional Molecules. *Mol. Inf.* **2014**, *33* (6–7), 477–487.
- (34) Catalán, J.; López, V.; Pérez, P.; Martín-Villamil, R.; Rodríguez, J.-G. Progress towards a Generalized Solvent Polarity Scale: The Solvatochromism of 2-(Dimethylamino)-7-Nitrofluorene and Its Homomorph 2-Fluoro-7-Nitrofluorene. *Liebigs Ann.* **1995**, 1995 (2), 241–252.
- (35) Catalán, J.; Díaz, C. A Generalized Solvent Acidity Scale: The Solvatochromism of o-Tert-Butylstilbazolium Betaine Dye and Its Homomorph o,O'-Di-Tert-Butylstilbazolium Betaine Dye. *Liebigs Ann.* **1997**, 1997 (9), 1941–1949.
- (36) Catalán, J.; Díaz, C.; López, V.; Pérez, P.; De Paz, J.-L. G.; Rodríguez, J. G. A Generalized Solvent Basicity Scale: The Solvatochromism of 5-Nitroindoline and Its Homomorph 1-Methyl-5-Nitroindoline. *Liebigs Ann.* **1996**, 1996 (11), 1785–1794.
- (37) Yokoyama, T.; Taft, R. W.; Kamlet, M. J. The Solvatochromic Comparison Method. 3. Hydrogen Bonding by Some 2-Nitroaniline Derivatives. *J. Am. Chem. Soc.* **1976**, *98* (11), 3233–3237.
- (38) Kamlet, M. J.; Taft, R. W. The Solvatochromic Comparison Method. I. The Beta-Scale of Solvent Hydrogen-Bond Acceptor (HBA) Basicities. *J. Am. Chem. Soc.* **1976**, *98* (2), 377–383.
- (39) Kamlet, M. J.; Abboud, J. L.; Taft, R. W. The Solvatochromic Comparison Method. 6. The Pi* Scale of Solvent Polarities. *J. Am. Chem. Soc.* **1977**, *99* (18), 6027–6038.
- (40) Gimadiev, T.; Madzhidov, T.; Tetko, I.; Nugmanov, R.; Casciuc, I.; Klimchuk, O.; Bodrov, A.; Polishchuk, P.; Antipin, I.; Varnek, A. Bimolecular Nucleophilic Substitution Reactions: Predictive Models for Rate Constants and Molecular Reaction Pairs Analysis. *Mol. Inf.* **2019**, *38*, 1800104.
- (41) Nugmanov, R. I.; Madzhidov, T. I.; Khaliullina, G. R.; Baskin, I. I.; Antipin, I. S.; Varnek, A. A. Development of “Structure-Property” Models in Nucleophilic Substitution Reactions Involving Azides. *J. Struct. Chem.* **2014**, *55* (6), 1026–1032.
- (42) Kingma, D. P.; Ba, J. Adam: A Method for Stochastic Optimization. *arXiv.org* **2014**, 1412.6980.
- (43) Varnek, A.; Fourches, D.; Horvath, D.; Klimchuk, O.; Gaudin, C.; Vayer, P.; Solov'ev, V.; Hoonakker, F.; Tetko, I. V.; Marcou, G. ISIDA - Platform for Virtual Screening Based on Fragment and Pharmacophoric Descriptors. *Curr. Comput.-Aided Drug Des.* **2008**, *4* (3), 191–198.

Conclusion

In this study, a tautomeric equation relating the tautomeric equilibrium constant and the acidity of the corresponding tautomers was integrated with ridge regression and neural network algorithms. Three main approaches for predicting the tautomeric constant were compared: the individual model, the equation-based model, and the conjugated model. The individual $\log K_T$ the model predicts the tautomeric constant more accurately than the equation-based model, which calculates the tautomeric constant based on the predicted acidities of tautomers and the tautomeric equation. The reason for the poor performance of the equation-based model is that the pK_a predictions of minor tautomers (e.g., enols) have a high prediction error since they are not represented in the training set. However, the conjugated model accurately predicts the acidity of the minor forms and, consequently, the tautomeric constant.

Conjugated models can be built using ridge regression and neural network algorithms. The current architecture of the conjugated ridge regression ignores the conditions (solvent, temperature) of tautomerism reactions, which decreases the predictions accuracy of $\log K_T$. On the contrary, the conjugated model based on neural networks takes into account these conditions, which leads to slightly higher accuracy in predicting $\log K_T$. In addition, in the case of large datasets, matrix calculations in ridge regression can be significantly slower, as well as require more memory resources. In this case, neural networks can be trained on batches of data, which makes it possible to use them to build conjugated models on large datasets.

3.3 Modeling of Arrhenius equation parameters

Introduction

A chemical reaction can be quantitatively described by such kinetic characteristics as the rate constant ($\log k$), the pre-exponential factor ($\log A$), and activation energy (E_a). Their knowledge is of particular importance because the distribution of reactants and product concentration at any moment can be calculated based on known kinetics. QSPR modeling of chemical reactions has made significant progress in recent years [172–174]. QSPR methodology employs machine learning algorithms to the data on reaction characteristics measured in the experiment to predict the same characteristics for new reactions. Many approaches were proposed for reaction rate calculation. Usually, quantum chemistry approaches are used for the search for elementary reaction mechanisms and estimate reaction barriers and rates [175–177]. Computationally efficient machine learning potentials were shown to be a valuable alternative to quantum chemistry in the estimation of local minima and transition states energy [178]. Machine learning is currently widely used to predict reaction rate constants based on structural features of reactants and products represented by a set of chemical descriptors [179]. Thus approach may be dated back to early studies based on the Linear Free Energy Principle [180] and the application of substituent constants as descriptors [181]. It has also been shown that quantum chemical descriptors are a good alternative to structural descriptors [182].

In our previous publications, we reported predictive models for the rate constants of S_N2 [183,184] and $E2$ [185,186] reactions. There are also examples of machine learning applications for predicting the activation energies of reactions. Singh *et al.* applied popular machine learning algorithms to predict the activation barriers of hydrogenation/dehydrogenation reactions [187]. Gambow and coworkers developed a deep graph convolutional neural network trained on the activation barriers of gas-phase reactions obtained with quantum-chemical calculations [175,188]. Jorner *et al.* proposed an approach that combines traditional DFT transition state modeling and machine learning [182] and trained the model using different machine learning algorithms to accurately predict the reaction barriers of the nucleophilic aromatic substitution reaction (S_NAr).

Previously, the temperature dependence of the reaction rate was mostly modeled by adding the temperature to the set of structural descriptors [186]. In this case, the dependence of the rate constant ($\log k$) on the temperature known to be expressed by the Arrhenius equation (1) that relates reaction rate with the temperature and two other parameters that are assumed to be temperature independent: the pre-exponential factor (A), and activation energy (E_a) was assumed to be learnt by the machine learning model.

In our previous study [166] we reported SVR (Support Vector Regression) and GTM (Generative Topographic Mapping) modeling of $\log k$, $\log A$ and E_A of cycloaddition reactions. Two scenarios for $\log k$ assessment was examined. In the first scenario, the SVR algorithm learns to predict $\log k$ directly from descriptors. In the second scenario, two independent individual models are built: (i) for predicting the $\log A$ and (ii) for predicting the E_A , which were used to calculate $\log k$ using the Arrhenius equation:

$$\log k = \log A - \frac{E_A}{2.303RT} \quad (16)$$

We observed that the predicted values of $\log k$ calculated using the Arrhenius equation (*Arrhenius-based* model) were less accurate in comparison to the *individual* model built using the experimental values of $\log k$.

Models with embedded thermodynamic and kinetic laws were called conjugated QSPR models and were proposed in our previous paper [165]. In a follow-up study, we proposed a machine learning model that combines ridge regression and a neural network with an equation that relates tautomer acidities with their equilibrium constants. The predictive performance of such conjugated models was shown to be as good as for the individual ones, while the former had some additional benefits like a good prediction of acidities for minor tautomers. Motivated by the above project, here we demonstrate that the Arrhenius equation can be embedded into the ridge regression and neural network algorithms for building QSPR models.

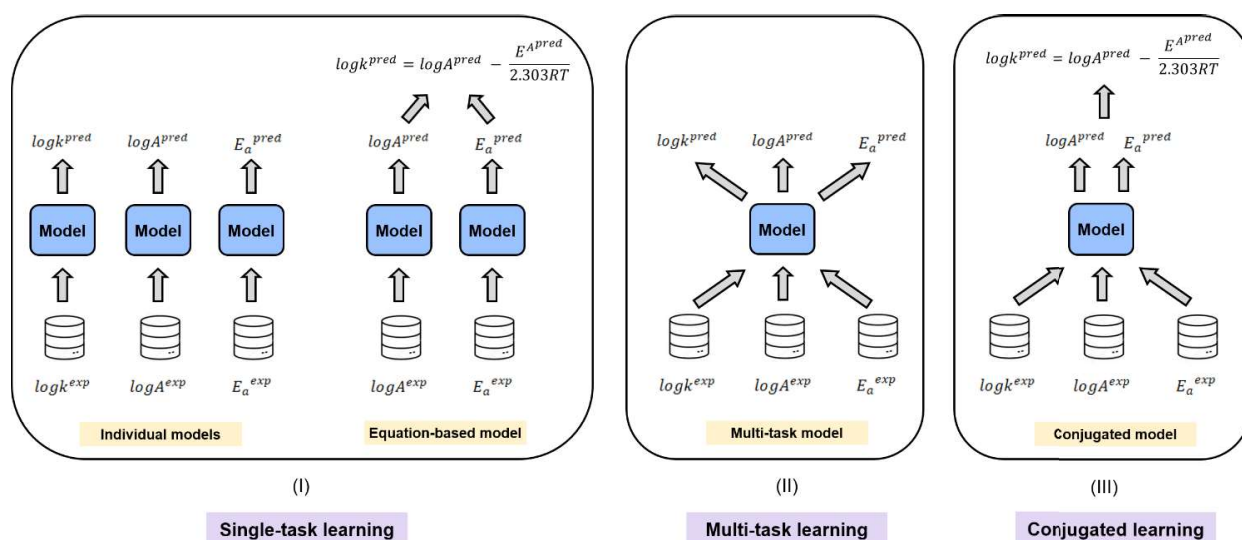


Figure 23. Approaches to modeling kinetic characteristics related by Arrhenius equation. In ordinary single-task learning (I) each characteristic is modeled independently. Multi-task learning (II) performs simultaneous prediction of all three characteristics, whereas conjugated learning (III) embeds the strict mathematical relationship relating the kinetics characteristics (Arrhenius equation) into the machine learning algorithm.

We used the dataset from our previous study [166] to build *individual* (*single-task*), *equation-based* (*Arrhenius-based*), *multi-task*, and *conjugated* models for predicting $\log k$, $\log A$ and E_A of cycloaddition reactions. *Individual* models were built independently for each kinetic characteristic (Figure 23, I). The *Arrhenius-based* model uses the Arrhenius equation to calculate the $\log k$ with $\log A$ and E_A predicted by individual models (Figure 23, I). The multi-task approach (Figure 23, II) uses all available data across the different reaction characteristics and models them cooperatively in contrast to single-task learning. Multi-task learning can improve the prediction accuracy of modeled characteristics when tasks correlate or share some information. Conjugated learning (Figure 23, III) uses all available data on multiple tasks, but, in contrast to the multi-task approach, explicitly embeds a mathematical equation (in this study it is the Arrhenius equation) relating the tasks to the machine learning algorithm. This approach ensures that the predicted reaction characteristics satisfy the fundamental chemical laws and empowers the conjugated QSPR models with new capabilities.

Design of conjugated learning algorithms

Ridge regression individual models

Ridge regression (RR) is a popular machine learning algorithm that was extensively used in practice [189]. In ridge regression, the prediction of reaction characteristic y^{pred} is performed by multiplying the reaction descriptors x' by the vector of regression coefficients w :

$$y^{pred} = X'w \quad (17)$$

The regression coefficients w can be calculated using the following expression:

$$w = (X^T X + \lambda I)^{-1} X^T y^{exp} \quad (18)$$

where x is the descriptor matrix of training reactions associated with experimental values y^{exp} of the target characteristic. Hyperparameter λ is a regularization coefficient controlling the complexity of the model. We used ridge regression to independently build three *individual* models for predicting the $\log k$, $\log A$ and E_A of cycloaddition reactions. The regularization coefficient was adjusted using the grid search technique.

Ridge regression conjugated models

In conjugated models, fundamental chemical laws are integrated with machine learning algorithms. In this study, we consider the Arrhenius equation, which can be embedded into the ridge regression algorithm. Let us consider an *equation-based (Arrhenius-based)* model, where the rate constant y_K^{pred} is calculated using the Arrhenius equation applied to the values of $\log A$ and E_A predicted by *individual* QSPR models:

$$\log k = \log A - \frac{E_A}{2.303RT} \Rightarrow y_K^{pred} = X_K w_A - T X_K w_E \quad (19)$$

where T is the diagonal matrix with the elements that are calculated as:

$$\frac{1}{2.303RT_i} \quad (20)$$

and T_i is the temperature of the i -th reaction. On the other hand, if experimental data on $\log k$ are available, the Arrhenius equation can be integrated with ridge regression using a special quadratic loss function:

$$E_K(w_A, w_E) = \|y_K^{exp} - y_K^{pred}\|^2 = \|y_K^{exp} - X_K w_A + T X_K w_E\|^2 \quad (21)$$

In the case of $E_K(w_A, w_E)$, there are two sets of regression coefficients, w_A (for predicting $\log A$) and w_E (for predicting E_A), which can be optimized to predict the $\log k$. To enable correct prediction of $\log A$ and the E_A , loss function $E_K(w_A, w_E)$ can be combined with individual quadratic loss functions for the $\log A$ and E_A and regularization terms:

$$E_A(w_A) = \|y_A^{exp} - y_A^{pred}\|^2 = \|y_A^{exp} - X_A w_A\|^2 + \lambda_A w_A^T w_A \quad (22)$$

$$E_E(w_E) = \|y_E^{exp} - y_E^{pred}\|^2 = \|y_E^{exp} - X_E w_E\|^2 + \lambda_E w_E^T w_E \quad (23)$$

resulting in a conjugated model loss function:

$$E(w_A, w_E) = c_K E_K(w_A, w_E) + c_A E_A(w_A) + c_E E_E(w_E) + \lambda_A w_A^T w_A + \lambda_E w_E^T w_E \quad (24)$$

where c_K, c_A, c_E are trade-off coefficients that control the contribution of each type of the loss function to conjugated loss $E(w_A, w_E)$, λ_A and λ_E are regularization coefficients. After differentiation of the loss function $E(w_A, w_E)$, the optimal regression weights w_A and w_E can be calculated using the following analytical expressions:

$$w_A = (I - BD)^{-1}(A + BC) \quad (25)$$

$$w_E = (I - DB)^{-1}(C + DA) \quad (26)$$

where matrices A, B, C, D are obtained as follows:

$$\begin{aligned} A &= (c_K X_K^T X_K + c_A X_A^T X_A + \lambda_A I)^{-1} (c_K X_K^T y_K + c_A X_A^T y_A) \\ B &= (c_K X_K^T X_K + c_A X_A^T X_A + \lambda_A I)^{-1} (c_K X_K^T T X_K) \\ C &= (c_K X_K^T T^T T X_K + c_E X_E^T X_E + \lambda_E I)^{-1} (c_E X_E^T y_E - c_K X_K^T T y_K) \\ D &= (c_K X_K^T T^T T X_K + c_E X_E^T X_E + \lambda_E I)^{-1} (c_K X_K^T T X_K) \end{aligned} \quad (27)$$

As a result, regression coefficients w_A and w_E in the *conjugated* model are estimated using the training sets of $\log k (X_K)$, $\log A (X_A)$ and $E_A (X_E)$ data.

Neural network individual, multi-task and conjugated models

Individual, *multi-task*, and *conjugated* models can be built using neural networks (NN). In *individual* models, each characteristic is modeled independently using a standard multilayer neural network with one or more hidden layers and one output neuron (Figure 24a). *Multi-task* models can be built using a neural network with three output neurons, each predicting one of the kinetic characteristics (Figure 24b). Such neural network can be trained using the multi-task loss:

$$\text{Multi-task loss} = c_K (\log k^{\text{exp}} - \log k^{\text{pred}})^2 + c_A (\log A^{\text{exp}} - \log A^{\text{pred}})^2 + c_E (E_a^{\text{exp}} - E_a^{\text{pred}})^2 \quad (28)$$

where c_K, c_A, c_E are coefficients that control the contribution of each type of error to the multi-task loss.

The *conjugated* models can be built using the neural networks shown in Figure 24c. This neural network has two output neurons. The first output neuron predicts $\log A$ and the second one predicts E_A (Figure 24c). The predicted values of $\log A$ and E_A are then used to calculate the prediction of $\log k$ using the Arrhenius equation. Finally, the obtained predicted values of $\log k$, $\log A$ and E_A are used to calculate the conjugated loss:

$$\text{Conjugated loss} = c_K \left(\log k^{\text{exp}} - \left(\log A^{\text{pred}} - \frac{E_a^{\text{pred}}}{2.303RT} \right) \right)^2 + c_A (\log A^{\text{exp}} - \log A^{\text{pred}})^2 + c_E (E_a^{\text{exp}} - E_a^{\text{pred}})^2 \quad (29)$$

Individual, *multi-task*, and *conjugated* NN models discussed hereafter had one hidden layer with 256 neurons. Neural network weights were optimized using a gradient descent algorithm at a learning rate of 0.001. The complexity of the *individual* and *conjugated* NN models was controlled by the weight decay parameter (L2 regularization), which took values from 10^{-3} to 10^1 . Neural networks were implemented using the PyTorch package [167].

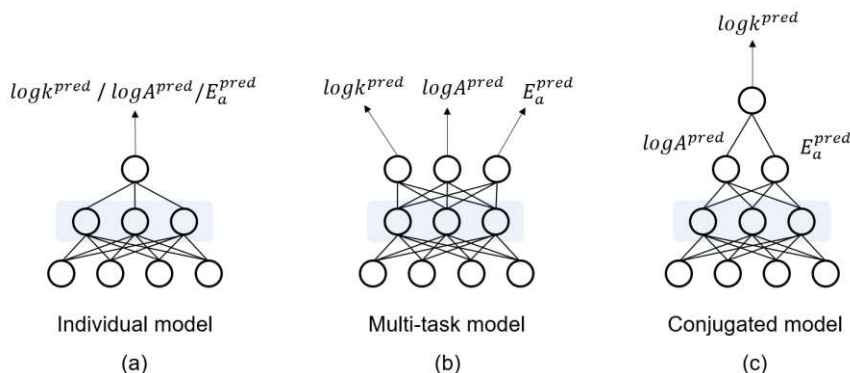


Figure 24. Neural network architectures for building an *individual* (a), *multi-task* (b), and *conjugated* (c) model for prediction of the kinetic characteristics related by the Arrhenius equation.

Computational details

Data

The data on cycloaddition reactions were taken from our previous paper [166]. The dataset includes 1849 reactions with 1849 experimental values of $\log k$, 1236 experimental values of $\log A$, and 1350 experimental values of E_a (kJ/mol). The rate constants $\log k$ were measured in different solvents and at different temperatures T . The dataset contains Diels-Alder (4+2) cycloaddition, (3+2) dipolar cyclization, and (2+2) cycloadditions. Within the 1849 reactions, there are 763 unique structural transformations (Table 3).

The dataset was divided into training and test sets (in the proportion of 90/10) so that the test set contained structural transformations which did not occur in the training set (Table 3). As a result, the test set contained 73 unique structural transformations that were not represented in the training set, which consisted of 690 unique structural transformations (Table 3). The training set was used to build the *individual*, *Arrhenius-based*, *multi-task*, and *conjugated* models, while the test set was used to evaluate the predictive performance of the models.

Table 3. Description of the training and test set on cycloaddition reactions.

	# reactions	# unique structural transformations	# kinetic characteristics		
			$\log k$	$\log A$	E_A
Training set	1478	690	1478	1008	1120
Test set	371	73	371	228	230

Descriptors

Each cycloaddition reaction was transformed into the corresponding Condensed Graph of Reaction (CGR) (Figure 25) [153] generated using the CGRtools package [152]. A CGR is derived from the superposition of products and reactants and contains both conventional chemical bonds (single, double, triple, aromatic, etc.) and so-called “dynamic” bonds describing chemical transformations, i.e., breaking or forming a bond or changing bond order.

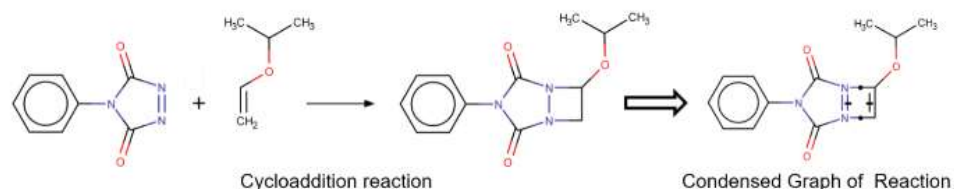


Figure 25. A cycloaddition reaction from the dataset and the corresponding CGR describing the structural transformation. The formed bonds are denoted with a circle, while the broken ones are crossed.

All generated CGRs were processed using the ISIDA tool [161,162] to calculate fragment descriptors by counting the occurrence of particular subgraphs (structural fragments) of different topologies and sizes. We tested different types of fragment descriptors and selected atom-centered descriptors with a radius from 2 to 5. The total number of fragment descriptors was 3733. The vector of fragment descriptors for each reaction was concatenated with the vector of solvent descriptors, which included 14 descriptors describing such properties of solvent as polarity, polarizability, Catalan constants SPP, SA, SB, Kamlet-Taft constants α , β , π^* , dielectric constants, function of the refractive index. These descriptors were successfully applied in our previous publications [163–166].

To build *individual* and *multi-task* models, the fragment/solvent descriptor matrices were concatenated with the temperature descriptor. In *conjugated* models, only fragment and solvent descriptors were used as reaction descriptors, while reaction temperatures were included in the model using the Arrhenius equation. The calculated descriptors constituted three matrices: X_K , X_A and X_E , where the number of rows in each matrix corresponds to the number of experimental values of $\log k$, $\log A$ and E_A for cycloaddition reactions (Table 3).

Model building

The best models were selected with the coefficient of determination (R^2) calculated using the 5-fold transformation-out cross-validation procedure [190] implemented in the in-house CIMtools package (<https://github.com/cimm-kzn/CIMtools>). Transformation-out cross-validation prepares test folds that include structural transformations that are not presented in training folds. This cross-

validation strategy provides an unbiased estimation of the predictive performance of the models for novel types of structural transformations.

Building ridge regression models. *Individual* and *conjugated* RR models were implemented using *PyTorch* tensors [167], which enabled the training of RR models on both CPU and GPU. *Individual* RR models have hyperparameter λ , the regularization coefficient, which controls the model complexity. For *individual* models, we tested values of λ between 10^{-10} to 10^5 and found the optimal value using the grid search technique.

Conjugated RR models have hyperparameters c_K , c_A and c_E that balance the prediction error of the $\log k$, $\log A$ and E_A characteristics. The other two hyperparameters of the *conjugated* model are the regularization coefficients λ_A and λ_E (Figure 26). To optimize the hyperparameters of the RR *conjugated* models, we used the *hyperopt* package [158], which applies advanced optimization algorithms to navigate in the hyperparameters space. The values of coefficients c_K , c_A and c_E were sampled from a continuous space defined between 0 to 1, while the regularization coefficients λ_A and λ_E took discrete values between 10^{-10} to 10^5 (Figure 26). The *hyperopt* algorithm adjusts the hyperparameters by maximizing the value of the objective function which was calculated as an average prediction accuracy of all characteristics: $[R^2(\log k) + R^2(\log A) + R^2(E_A)] / 3$. The *hyperopt* algorithm takes the average accuracy and proposes the next combination of possible optimal hyperparameters (Figure 26).

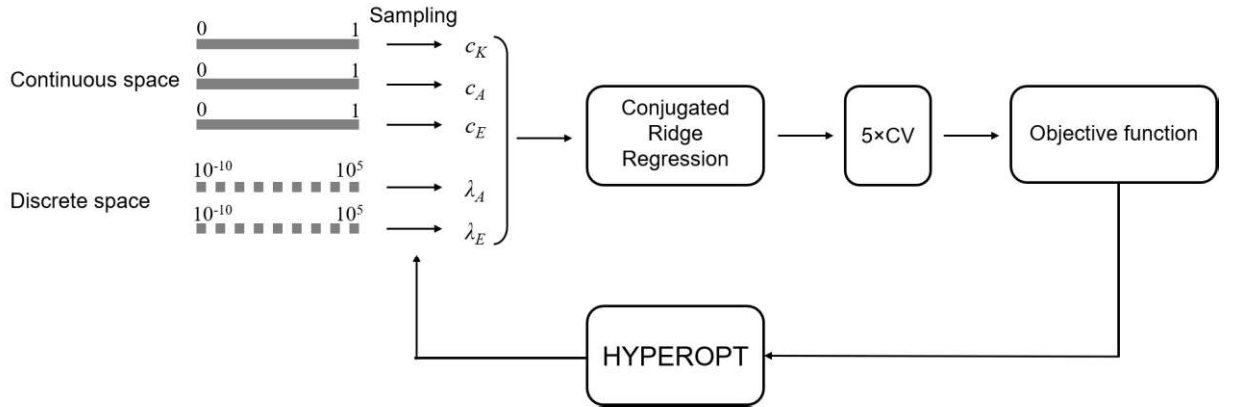


Figure 26. The workflow for optimization of hyperparameters of ridge regression *conjugated* models using *hyperopt* package. The trade-off coefficients were sampled from continuous space defined between 0 to 1. The regularization coefficients λ_A and λ_E took values from discrete 10^{-10} to 10^5 . *Conjugated* models were built with sampled hyperparameters and evaluated using internal 5-fold cross-validation.

Building neural network models. *Individual*, *multi-task*, and *conjugated* NN models were built with the architectures depicted in Figure 24. In NN *multi-task* and *conjugated* models, the coefficients c_K , c_A , and c_E were automatically adjusted together with other neural network weights using the gradient descent algorithm. This means that the trade-off coefficients are learned directly

from the training set, rather than being fixed as hyperparameters before model training as in RR *conjugated* models. This approach to optimization of the trade-off coefficients in the NN *multi-task* and *conjugated* models significantly reduces the computational resources required for model training and hyperparameters optimization.

Results and discussion

Comparison of individual, Arrhenius-based, multi-task, and conjugated models

This section reports the results of the performance comparison of *individual*, *Arrhenius-based*, *multi-task*, and *conjugated* models. The prediction accuracy of the models on the external test set is presented in Table 4. For clarity, we discuss NN models only, whereas the results obtained for RR models are available in Table 2 and share similar trends. We tested two single-task approaches for the prediction of $\log k$: (1) direct modeling of $\log k$, when the *individual* model was built on experimental data on $\log k$ and (2) *Arrhenius-based model* when first individual models for predicting the $\log A$ and E_a were built and then used to calculate the prediction of $\log k$ with the Arrhenius equation. The results demonstrate (Table 4) that the direct predictions of $\log k$ by the *individual model* are more accurate ($R^2_{\text{Test}} = 0.76$) than those calculated with the Arrhenius equation in the *Arrhenius-based model* ($R^2_{\text{Test}} = 0.35$). The prediction accuracy of the *conjugated model* ($R^2_{\text{Test}} = 0.71$) is close to the *individual* ($R^2_{\text{Test}} = 0.76$) and *multi-task* model ($R^2_{\text{Test}} = 0.76$).

Table 4. Predictive performance of individual, Arrhenius-based, multi-task, and conjugated models. RR – Ridge Regression models and NN – Neural Network models.

Model	Training set	Method	R^2 (Test set)		
			$\log k$	$\log A$	E_a
Individual model	$\log k$	RR	0.78	-	-
		NN	0.76	-	-
Individual model	$\log A$	RR	-	0.46	-
		NN	-	0.56	-
Individual model	E_a	RR	-	-	0.91
		NN	-	-	0.90
Arrhenius-based model	$\log A, E_a$	RR	0.27	-	-
		NN	0.35	-	-
Multi-task model	$\log k, \log A, E_a$	NN	0.76	0.48	0.83
Conjugated model	$\log k, \log A, E_a$	RR	0.75	0.57	0.90
		NN	0.71	0.56	0.84

Individual and *Arrhenius-based* models often disagree and provide significantly different predictions of $\log k$ for the same reaction. The assessment of this difference in $\log k$ predictions is illustrated in Figure 27. For demonstration, $\log k$ of each reaction in the test set was predicted by both the *individual* model and the *Arrhenius-based* model, while the difference between the predicted values was calculated as:

$$\Delta \log k^{pred} = \log k^{pred} - \left(\log A^{pred} - \frac{E_a^{pred}}{2.303RT} \right) \quad (30)$$

The *conjugated* model predicts $\log k$, $\log A$ and E_A with similar accuracy as the *individual* models, while the predictions exactly follow the Arrhenius equation (Figure 27b), which is embedded into the conjugated learning algorithm. This feature of *conjugated* models is important because it bridges QSPR models with fundamental chemical laws.

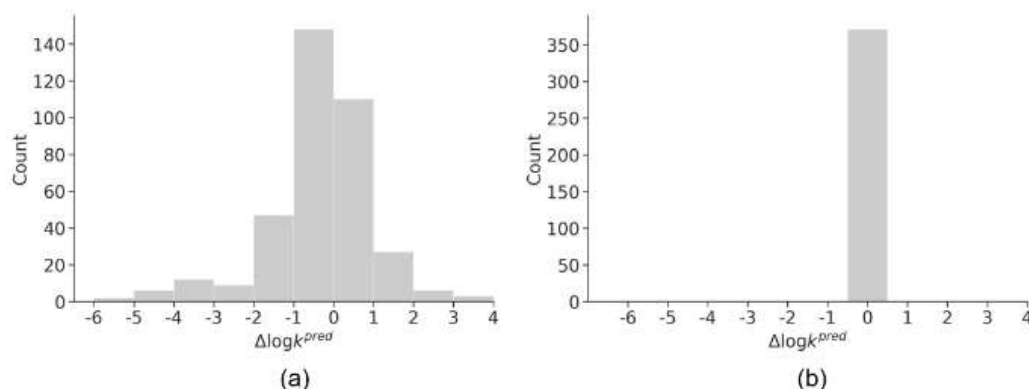


Figure 27. The difference between the $\log k$ values predicted directly and with the Arrhenius-based model according to eq. 15. Two scenarios are considered: predicted values obtained with the individual (a) and conjugated (b) models.

Table 4 demonstrates that the RR and NN conjugated models have similar accuracy. Ridge regression models are easy to build since the optimal regression weights are calculated using analytical expressions. However, more sophisticated optimization of the hyperparameters (trade-off and regularization coefficients) may require a lot of time. On the other hand, the single NN model trains slower than the RR model, but the trade-off coefficients (c_K , c_A and c_E) in the NN model are optimized automatically during model training, which reduces the number of optimized hyperparameters. In addition, the current implementation of RR conjugated models requires a lot of computational resources in the case of large training sets (large sizes of descriptor matrices), while NN models can be trained on large datasets divided into smaller training batches.

Building models with limited data

As follows from Table 4, *individual*, *multi-task*, and *conjugated* models perform similarly if a training set is big enough. We hypothesized that in *multi-task* and *conjugated* models, abundant data for one modeled characteristic (e.g. $\log k$) can compensate for the lack of training data for another characteristic (e.g. $\log A$ or E_A). In contrast to the standard case, we simulated a scenario in which the training sets for the $\log A$ or E_A characteristics were significantly reduced and tested the performance of the models under these conditions. We used the same test set of 371 reactions for the model evaluation (Table 4) but varied the size of the training set. For the sake of clarity, only results for NN models are reported.

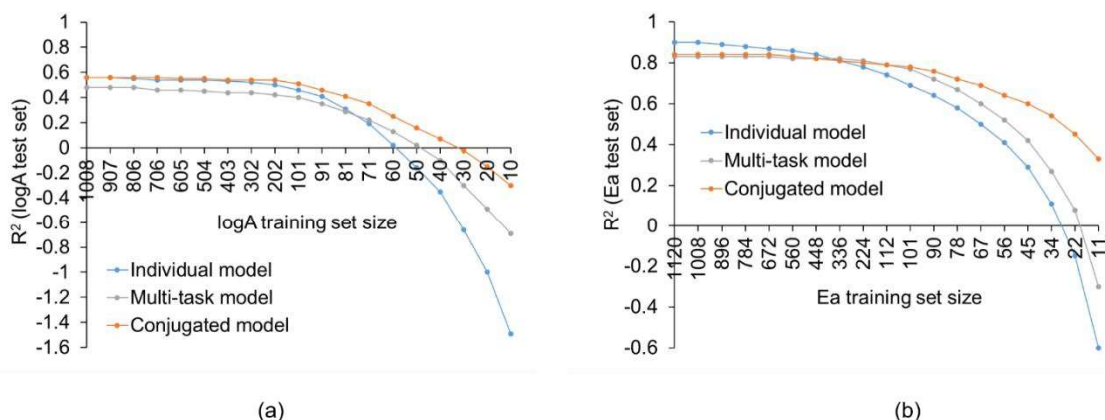


Figure 28. Predictive performance of an individual, multi-task, and conjugated neural network models on test set reactions at different sizes $\log A$ (a) and E_A (b) training sets.

The initial training set contained 1480 experimental values of $\log k$, 1008 values of $\log A$ and 1120 values of E_A . We gradually reduced the number of $\log A$ and E_A training data and evaluated the resulting models on the test set. For this purpose, we randomly selected and removed $N\%$ ($N = 90, 80, 70, 60, 50, 40, 30, 20, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0$) of training reactions associated with $\log A$ and E_A from the initial training set and used reduced training sets to build *individual* $F_{Ind}(\log A^{reduced})$ and $F_{Ind}(E_A^{reduced})$ models. The same reduced training sets on $\log A$ and E_A , as well as all available training data for $\log k$, were used to build the *multi-task* $F_{MT}(\log k, \log A^{reduced}, E_A^{reduced})$ and *conjugated* $F_{Conj}(\log k, \log A^{reduced}, E_A^{reduced})$ model. The models built on the reduced training sets were then used to predict the $\log A$ and E_A for reactions from the test set.

To alleviate the effect of random reduction of the training sets, the above procedure was repeated 20 times, followed by the averaging of related R^2 values. Figure 28 reports the average R^2 on the test set at different sizes of the training set of $\log A^{reduced}$ and $E_A^{reduced}$. For $\log A$ models built on small training sets, *conjugated* learning has no advantages over *single* and *multi-task learning*. The performance of all models gradually decreases as the $\log A$ and E_A training sets were

reduced until the models lose their predictive power at extremely small training sets < 6% (< 70 training reactions). Notice that conjugated models are more stable toward data shrinkage than other approaches.

Similar behavior is observed in modeling E_A on reduced training sets. When the size of the training set is large (e.g. 1120 training reactions with known E_A , Figure 28b), the *individual* $F_{Ind}(E_A^{reduced})$ ($R^2_{Test} = 0.90$) and *multi-task* model $F_{MT}(\log k, \log A^{reduced}, E_A^{reduced})$ ($R^2_{Test} = 0.83$) demonstrate the accuracy comparable with the *conjugated* model $F_{Conj}(\log k, \log A^{reduced}, E_A^{reduced})$ ($R^2_{Test} = 0.84$). However, for significantly reduced E_A training set (11 training reactions corresponding to 1% of the initial set), the *conjugated* models were still predictive ($R^2_{Test} = 0.33$), whereas the *individual* ($R^2_{Test} = -0.60$) and *multi-task* ($R^2_{Test} = -0.30$) models failed.

Thus, *conjugated* models can correctly predict a target characteristic of reactions even for a few training instances if data on another characteristic related to the target characteristic by a strict mathematical relationship is available.

Modeling the temperature dependence of the reaction rate constant

The dependence of the reaction rate constant on temperature is described by the Arrhenius equation. In the *conjugated* model, the Arrhenius equation is directly embedded into the machine learning algorithm (ridge regression or neural network). In the *Arrhenius-based* model, the $\log k$ is calculated using individual $\log A$ and E_A predictions and Arrhenius equation.

In building *individual* and *multi-task* models, the reaction temperature is a descriptor along with fragment and solvent descriptors. Therefore, the *individual* and *multi-task* model can only capture the statistical relationship between $\log k$ and temperature. In this context, we were interested to examine the models' performance as a function of reaction temperature. For this purpose, we generated a new temperature test set. The initial test set (Table 3) contained 1 reaction in 1,4-dioxane, 3 reactions in chlorobenzene, 4 reactions in benzene, and 53 reactions in toluene (a total of 61 reactions) for which $\log A$ and E_A were experimentally determined. We used the experimental $\log A$ and E_A values of these 61 reactions to calculate new $\log k$ using the Arrhenius equation at hypothetical temperatures, which significantly deviates from the temperature range of the training set. For example, for each cycloaddition reaction in toluene, the $\log k$ was calculated for a list of temperatures that start with the freezing temperature of toluene, change in increments of 5K, and end with the boiling temperature of toluene. Thus, for each cycloaddition reaction in toluene, $\log k$ were calculated at 42 hypothetical temperatures (from freezing to the boiling point of toluene). The same procedure was repeated for reactions in 1,4-dioxane (18 hypothetical temperatures), chlorobenzene (36 hypothetical temperatures), and benzene (15 hypothetical temperatures). As a

result, the temperature test set consisted of 61 reactions associated with 2412 $\log k$ values calculated from the Arrhenius equation for hypothetical temperatures; all remaining reactions with experimental temperatures were included in the training set.

The lists of hypothetical temperatures were used in the $\log k$ predictions by the NN models. In the *conjugated* and *Arrhenius-based* models, the hypothetical temperatures were directly used in predicting the $\log k$, while in the *individual* and *multi-task models*, these temperatures were used as a descriptor. Then, predicted with each model $\log k$ values were compared with $\log k$ for hypothetical temperatures. As a result, the *conjugated* and *Arrhenius-based* models had similar performance with mean RMSE of $\log k$ predictions of 0.24 and 0.29, respectively. However, the *individual* and *multi-task* models demonstrated errors (0.56 and 0.57, respectively) of $\log k$ predictions almost twice as high as the *conjugated* model.

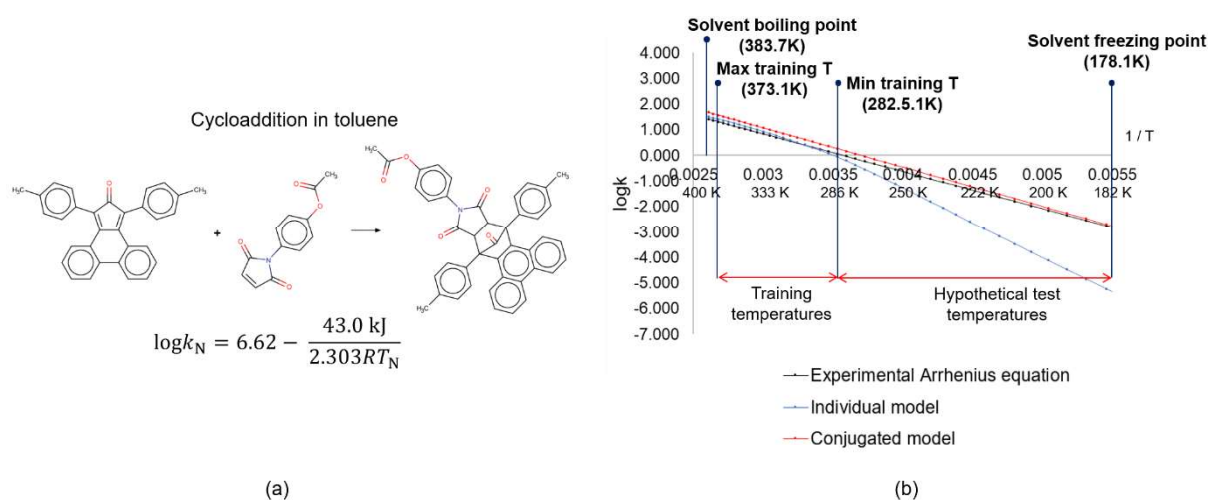


Figure 29. Calculated with experimental Arrhenius equation and predicted $\log k$ with individual and conjugated models for the cycloaddition reaction at different hypothetical temperatures in toluene.

To take a closer look at the reasons for this behavior of the models we extracted one of the test cycloaddition reactions in toluene, for which we plotted the $\log k$ predicted at hypothetical temperatures by the *individual* and the *conjugated* models (Figure 29). We can see (Figure 29) that both models perfectly predict the rate constant at temperatures inside the training temperature range (for all reactions in all solvents). However, in the range beyond the training temperatures, the $\log k$ predicted by the individual model significantly deviate from the experimental ones, while the *conjugated* model predicts the $\log k$ accurately, even at extremely low temperatures close to the freezing point of the solvent. This can be explained by the fact that the *individual* model accounts for only the statistical relationship between the reaction rate constant and the temperature descriptor, whereas the *conjugated* model includes the true relationship in the form of the Arrhenius equation.

Conclusion

In this study, the concept of conjugated learning was applied to model kinetic characteristics related by the Arrhenius equation: rate constant $\log k$, pre-exponential factor $\log A$, and activation energy E_A of cycloaddition reactions. In conjugated QSPR models, the Arrhenius equation was embedded into ridge regression and neural network machine learning algorithms. The conjugated models were compared with individual (single-task) models that were trained independently for each characteristic and multi-task model, where the kinetic characteristics were modeled cooperatively. An equation-based (Arrhenius-based) model was also considered in which the rate constant $\log k$ is calculated using the Arrhenius equation and predicted by individual models $\log A$ and E_A .

It was observed that the individual $\log k$ model is more accurate in predicting the rate constant than the Arrhenius-based model, which calculates $\log k$ using the Arrhenius equation. The predictions of the $\log k$ of individual and Arrhenius-based models often disagree, which demonstrates that the standard QSPR models do not always obey the fundamental chemical laws. However, the conjugated model predicts $\log k$, $\log A$ and E_A with similar accuracy to the individual models, but the predicted characteristics exactly comply with the Arrhenius equation. Furthermore, the conjugated models are more accurate in predicting $\log k$ at the wide range of reaction temperatures. In the individual model, the temperature is treated as a descriptor, whereas in the conjugated models the exact relationship between the rate constant and the temperature is embedded into the model in the form of the Arrhenius equation. To validate the models in new scenarios, a new temperature test set was generated which included $\log k$ values associated with “virtual” temperatures significantly deviating from the temperature range of the training set. It was demonstrated that the individual model cannot correctly predict the values of $\log k$ at temperatures that are significantly different from the training data, while the conjugated model correctly predicts $\log k$ even for the temperatures close to the freezing and boiling points of the solvent.

3.4 Modeling of selectivity constant of competing reactions

Introduction

The ratio of products $\log(E2/S_N2)$ (selectivity constant) of competing for $E2/S_N2$ reactions can be estimated as the difference between the rate constants of the corresponding reactions:

$$\log(E2/S_N2) = \log k_{E2} - \log k_{S_N2} \quad (31)$$

This equation can be used to calculate the prediction of the selectivity constant using the $\log k_{E2}$ and $\log k_{S_N2}$ values predicted by the individual models. On the other hand, in conjugated learning, this equation can be directly integrated with a machine learning algorithm, which allows all three characteristics to be predicted simultaneously.

Conjugated model building. Conjugated models can be built based on ridge regression algorithms and neural networks.

1) Integrate the equation of the main characteristic ($\log k$) by constructing an equation-based loss function E_K .

Individual $\log k$ model:

$$E_{K'}(w) = \|y_K^{exp} - y_{K'}^{pred}\|^2 = \|y_K^{exp} - X_K w\|^2 \quad (32)$$

Equation-based $\log k$ model:

$$\log(E2/S_N2) = \log k_{E2} - \log k_{S_N2} \Rightarrow y_K^{pred} = X_K w_E - X_K w_S \quad (33)$$

$$E_K(w_E, w_S) = \|y_K^{exp} - y_K^{pred}\|^2 = \|y_K^{exp} - X_K w_E + X_K w_S\|^2 \quad (34)$$

2) Combine equation-based loss function E_K with individual loss functions of related characteristics ($\log k_{E2}$ and $\log k_{S_N2}$) and regularization terms of model complexity.

Individual $\log k_{E2}$ model:

$$E_E(w_E) = \|y_E^{exp} - y_E^{pred}\|^2 = \|y_E^{exp} - X_E w_E\|^2 \quad (35)$$

Individual $\log k_{S_N2}$ model:

$$E_S(w_S) = \|y_S^{exp} - y_S^{pred}\|^2 = \|y_S^{exp} - X_S w_S\|^2 \quad (36)$$

Conjugated model:

$$E(w_E, w_S) = aE_K + bE_E + cE_S + \lambda_E \|w_E\|^2 + \lambda_S \|w_S\|^2 \quad (37)$$

where a, b, c are coefficients that control the contribution of each type of loss function into conjugated loss $E(w_E, w_S)$ and λ_E and λ_S are the regularization coefficients.

3) Then derivatives wrt to weights w_E, w_S were calculated and were set equal to 0 in the extremum point. After some mathematical operations one has:

$$\begin{aligned} w_E &= (I - BD)^{-1}(A + BC) \\ w_S &= (I - DB)^{-1}(C + DA) \end{aligned} \quad (38)$$

where matrices A, B, C, D can be obtained as follows:

$$\begin{aligned} A &= (aX_E^T X_E + bX_E^T X_E + \lambda_E I)^{-1}(bX_E^T y_E + aX_E^T y_K) \\ B &= (aX_E^T X_E + bX_E + \lambda_E I)^{-1}aX_E^T X_S \\ C &= (aX_S^T X_S + cX_S^T X_S + \lambda_S I)^{-1}(cX_S^T y_S - aX_S^T y_K) \\ D &= (aX_S^T X_S + cX_S^T X_S + \lambda_S I)^{-1}aX_S^T X_E \end{aligned} \quad (39)$$

Optimal regression weights w_E and w_S (parameters) can also be found by the gradient descent method. Also, conjugated models can be built using special neural networks with conjugated loss functions (Figure 30).

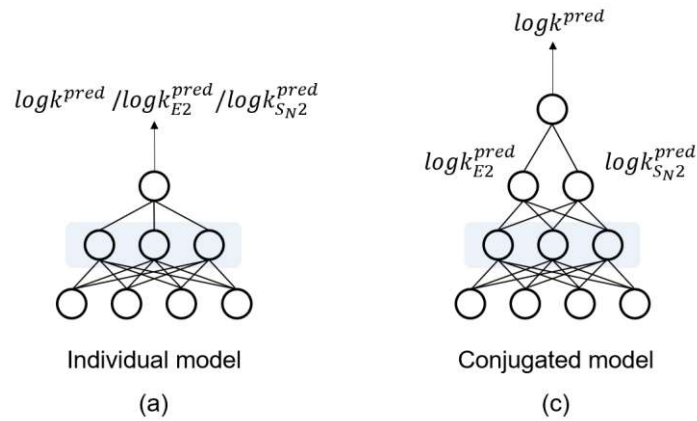


Figure 30. The general architecture of neural networks for building (a) individual and (b) conjugated models for predicting the selectivity constant of competing for $E2/S_N2$ reactions.

Conjugated ridge regression models can be built quickly by calculating optimal weights using matrix equations (38) and (39). However, in the case of large datasets, the standard implementation of conjugated ridge regression can be expensive on memory resources due to the large matrices in equations (39). In this case, conjugated neural networks can be trained on batches of data using gradient descent. Contrary to linear ridge regression, neural networks can capture the nonlinear relationship between reaction descriptors and rate constant.

Model building

Data. There were two types of data to build individual, equation-based, and conjugated models: (i) competing reactions $E2$ and S_N2 (489 reactions) with known reaction rates $\log k_{E2}$ and $\log k_{S_N2}$ and selectivity constants $\log(E2/S_N2)$ for these reactions and (ii) reactions $E2$ (1275 reactions) with known $\log k_{E2}$ (and unknown $\log k_{S_N2}$) and reactions S_N2 (4830 reactions) with known $\log k_{S_N2}$ (and unknown $\log k_{E2}$). In the second type of data the selectivity constant of the competing $E2$ and S_N2 reactions are unknown. A dataset of 489 reactions with known $\log k_{E2}$ was randomly divided into a training and test set in the proportion of 90/10. The second type of data (1275 $E2$ reactions and 4830 S_N2 reactions) were included in the training set.

Descriptors. Each $E2$ and S_N2 reaction was converted into a condensed graph of the reaction, which was encoded with ISIDA fragment descriptors. The total number of descriptors was 1922.

Model optimization. Individual, equation-based and conjugated models were implemented using the PyTorch package, in which matrix operations can be executed using CPUs and GPUs. Regularization and contribution coefficients a , b , c were optimized using the in-house implementation of the genetic algorithm.

Results and discussion

Three types of models for predicting the selectivity constant were compared: the individual model, the Equation-based model, and the conjugated model. The performance of the models is reported in Table 5. The development of neural networks for building conjugated models is part of future research.

The $\log k_{E2}$ the individual model demonstrated moderate performance ($R^2_{\text{Test}} = 0.37$, Figure 31a), while the accuracy of $\log k_{S_N2}$ individual model was unacceptable ($R^2_{\text{Test}} = -0.11$, Figure 32a). Due to the low accuracy of the individual models for $\log k_{E2}$ and $\log k_{S_N2}$, the values of the selectivity constant $\log(E2/S_N2)$ calculated from equation (31) in the equation-based model were

also inaccurate ($R^2_{\text{Test}} = -0.93$). In contrast to equation-based models, an individual model built directly on experimental data on $\log(E2/S_N2)$ provides very accurate predictions ($R^2_{\text{Test}} = 0.89$).

Table 5. Performance (R^2_{Test}) of the individual, equation-based and conjugated models on 49 test reactions.

Approach	Training data	$E2$	S_N2	$\log(E2/S_N2)$
Individual model	$\log k_{E2}$	0.37	-	-
Individual model	$\log k_{S_N2}$	-	-0.11	-
Individual model	$\log(E2/S_N2)$	-	-	0.89
Equation-based model	$\log k_{E2}, \log k_{S_N2}$	0.37	-0.11	-0.93
Conjugated model	$\log k_{E2}, \log k_{S_N2}, \log(E2/S_N2)$	0.60	0.31	0.72

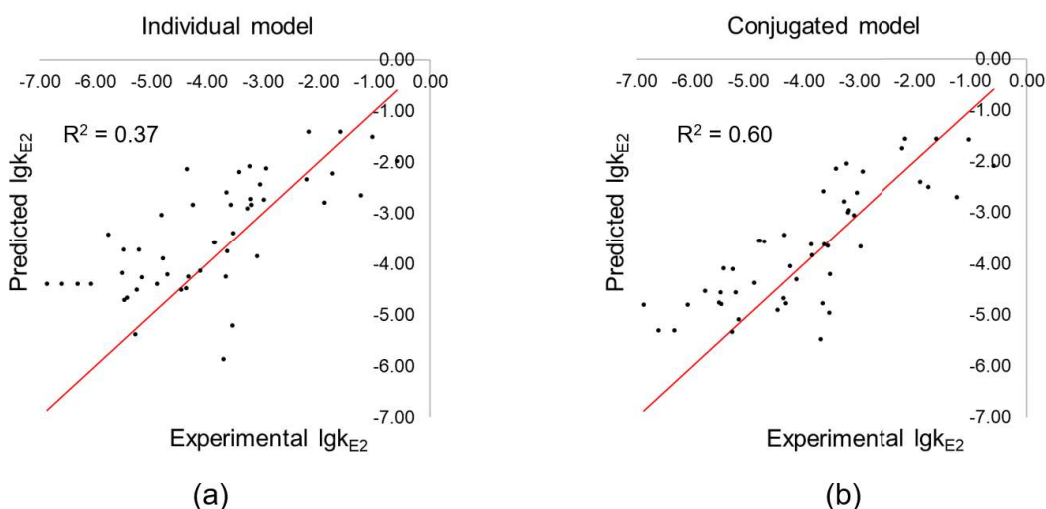


Figure 31. Experimental and predicted values of the rate constant $\log k_{E2}$ for 49 test reactions.

The conjugated model built on the data on $\log k_{E2}$, $\log k_{S_N2}$ and $\log(E2/S_N2)$ significantly improved the accuracy of predictions of $\log k_{E2}$ and $\log k_{S_N2}$ in comparison with individual models ($R^2_{\text{Test}} = 0.37$ vs. 0.60 and $R^2_{\text{Test}} = -0.11$ vs. 0.31) (Figure 31 and Figure 32). However, the prediction accuracy of the $\log(E2/S_N2)$ of the conjugated model is lower than that of the individual model ($R^2_{\text{Test}} = 0.72$ vs. 0.89). For clarity, the experimental and predicted by the individual and conjugated model selectivity constants $\log(E2/S_N2)$ were converted to $E2$ reaction yield and plotted in Figure 33.

Thus, the conjugated ridge regression algorithm increases the prediction accuracy of the rate constants of $E2$ and S_N2 reactions compared with independently constructed individual models.

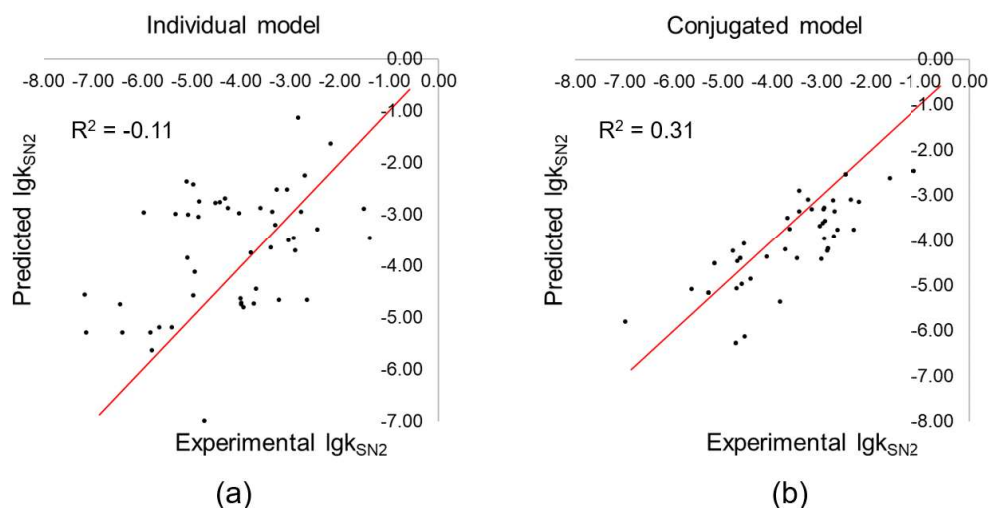


Figure 32. Experimental and predicted values of the rate constant $\log k_{SN2}$ for 49 test reactions.

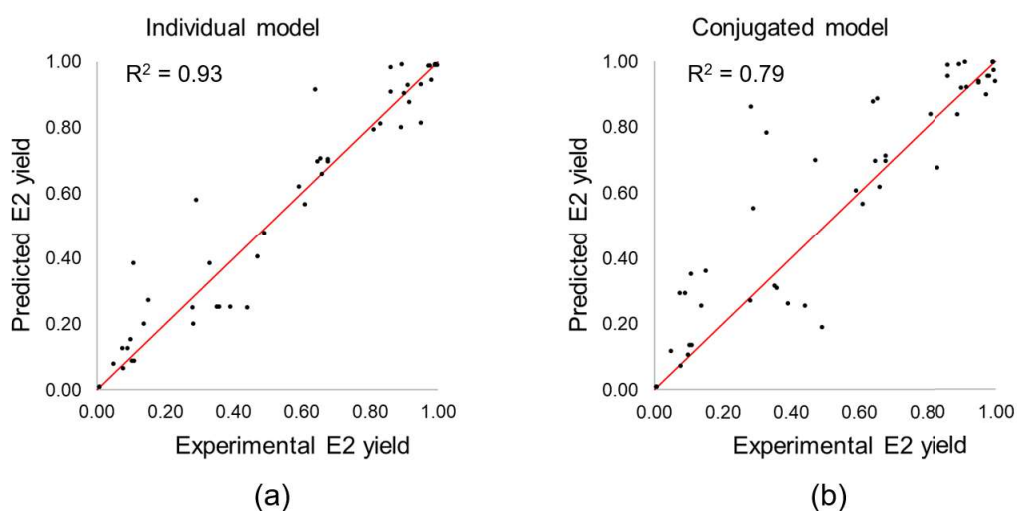


Figure 33. Experimental and predicted values of the yield for 49 E2 test reactions.

Conclusion

In this project, the concept of conjugated learning was applied to model the selectivity constant of competing $E2/S_N2$ reactions. The kinetic equation relating the rate constants of $E2$ and S_N2 reactions were integrated with the ridge regression method. The conjugated models significantly improved the accuracy of $\log k_{E2}$ and $\log k_{SN2}$ predictions compared to the individual models.

Conclusion

1. This project is devoted to the development of advanced machine learning approaches accounting for the complexity of chemical objects (molecules) and processes (reactions). Two approaches and related software tools have been developed: (i) multi-instance machine learning (MIL) considering an ensemble of conformers of each considered molecule, and (ii) conjugated machine learning algorithms accounting for fundamental thermodynamic and kinetic relationships in the modeling of reaction characteristics.

2. A set of multi-instance algorithms, including a naive Wrapper and several multi-instance neural network architectures based on the attention mechanism, dynamic pooling, and gaussian pooling, have been implemented. Various techniques for regularization of instance weights for better identification of key instances have been applied. The developed tools: (i) do not require selection and alignment of conformers, (ii) use only open-source software based on Python 3 packages, and (iii) are fully automated.

3. The MIL-kmeans algorithm for the classification modeling of bioactive compounds has been developed. In this algorithm, each conformer of a given molecule was represented by the 3D *pmapper* descriptors followed by the clustering with the k-means algorithm. The obtained clusters were used to generate a new descriptor vector of a given compound (mapping process) further used in any conventional regression or classification machine learning algorithm.

4. The developed MIL algorithms in combination with the *pmapper* descriptors were applied to the modeling of (i) the bioactivity of compounds from the ChEMBL-23 database and (ii) the enantioselectivity of chiral organic catalysts in asymmetric reactions. The obtained models performed better than related 3D single-conformer models and models involving 2D descriptors.

(i) In a large-scale benchmark on 175 datasets from ChEMBL-23, we have demonstrated that the 3D multi-conformer models approach performed better than 3D single-conformer models built with the lowest-energy conformer and in most cases (>60%) better than the models built on 2D descriptors. In some cases, 2D models completely failed to predict bioactivity whereas 3D multi-conformer models demonstrated a reasonable performance. It has also been demonstrated that the attention-based multi-instance neural network was able to identify bioactive conformers that are similar ($\text{RMSD} < 2\text{\AA}$) to experimental structures extracted from Protein Data Bank.

(ii) The developed 3D modeling approach was applied to the modeling of enantioselectivity in the reaction of asymmetric nucleophilic addition catalyzed by chiral phosphoric acids and phase-

transfer asymmetric alkylation catalyzed by *cinchona* alkaloid-based catalysts. The descriptor vectors resulted from the concatenation of the reaction descriptors generated for Condensed Graphs of Reaction and *pmapper* descriptors encoding the catalyst conformers. Obtained results demonstrated that the 3D multi-conformer models performed similarly or better than the alternative state-of-the-art 2D and 3D approaches reported in the literature.

5. In the conjugated learning approach mathematical equations relating thermodynamic or kinetic characteristics of chemical reactions were used in combination with two different machine learning algorithms - ridge regression and artificial neural networks. The new approach was applied to the modeling of (i) equilibrium constants of tautomerism reactions, (ii) parameters of the Arrhenius equation for cycloaddition reactions, and selectivity constant for competing for E2/S_N2 reactions. In tautomeric equilibria, the conjugated models provide a reasonable estimation of the pK_a of minor tautomers, which can hardly be measured experimentally. In cycloaddition reactions, conjugated models were able to predict the experimentally unreachable rate constant of reactions at extremely low and high temperatures. In some cases, conjugated learning helps to increase the prediction accuracy of the characteristics related by the equation, as demonstrated in the case study of competing E2 and S_N2 reactions.

List of abbreviations

AI - Artificial Intelligence
ML - Machine Learning
SIL - Single-Instance Learning
MIL - Multi-Instance Learning
MIML - Multi-Instance Multi-Label
KID - Key Instance Detection
RF - Random Forest
SVM - Support Vector Machines
LSTM - Long Short-Term Memory
RNN - Recurrent Neural Network
GNN - Graph Neural Network
CNN - Convolutional Neural Network
ILP - Inductive Logic Programming
PPI - Protein-Protein Interactions
III - Isoform-Isoform Interactions
PDB - Protein Data Bank
PBM - Protein Binding Microarray
PBS - Potential Binding Sites
FBS - Functional Binding Sites
TF - Transcription Factor
TFBS - Transcription Factor Binding Sites
MHC - Major Histocompatibility Complex
QM - Quantum Mechanics
MIF - Molecular Interaction Fields
MMFF - Merck Molecular Force Field
DFT – Density Functional Theory
CGR - Condensed Graph of Reaction
ISIDA - In Silico Design and Data Analysis
ECFP - Extended Connectivity Fingerprints
QSAR - Quantitative Structure-Activity Relationship
QSPR - Quantitative Structure-Property Relationship
QSSR - Quantitative Structure–Selectivity Relationship

AUC - Area under the ROC Curve

MAE - Mean Absolute Error

RMSE - Root Mean Square Error

RMSE - Root Mean Square Deviation

API - Application Programming Interface

GPU - Graphics Processing Unit

References

- [1] Y. LeCun, B. Boser, J.S. Denker, D. Henderson, R.E. Howard, W. Hubbard, L.D. Jackel, Backpropagation Applied to Handwritten Zip Code Recognition, *Neural Comput.* 1 (1989) 541–551. <https://doi.org/10.1162/neco.1989.1.4.541>.
- [2] D.E. Rumelhart, G.E. Hinton, R.J. Williams, Learning representations by back-propagating errors, *Nature.* 323 (1986) 533–536. <https://doi.org/10.1038/323533a0>.
- [3] S. Hochreiter, J. Schmidhuber, Long short-term memory, *Neural Comput.* 9 (1997) 1735–1780.
- [4] D. Bahdanau, K.H. Cho, Y. Bengio, Neural machine translation by jointly learning to align and translate, 3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc. (2015).
- [5] T.G. Dietterich, R.H. Lathrop, T. Lozano-Pérez, Solving the multiple instance problem with axis-parallel rectangles, *Artif. Intell.* 89 (1997) 31–71. [https://doi.org/10.1016/s0004-3702\(96\)00034-3](https://doi.org/10.1016/s0004-3702(96)00034-3).
- [6] G. Liu, J. Wu, Z.H. Zhou, Key instance detection in multi-instance learning, in: *J. Mach. Learn. Res.*, 2012: pp. 253–268.
- [7] K. V. Chuang, M.J. Keiser, Attention-Based Learning on Molecular Ensembles, *ArXiv Prepr. ArXiv2011.12820*. (2020). <http://arxiv.org/abs/2011.12820>.
- [8] Z. Zhao, G. Fu, S. Liu, K.M. Elokely, R.J. Doerksen, Y. Chen, D.E. Wilkins, Drug activity prediction using multiple-instance learning via joint instance and feature selection, *BMC Bioinformatics.* 14 (2013). <https://doi.org/10.1186/1471-2105-14-S14-S16>.
- [9] G. Fu, X. Nan, H. Liu, R.Y. Patel, P.R. Daga, Y. Chen, D.E. Wilkins, R.J. Doerksen, Implementation of multiple-instance learning in drug activity prediction., in: *BMC Bioinformatics*, 2012: p. S3. <https://doi.org/10.1186/1471-2105-13-S15-S3>.
- [10] R. Teramoto, H. Kashima, Prediction of protein-ligand binding affinities using multiple instance learning, *J. Mol. Graph. Model.* 29 (2010) 492–497. <https://doi.org/10.1016/j.jmgm.2010.09.006>.
- [11] J. Davis, V.S. Costa, S. Ray, D. Page, An integrated approach to feature invention and model construction for drug activity prediction, in: *ACM Int. Conf. Proceeding Ser.*, 2007: pp. 217–224. <https://doi.org/10.1145/1273496.1273524>.
- [12] D. V. Zankov, M. Matveieva, A. V. Nikonenko, R.I. Nugmanov, I.I. Baskin, A. Varnek, P. Polishchuk, T.I. Madzhidov, QSAR Modeling Based on Conformation Ensembles Using a

- Multi-Instance Learning Approach, *J. Chem. Inf. Model.* 61 (2021) 4913–4923. <https://doi.org/10.1021/acs.jcim.1c00692>.
- [13] D. V. Zankov, M.D. Shevelev, A. V. Nikonenko, P.G. Polishchuk, A.I. Rakhimbekova, T.I. Madzhidov, Multi-instance learning for structure-activity modeling for molecular properties, in: W.M.P. van der Aalst, V. Batagelj, D.I. Ignatov, M. Khachay, V. Kuskova, A. Kutuzov, S.O. Kuznetsov, I.A. Lomazova, N. Loukachevitch, A. Napoli, P.M. Pardalos, M. Pelillo, A. V. Savchenko, E. Tutubalina (Eds.), *Commun. Comput. Inf. Sci.*, 8th International Conference Analysis of Images, Social networks, and Texts, Kazan, 2020: pp. 62–71. https://doi.org/10.1007/978-3-030-39575-9_7.
- [14] A. Nikonenko, D. Zankov, I. Baskin, T. Madzhidov, P. Polishchuk, Multiple Conformer Descriptors for QSAR Modeling, *Mol. Inform.* 40 (2021) minf.202060030. <https://doi.org/10.1002/minf.202060030>.
- [15] D. Zankov, P. Polishchuk, T. Madzhidov, A. Varnek, Multi-Instance Learning Approach to Predictive Modeling of Catalysts Enantioselectivity, *Synlett.* 32 (2021) 1833–1836. <https://doi.org/10.1055/a-1553-0427>.
- [16] J. Xiong, Z. Li, G. Wan, Z. Fu, F. Zhong, T. Xu, X. Liu, Z. Huang, X. Liu, K. Chen, H. Jiang, M. Zheng, Multi-instance learning of graph neural networks for aqueous pKa prediction, *Bioinformatics.* 38 (2022) 792–798. <https://doi.org/10.1093/bioinformatics/btab714>.
- [17] G. Yu, J. Zeng, J. Wang, H. Zhang, X. Zhang, M. Guo, Imbalance deep multi-instance learning for predicting isoform–isoform interactions, *Int. J. Intell. Syst.* 36 (2021) 2797–2824. <https://doi.org/10.1002/int.22402>.
- [18] J. Cheng, K. Bendjama, K. Rittner, B. Malone, BERTMHC: improved MHC–peptide class II interaction prediction with transformer and multiple instance learning, *Bioinformatics.* 37 (2021) 4172–4179. <https://doi.org/10.1093/bioinformatics/btab422>.
- [19] Z. Gao, J. Ruan, A structure-based Multiple-Instance Learning approach to predicting in vitro transcription factor–DNA interaction, *BMC Genomics.* 16 (2015) S3. <https://doi.org/10.1186/1471-2164-16-S4-S3>.
- [20] Y.P. Zhang, Y. Zha, X. Li, S. Zhao, X. Du, Using the multi-instance learning method to predict protein–protein interactions with domain information, in: *Lect. Notes Comput. Sci.* (Including Subser. *Lect. Notes Artif. Intell. Lect. Notes Bioinformatics*), 2014: pp. 249–259. https://doi.org/10.1007/978-3-319-11740-9_24.

- [21] Y. Xu, C. Luo, M. Qian, X. Huang, S. Zhu, MHC2MIL: A novel multiple instance learning based method for MHC-II peptide binding prediction by considering peptide Flanking Region and residue positions, *BMC Genomics*. 15 (2014) S9. <https://doi.org/10.1186/1471-2164-15-S9-S9>.
- [22] J.S. Wu, S.J. Huang, Z.H. Zhou, Genome-wide protein function prediction through multi-instance multi-label learning, *IEEE/ACM Trans. Comput. Biol. Bioinforma.* 11 (2014) 891–902. <https://doi.org/10.1109/TCBB.2014.2323058>.
- [23] Y. Zhang, Y. Chen, W. Bao, Y. Cao, A Hybrid Deep Neural Network for the Prediction of In-Vivo Protein-DNA Binding by Combining Multiple-Instance Learning, in: *Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, 2021: pp. 374–384. https://doi.org/10.1007/978-3-030-84532-2_34.
- [24] A. Emamjomeh, D. Choobineh, B. Hajieghrari, N. MahdiNezhad, A. Khodavirdipour, DNA–protein interaction: identification, prediction and data analysis, *Mol. Biol. Rep.* 46 (2019) 3571–3596. <https://doi.org/10.1007/s11033-019-04763-1>.
- [25] J. Zeng, G. Yu, J. Wang, M. Guo, X. Zhang, DMIL-III: Isoform-isoform interaction prediction using deep multi-instance learning method, in: *Proc. - 2019 IEEE Int. Conf. Bioinforma. Biomed. BIBM 2019*, 2019: pp. 171–176. <https://doi.org/10.1109/BIBM47256.2019.8982956>.
- [26] X. Pan, H. Bin Shen, Predicting RNA-protein binding sites and motifs through combining local and global deep convolutional neural networks, *Bioinformatics*. 34 (2018) 3427–3436. <https://doi.org/10.1093/bioinformatics/bty364>.
- [27] Z. Gao, J. Ruan, Computational modeling of in vivo and in vitro protein-DNA interactions by multiple instance learning, *Bioinformatics*. 33 (2017) 2097–2105. <https://doi.org/10.1093/bioinformatics/btx115>.
- [28] W.A. Abbasi, A. Asif, S. Andleeb, F. ul A.A. Minhas, CaMELS: In silico prediction of calmodulin binding proteins and their binding sites, *Proteins Struct. Funct. Bioinforma.* 85 (2017) 1724–1740. <https://doi.org/10.1002/prot.25330>.
- [29] B. Alipanahi, A. DeLong, M.T. Weirauch, B.J. Frey, Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning, *Nat. Biotechnol.* 33 (2015) 831–838. <https://doi.org/10.1038/nbt.3300>.
- [30] S. Bandyopadhyay, D. Ghosh, R. Mitra, Z. Zhao, MBSTAR: Multiple instance learning for

- predicting specific functional binding sites in microRNA targets, *Sci. Rep.* 5 (2015) 1–12. <https://doi.org/10.1038/srep08004>.
- [31] B.G. Buchanan, E.A. Feigenbaum, Dendral and meta-dendral: Their applications dimension, *Artif. Intell.* 11 (1978) 5–24. [https://doi.org/10.1016/0004-3702\(78\)90010-3](https://doi.org/10.1016/0004-3702(78)90010-3).
 - [32] K. Aikawa, Phoneme recognition using time-warping neural networks, *J. Acoust. Soc. Japan.* 13 (1992) 395–402. <https://doi.org/10.1250/ast.13.395>.
 - [33] D.E. Rumelhart, A Self-Organizing Integrated Segmentation and Recognition Neural Net, *Aerosp. Sensing*, 1992. 4 (1991) 496–503.
 - [34] A.N. Jain, T.G. Dietterich, R.H. Lathrop, D. Chapman, R.E. Critchlow, B.E. Bauer, T.A. Webster, T. Lozano-Perez, Compass: A shape-based machine learning tool for drug design, *J. Comput. Aided. Mol. Des.* 8 (1994) 635–652. <https://doi.org/10.1007/BF00124012>.
 - [35] H.G. Rammensee, T. Friede, S. Stevanović, MHC ligands and peptide motifs: first listing, *Immunogenetics.* 41 (1995) 178–228. <https://doi.org/10.1007/BF00172063>.
 - [36] A. Tibo, M. Jaeger, P. Frasconi, Learning and interpreting multi-multi-instance learning networks, *J. Mach. Learn. Res.* 21 (2020) 191–193. <http://arxiv.org/abs/1810.11514>.
 - [37] Z.H. Zhou, M.L. Zhang, Multi-instance multi-label learning with application to scene classification, in: *Adv. Neural Inf. Process. Syst.*, 2007: pp. 1609–1616. <https://doi.org/10.7551/mitpress/7503.003.0206>.
 - [38] H.-P. Kriegel, A. Pryakhin, M. Schubert, An EM-approach for clustering multi-instance objects, in: *Pacific-Asia Conf. Knowl. Discov. Data Min.*, 2006: pp. 139–148.
 - [39] C. Bergeron, J. Zaretzki, C. Breneman, K.P. Bennett, Multiple instance ranking, in: *Proc. 25th Int. Conf. Mach. Learn.*, ACM Press, New York, New York, USA, 2008: pp. 48–55. <https://doi.org/10.1145/1390156.1390163>.
 - [40] M.A. Carbonneau, V. Cheplygina, E. Granger, G. Gagnon, Multiple instance learning: A survey of problem characteristics and applications, *Pattern Recognit.* 77 (2018) 329–353. <https://doi.org/10.1016/j.patcog.2017.10.009>.
 - [41] M.-A. Carbonneau, Multiple Instance Learning Under Real-World Conditions, PhD Thesis, Univ. Du Québec. (2017) 1–271. [https://www.etsmtl.ca/getattachment/Unites-de-recherche/LIVIA/Recherche-et-innovation/Theses/multiple-instance-learning\(3\).pdf](https://www.etsmtl.ca/getattachment/Unites-de-recherche/LIVIA/Recherche-et-innovation/Theses/multiple-instance-learning(3).pdf).
 - [42] F. Herrera, S. Ventura, R. Bello, C. Cornelis, A. Zafra, D. Sánchez-Tarragó, S. Vluymans, Multiple instance learning: Foundations and algorithms, in: *Mult. Instance Learn. Found.*

- Algorithms, Springer, 2016: pp. 1–233. <https://doi.org/10.1007/978-3-319-47759-6>.
- [43] G. Doran, S. Ray, A theoretical and empirical analysis of support vector machine methods for multiple-instance classification, *Mach. Learn.* 97 (2014) 79–102. <https://doi.org/10.1007/s10994-013-5429-5>.
 - [44] J. Amores, Multiple instance classification: Review, taxonomy and comparative study, *Artif. Intell.* 201 (2013) 81–105. <https://doi.org/10.1016/j.artint.2013.06.003>.
 - [45] J. Foulds, E. Frank, A review of multi-instance learning assumptions, *Knowl. Eng. Rev.* 25 (2010) 1–25. <https://doi.org/10.1017/S026988890999035X>.
 - [46] B. Babenko, Multiple instance learning: algorithms and applications, *View Artic. PubMed/NCBI Google Sch.* (2008) 1–19. http://vision.ucsd.edu/~bbabenko/data/bbabenko_re.pdf%5Cnpapers3://publication/uuid/2CDB4FD4-9E25-4F12-826C-E67049137B7C.
 - [47] D.R. Dooley, Q. Zhang, S.A. Goldman, R.A. Amar, Multiple-instance learning of real-valued data, *J. Mach. Learn. Res.* 3 (2003) 651–678.
 - [48] X. Xu, *Statistical Learning in Multiple Instance Problems*, University of Waikato, 2003. <https://hdl.handle.net/10289/2328>.
 - [49] J. Foulds, *Learning instance weights in multi-instance learning*, The University of Waikato, 2008.
 - [50] J. Wang, J.-D. Zucker, Solving Multiple-Instance Problem: A Lazy Learning Approach, *Proc. 17th Int. Conf. Mach. Learn.* (2000) 1119–1125. <http://cogprints.org/2124/>.
 - [51] T. Gärtner, P.A. Flach, A. Kowalczyk, A.J. Smola, Multi-instance kernels, in: *ICML, 2002*: p. 7.
 - [52] V. Cheplygina, D.M.J. Tax, M. Loog, Multiple instance learning with bag dissimilarities, *Pattern Recognit.* 48 (2015) 264–275. <https://doi.org/10.1016/j.patcog.2014.07.022>.
 - [53] O. Maron, T. Lozano-Pérez, A framework for multiple-instance learning, in: *Adv. Neural Inf. Process. Syst.*, 1998: pp. 570–576.
 - [54] Q. Zhang, S.A. Goldman, Em-dd: An improved multiple-instance learning technique, in: *Adv. Neural Inf. Process. Syst.*, 2002: pp. 1073–1080.
 - [55] S. Ray, M. Craven, Supervised versus multiple instance learning: An empirical comparison, in: *ICML 2005 - Proc. 22nd Int. Conf. Mach. Learn.*, 2005: pp. 697–704.

<https://doi.org/10.1145/1102351.1102439>.

- [56] P. Auer, R. Ortner, A boosting approach to multiple instance learning, in: *Lect. Notes Artif. Intell. (Subseries Lect. Notes Comput. Sci., 2004: pp. 63–74.* https://doi.org/10.1007/978-3-540-30115-8_9.
- [57] Y. Chevaleyre, J.D. Zucker, Solving multiple-instance and multiple-part learning problems with decision trees and rule sets. Application to the mutagenesis problem, in: *Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), 2001: pp. 204–214.* https://doi.org/10.1007/3-540-45153-6_20.
- [58] H. Blockeel, D. Page, A. Srinivasan, Multi-instance tree learning, in: *ICML 2005 - Proc. 22nd Int. Conf. Mach. Learn., 2005: pp. 57–64.* <https://doi.org/10.1145/1102351.1102359>.
- [59] L. Bjerring, E. Frank, Beyond trees: Adopting MITI to learn rules and ensemble classifiers for multi-instance data, in: *Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), 2011: pp. 41–50.* https://doi.org/10.1007/978-3-642-25832-9_5.
- [60] C. Leistner, A. Saffari, H. Bischof, MIForests: Multiple-instance learning with randomized trees, in: *Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), 2010: pp. 29–42.* https://doi.org/10.1007/978-3-642-15567-3_3.
- [61] A. Zafra, S. Ventura, G3P-MI: A genetic programming algorithm for multiple instance learning, *Inf. Sci. (Ny)*. 180 (2010) 4496–4513. <https://doi.org/10.1016/j.ins.2010.07.031>.
- [62] M.L. Zhang, A k-nearest neighbor based multi-instance multi-label learning algorithm, in: *Proc. - Int. Conf. Tools with Artif. Intell. ICTAI, 2010: pp. 207–212.* <https://doi.org/10.1109/ICTAI.2010.102>.
- [63] X. Xu, E. Frank, Logistic regression and boosting for labeled bags of instances, in: *Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), 2004: pp. 272–281.* https://doi.org/10.1007/978-3-540-24775-3_35.
- [64] S. Andrews, T. Hofmann, Multiple instance learning via disjunctive programming boosting, *Adv. Neural Inf. Process. Syst.* 16 (2004).
- [65] S. Andrews, I. Tsochantaridis, T. Hofmann, Support Vector Machines for Multiple-instance Learning, in: *Proc. 15th Int. Conf. Neural Inf. Process. Syst., MIT Press, Cambridge, MA, USA, 2002: pp. 577–584.* <http://dl.acm.org/citation.cfm?id=2968618.2968690>.
- [66] Y. Chen, J.Z. Wang, Image categorization by learning and reasoning with regions, *J. Mach.*

- Learn. Res. 5 (2004) 913–939.
- [67] Y. Chen, J. Bi, J.Z. Wang, MILES: Multiple-instance learning via embedded instance selection, *IEEE Trans. Pattern Anal. Mach. Intell.* 28 (2006) 1931–1947. <https://doi.org/10.1109/TPAMI.2006.248>.
 - [68] J. Ramon, L. De Raedt, Multi Instance Neural Networks, *ICML-2000 Work. Attrib. Relational Learn.* (2000) 53–60.
 - [69] Z.-H. Zhou, M.-L. Zhang, *Neural Networks for Multi-Instance Learning*, 2002.
 - [70] M.L. Zhang, Z.H. Zhou, Improve Multi-Instance Neural Networks through Feature Selection, *Neural Process. Lett.* 19 (2004) 1–10. <https://doi.org/10.1023/B:NEPL.0000016836.03614.9f>.
 - [71] M.L. Zhang, Z.H. Zhou, Ensembles of multi-instance neural networks, in: *IFIP Adv. Inf. Commun. Technol.*, 2005: pp. 471–474. https://doi.org/10.1007/0-387-23152-8_58.
 - [72] M.L. Zhang, Z.H. Zhou, Adapting RBF neural networks to multi-instance learning, *Neural Process. Lett.* 23 (2006) 1–26. <https://doi.org/10.1007/s11063-005-2192-z>.
 - [73] M.L. Zhang, Z.H. Zhou, Multi-instance regression algorithm based on neural network, *Ruan Jian Xue Bao/Journal Softw.* 14 (2003) 1238–1242.
 - [74] X. Wang, Y. Yan, P. Tang, X. Bai, W. Liu, Revisiting multiple instance neural networks, *Pattern Recognit.* 74 (2018) 15–24. <https://doi.org/10.1016/j.patcog.2017.08.026>.
 - [75] M. Ilse, J.M. Tomczak, M. Welling, Attention-based deep multiple instance learning, *35th Int. Conf. Mach. Learn. ICML 2018.* 5 (2018) 3376–3391. <http://arxiv.org/abs/1802.04712>.
 - [76] Z.H. Zhou, J.M. Xu, On the relation between multi-instance learning and semi-supervised learning, in: *ACM Int. Conf. Proceeding Ser.*, ACM, New York, NY, USA, 2007: pp. 1167–1174. <https://doi.org/10.1145/1273496.1273643>.
 - [77] M. Tu, J. Huang, X. He, B. Zhou, Multiple instance learning with graph neural networks, (2019). <http://arxiv.org/abs/1906.04881>.
 - [78] A.S. D’avila Garcez, G. Zaverucha, Multi-instance learning using recurrent neural networks, *Proc. Int. Jt. Conf. Neural Networks.* (2012) 10–15. <https://doi.org/10.1109/IJCNN.2012.6252784>.
 - [79] K. Wang, J. Oramas, T. Tuytelaars, In Defense of LSTMs for Addressing Multiple Instance Learning Problems, *Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell.*

- Lect. Notes Bioinformatics). 12627 LNCS (2021) 444–460. https://doi.org/10.1007/978-3-030-69544-6_27.
- [80] Y. Yan, X. Wang, J. Fang, W. Liu, J. Huang, J. Zhu, I. Takeuchi, Deep Multi-instance Learning with Dynamic Pooling, in: *Acml*, 2018: pp. 662–677. <https://proceedings.mlr.press/v95/yan18a.html>.
 - [81] S. Sabour, N. Frosst, G.E. Hinton, Dynamic routing between capsules, in: *Adv. Neural Inf. Process. Syst.*, 2017: pp. 3857–3867.
 - [82] J. Lee, Y. Lee, J. Kim, A.R. Kosiorek, S. Choi, Y.W. Teh, Set Transformer: A Framework for Attention-based Permutation-Invariant Neural Networks, *ArXiv Prepr. ArXiv1810.00825*. (2018). <http://arxiv.org/abs/1810.00825> (accessed April 19, 2020).
 - [83] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, *Adv. Neural Inf. Process. Syst. 2017-Decem* (2017) 5999–6009.
 - [84] J. Early, C. Evers, S. Ramchurn, Model Agnostic Interpretability for Multiple Instance Learning, *ArXiv Prepr. ArXiv2201.11701*. (2022). <http://arxiv.org/abs/2201.11701>.
 - [85] D. Wang, J. Li, B. Zhang, Multiple-instance learning via random walk, in: *Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, 2006: pp. 473–484. https://doi.org/10.1007/11871842_45.
 - [86] D. Zhou, B. Schölkopf, T. Hofmann, Semi-supervised learning on directed graphs, *Adv. Neural Inf. Process. Syst.* 17 (2005).
 - [87] M.A. Carbonneau, E. Granger, A.J. Raymond, G. Gagnon, Robust multiple-instance learning ensembles using random subspace instance selection, *Pattern Recognit.* 58 (2016) 83–99. <https://doi.org/10.1016/j.patcog.2016.03.035>.
 - [88] J. Liu, R. Qiao, Y. Li, S. Li, Witness detection in multi-instance regression and its application for age estimation, *Multimed. Tools Appl.* 78 (2019) 33703–33722. <https://doi.org/10.1007/s11042-019-08203-x>.
 - [89] Y.F. Li, J.T. Kwok, I.W. Tsang, Z.H. Zhou, A convex method for locating regions of interest with multi-instance learning, in: *Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, 2009: pp. 15–30. https://doi.org/10.1007/978-3-642-04174-7_2.
 - [90] Z. Lin, M. Feng, C.N. Dos Santos, M. Yu, B. Xiang, B. Zhou, Y. Bengio, A structured self-

- attentive sentence embedding, 5th Int. Conf. Learn. Represent. ICLR 2017 - Conf. Track Proc. (2017).
- [91] X.C. Li, D.C. Zhan, J.Q. Yang, Y. Shi, Deep multiple instance selection, *Sci. China Inf. Sci.* 64 (2021) 1–15. <https://doi.org/10.1007/s11432-020-3117-3>.
 - [92] K. Xu, J.L. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhutdinov, R.S. Zemel, Y. Bengio, Show, attend and tell: Neural image caption generation with visual attention, in: 32nd Int. Conf. Mach. Learn. ICML 2015, 2015: pp. 2048–2057.
 - [93] B. Shin, J. Cho, H. Yu, S. Choi, Sparse network inversion for key instance detection in multiple instance learning, in: *Proc. - Int. Conf. Pattern Recognit.*, 2020: pp. 4083–4090. <https://doi.org/10.1109/ICPR48806.2021.9413230>.
 - [94] J. Kindermann, A. Linden, Inversion of neural networks by gradient descent, *Parallel Comput.* 14 (1990) 277–286. [https://doi.org/10.1016/0167-8191\(90\)90081-J](https://doi.org/10.1016/0167-8191(90)90081-J).
 - [95] M. Looks, M. Herreshoff, D. Hutchins, P. Norvig, T.D. Team, Deep Multiple Instance Learning With Gaussian Weighting, (2020) 1–12.
 - [96] J. Haab, Is Attention Interpretation ? A Quantitative Assessment On Sets, Grenoble 2022, ECML PKDD Int. Work. Explain. Knowl. Discov. Data Min. Sept. 19, 2022, Grenoble, Fr. 1 (2022).
 - [97] J.L. Paulsen, A.C. Anderson, Scoring ensembles of docked protein: ligand interactions for virtual lead optimization, *J. Chem. Inf. Model.* 49 (2009) 2813–2819.
 - [98] F. Milletti, A. Vulpetti, Tautomer preference in PDB complexes and its impact on structure-based drug discovery, *J. Chem. Inf. Model.* 50 (2010) 1062–1074. <https://doi.org/10.1021/ci900501c>.
 - [99] V.H. Masand, D.T. Mahajan, T. Ben Hadda, R.D. Jawarkar, A.M. Alafeefy, V. Rastija, M.A. Ali, Does tautomerism influence the outcome of QSAR modeling?, *Med. Chem. Res.* 23 (2014) 1742–1757. <https://doi.org/10.1007/s00044-013-0776-0>.
 - [100] V.H. Masand, D.T. Mahajan, P. Gramatica, J. Barlow, Tautomerism and multiple modeling enhance the efficacy of QSAR: Antimalarial activity of phosphoramidate and phosphorothioamidate analogues of amiprofos methyl, *Med. Chem. Res.* 23 (2014) 4825–4835. <https://doi.org/10.1007/s00044-014-1043-8>.
 - [101] A.P. Toropova, A.A. Toropov, E. Benfenati, G. Gini, D. Leszczynska, J. Leszczynski, QSAR modeling of anxiolytic activity taking into account the presence of keto- and enol-

- tautomers by balance of correlations with ideal slopes, *Cent. Eur. J. Chem.* 9 (2011) 846–854. <https://doi.org/10.2478/s11532-011-0064-0>.
- [102] H.A. Duarte, S. Carvalho, E.B. Paniago, A.M. Simas, Importance of tautomers in the chemical behavior of tetracyclines, *J. Pharm. Sci.* 88 (1999) 111–120. <https://doi.org/10.1021/js980181r>.
- [103] C.M. Baker, N.J. Kidley, K. Papachristos, M. Hotson, R. Carson, D. Gravestock, M. Pouliot, J. Harrison, A. Dowling, Tautomer Standardization in Chemical Databases: Deriving Business Rules from Quantum Chemistry, *J. Chem. Inf. Model.* 60 (2020) 3781–3791. <https://doi.org/10.1021/acs.jcim.0c00232>.
- [104] Z.H. Zhou, M.L. Zhang, Solving multi-instance problems with classifier ensemble based on constructive clustering, *Knowl. Inf. Syst.* 11 (2007) 155–170. <https://doi.org/10.1007/s10115-006-0029-3>.
- [105] W.S. Jen, J.J.M. Wiener, D.W.C. MacMillan, New strategies for organic catalysis: The first enantioselective organocatalytic 1,3-dipolar cycloaddition [20], *J. Am. Chem. Soc.* 122 (2000) 9874–9875. <https://doi.org/10.1021/ja005517p>.
- [106] B. List, R.A. Lerner, C.F. Barbas, Proline-catalyzed direct asymmetric aldol reactions [13], *J. Am. Chem. Soc.* 122 (2000) 2395–2396. <https://doi.org/10.1021/ja994280y>.
- [107] Y. Guan, V.M. Ingman, B.J. Rooks, S.E. Wheeler, AARON: An Automated Reaction Optimizer for New Catalysts, *J. Chem. Theory Comput.* 14 (2018) 5249–5261. <https://doi.org/10.1021/acs.jctc.8b00578>.
- [108] K.B. Lipkowitz, M. Pradhan, Computational studies of chiral catalysts: A Comparative Molecular Field Analysis of an asymmetric Diels-Alder reaction with catalysts containing bisoxazoline or phosphinooxazoline ligands, *J. Org. Chem.* 68 (2003) 4648–4656. <https://doi.org/10.1021/jo0267697>.
- [109] M.C. Kozlowski, S.L. Dixon, M. Panda, G. Lauri, Quantum mechanical models correlating structure with selectivity: Predicting the enantioselectivity of β -amino alcohol catalysts in aldehyde alkylation, *J. Am. Chem. Soc.* 125 (2003) 6614–6615. <https://doi.org/10.1021/ja0293195>.
- [110] A.F. Zahrt, J.J. Henle, B.T. Rose, Y. Wang, W.T. Darrow, S.E. Denmark, Prediction of higher-selectivity catalysts by computer-driven workflow and machine learning, *Science* (80-.). 363 (2019). <https://doi.org/10.1126/science.aau5631>.

- [111] J.L. Melville, B.I. Andrews, B. Lygo, J.D. Hirst, Computational screening of combinatorial catalyst libraries, *Chem. Commun.* 4 (2004) 1410–1411. <https://doi.org/10.1039/b402378a>.
- [112] R. Asahara, T. Miyao, Extended Connectivity Fingerprints as a Chemical Reaction Representation for Enantioselective Organophosphorus-Catalyzed Asymmetric Reaction Prediction, *ACS Omega*. 7 (2022) 26952–26964. <https://doi.org/10.1021/acsomega.2c03812>.
- [113] F. Sandfort, F. Strieth-Kalthoff, M. Kühnemund, C. Beecks, F. Glorius, A Structure-Based Platform for Predicting Chemical Reactivity, *Chem.* 6 (2020) 1379–1390. <https://doi.org/10.1016/j.chempr.2020.02.017>.
- [114] C. Bergeron, G. Moore, J. Zaretski, C.M. Breneman, K.P. Bennett, Fast bundle algorithm for multiple-instance learning, *IEEE Trans. Pattern Anal. Mach. Intell.* 34 (2012) 1068–1079. <https://doi.org/10.1109/TPAMI.2011.194>.
- [115] R.P. Sheridan, K.R. Korzekwa, R.A. Torres, M.J. Walker, Empirical regioselectivity models for human cytochromes P450 3A4, 2D6, and 2C9, *J. Med. Chem.* 50 (2007) 3173–3184. <https://doi.org/10.1021/jm0613471>.
- [116] H. Yamakawa, K. Maruhashi, Y. Nakao, Predicting types of protein-protein interactions using a multiple-instance learning model, *Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*. 4384 LNAI (2007) 42–53. https://doi.org/10.1007/978-3-540-69902-6_5.
- [117] H.D. Li, R. Menon, R. Eksi, A. Guerler, Y. Zhang, G.S. Omenn, Y. Guan, A Network of Splice Isoforms for the Mouse, *Sci. Rep.* 6 (2016) 1–11. <https://doi.org/10.1038/srep24507>.
- [118] N. Pfeifer, O. Kohlbacher, Multiple instance learning allows MHC class II epitope predictions across alleles, in: *Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, 2008: pp. 210–221. https://doi.org/10.1007/978-3-540-87361-7_18.
- [119] Y. El-Manzalawy, D. Dobbs, V. Honavar, Predicting MHC-II binding affinity using multiple instance regression, *IEEE/ACM Trans. Comput. Biol. Bioinforma.* 8 (2011) 1067–1079. <https://doi.org/10.1109/TCBB.2010.94>.
- [120] B. Reynisson, B. Alvarez, S. Paul, B. Peters, M. Nielsen, NetMHCpan-4.1 and NetMHCIIpan-4.0: Improved predictions of MHC antigen presentation by concurrent motif deconvolution and integration of MS MHC eluted ligand data, *Nucleic Acids Res.* 48 (2021)

W449–W454. <https://doi.org/10.1093/NAR/GKAA379>.

- [121] C. Andrews, Y. Xu, M. Kirberger, J.J. Yang, Structural aspects and prediction of calmodulin-binding proteins, *Int. J. Mol. Sci.* 22 (2021) 1–26. <https://doi.org/10.3390/ijms22010308>.
- [122] F.U.A.A. Minhas, A. Ben-Hur, Multiple instance learning of Calmodulin binding sites, *Bioinformatics*. 28 (2012) i416–i422. <https://doi.org/10.1093/bioinformatics/bts416>.
- [123] G. Stolovitzky, D. Monroe, A. Califano, Dialogue on reverse-engineering assessment and methods: The DREAM of high-throughput pathway inference, *Ann. N. Y. Acad. Sci.* 1115 (2007) 1–22. <https://doi.org/10.1196/annals.1407.021>.
- [124] D. Huang, B. Song, J. Wei, J. Su, F. Coenen, J. Meng, Weakly supervised learning of RNA modifications from low-resolution epitranscriptome data, *Bioinformatics*. 37 (2021) I222–I230. <https://doi.org/10.1093/bioinformatics/btab278>.
- [125] I.H. Witten, E. Frank, J. Geller, Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations, *SIGMOD Rec.* 31 (2002) 76–77. <https://doi.org/10.1145/507338.507355>.
- [126] J. Alcalá-Fdez, A. Fernández, J. Luengo, J. Derrac, S. García, L. Sánchez, F. Herrera, KEEL data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework, *J. Mult. Log. Soft Comput.* 17 (2011) 255–287.
- [127] D.M.J. Tax, V. Cheplygina, A Matlab Toolbox for Multiple Instance Learning, Version 0.7. 9 (2015). <http://prlab.tudelft.nl/david-tax/mil.html>.
- [128] J.M. Arrieta, MILpy: Multiple-Instance Learning Python Toolbox, (2016). <https://github.com/jmarrieta/MILpy>.
- [129] S. Riniker, G.A. Landrum, Better Informed Distance Geometry: Using What We Know to Improve Conformation Generation, *J. Chem. Inf. Model.* 55 (2015) 2562–2574. <https://doi.org/10.1021/acs.jcim.5b00654>.
- [130] N.M. O’Boyle, C. Morley, G.R. Hutchison, Pybel: A Python wrapper for the OpenBabel cheminformatics toolkit, *Chem. Cent. J.* 2 (2008) 1–7. <https://doi.org/10.1186/1752-153X-2-5>.
- [131] R. Todeschini, P. Gramatica, The Whim Theory: New 3D Molecular Descriptors for Qsar in Environmental Modelling, *SAR QSAR Environ. Res.* 7 (1997) 89–115. <https://doi.org/10.1080/10629369708039126>.

- [132] V. Consonni, R. Todeschini, M. Pavan, P. Gramatica, Structure/response correlations and similarity/diversity analysis by GETAWAY descriptors. 2. Application of the novel 3D molecular descriptors to QSAR/QSPR studies, *J. Chem. Inf. Comput. Sci.* 42 (2002) 693–705. <https://doi.org/10.1021/ci0155053>.
- [133] J.H. Schuur, P. Selzer, J. Gasteiger, The coding of the three-dimensional structure of molecules by molecular transforms and its application to structure-spectra correlations and studies of biological activity, *J. Chem. Inf. Comput. Sci.* 36 (1996) 334–344. <https://doi.org/10.1021/ci950164c>.
- [134] M.C. Hemmer, V. Steinhauer, J. Gasteiger, Deriving the 3D structure of organic molecules from their infrared spectra, *Vib. Spectrosc.* 19 (1999) 151–164. [https://doi.org/10.1016/s0924-2031\(99\)00014-4](https://doi.org/10.1016/s0924-2031(99)00014-4).
- [135] A. Kutlushina, A. Khakimova, T. Madzhidov, P. Polishchuk, Ligand-based pharmacophore modeling using novel 3D pharmacophore signatures, *Molecules.* 23 (2018) 3094. <https://doi.org/10.3390/molecules23123094>.
- [136] T.I. Madzhidov, A. Rakhimbekova, A. Kutlushina, P. Polishchuk, Probabilistic approach for virtual screening based on multiple pharmacophores, *Molecules.* 25 (2020) 385. <https://doi.org/10.3390/molecules25020385>.
- [137] P. Polishchuk, A. Kutlushina, D. Bashirova, O. Mokshyna, T. Madzhidov, Virtual screening using pharmacophore models retrieved from molecular dynamic simulations, *Int. J. Mol. Sci.* 20 (2019) 5834. <https://doi.org/10.3390/ijms20235834>.
- [138] Y.N. Dauphin, A. Fan, M. Auli, D. Grangier, Language modeling with gated convolutional networks, in: 34th Int. Conf. Mach. Learn. ICML 2017, 2017: pp. 1551–1559.
- [139] J. Zhang, Y. Zhao, H. Li, C. Zong, Attention with sparsity regularization for neural machine translation and summarization, *IEEE/ACM Trans. Audio Speech Lang. Process.* 27 (2019) 507–518. <https://doi.org/10.1109/TASLP.2018.2883740>.
- [140] E. Jang, S. Gu, B. Poole, Categorical reparameterization with gumbel-softmax, 5th Int. Conf. Learn. Represent. ICLR 2017 - Conf. Track Proc. (2017).
- [141] B. Han, X.H. He, Y.Q. Liu, G. He, C. Peng, J.L. Li, Asymmetric organocatalysis: An enabling technology for medicinal chemistry, *Chem. Soc. Rev.* 50 (2021) 1522–1586. <https://doi.org/10.1039/d0cs00196a>.
- [142] J.D. Oslob, B. Åkermark, P. Helquist, P.O. Norrby, Steric influences on the selectivity in

- palladium-catalyzed allylation, *Organometallics*. 16 (1997) 3015–3021. <https://doi.org/10.1021/om9700371>.
- [143] P.J. Goodford, A Computational Procedure for Determining Energetically Favorable Binding Sites on Biologically Important Macromolecules, *J. Med. Chem.* 28 (1985) 849–857. <https://doi.org/10.1021/jm00145a002>.
- [144] J.L. Melville, K.R.J. Lovelock, C. Wilson, B. Allbutt, E.K. Burke, B. Lygo, J.D. Hirst, Exploring phase-transfer catalysis with molecular dynamics and 3D/4D quantitative structure - Selectivity relationships, *J. Chem. Inf. Model.* 45 (2005) 971–981. <https://doi.org/10.1021/ci0500511>.
- [145] J.J. Henle, A.F. Zahrt, B.T. Rose, W.T. Darrow, Y. Wang, S.E. Denmark, Development of a Computer-Guided Workflow for Catalyst Optimization. Descriptor Validation, Subset Selection, and Training Set Analysis, *J. Am. Chem. Soc.* 142 (2020) 11578–11592. <https://doi.org/10.1021/jacs.0c04715>.
- [146] S.E. Denmark, N.D. Gould, L.M. Wolf, A systematic investigation of quaternary ammonium ions as asymmetric phase-transfer catalysts. Synthesis of catalyst libraries and evaluation of catalyst activity, *J. Org. Chem.* 76 (2011) 4260–4336. <https://doi.org/10.1021/jo2005445>.
- [147] M. Pastor, G. Cruciani, I. McLay, S. Pickett, S. Clementi, GRid-INdependent descriptors (GRIND): A novel class of alignment-independent three-dimensional molecular descriptors, *J. Med. Chem.* 43 (2000) 3233–3243. <https://doi.org/10.1021/jm000941m>.
- [148] S. Sciabola, A. Alex, P.D. Higginson, J.C. Mitchell, M.J. Snowden, I. Morao, Theoretical prediction of the enantiomeric excess in asymmetric catalysis. An alignment-independent molecular interaction field based approach, *J. Org. Chem.* 70 (2005) 9025–9027. <https://doi.org/10.1021/jo051496b>.
- [149] M. Hoogenraad, G.M. Klaus, N. Elders, S.M. Hooijschuur, B. McKay, A.A. Smith, E.W.P. Damen, Oxazaborolidine mediated asymmetric ketone reduction: Prediction of enantiomeric excess based on catalyst structure, *Tetrahedron Asymmetry*. 15 (2004) 519–523. <https://doi.org/10.1016/j.tetasy.2003.12.013>.
- [150] A.F. Zahrt, S. V. Athavale, S.E. Denmark, Quantitative Structure-Selectivity Relationships in Enantioselective Catalysis: Past, Present, and Future, *Chem. Rev.* 120 (2020) 1620–1689. <https://doi.org/10.1021/acs.chemrev.9b00425>.

- [151] F. Hoonakker, N. Lachiche, A. Varnek, Condensed Graph of Reaction: Considering a Chemical Reaction as One Single Pseudo Molecule, *Int. J. Artif. Intell. Tools.* 20 (2011) 253–270. <http://dtai.cs.kuleuven.be/ilp-mlg-srl/papers/ILP09-5.pdf>.
- [152] R.I. Nugmanov, R.N. Mukhametgaleev, T. Akhmetshin, T.R. Gimadiev, V.A. Afonina, T.I. Madzhidov, A. Varnek, CGRtools: Python Library for Molecule, Reaction, and Condensed Graph of Reaction Processing, *J. Chem. Inf. Model.* 59 (2019) 2516–2521. <https://doi.org/10.1021/acs.jcim.9b00102>.
- [153] A. Varnek, D. Fourches, F. Hoonakker, V.P. Solov'ev, Substructural fragments: An universal language to encode reactions, molecular and supramolecular structures, *J. Comput. Aided. Mol. Des.* 19 (2005) 693–703. <https://doi.org/10.1007/s10822-005-9008-0>.
- [154] D.H. Smith, R.E. Carhart, R. Venkataraghavan, Atom Pairs as Molecular Features in Structure-Activity Studies: Definition and Applications, *J. Chem. Inf. Comput. Sci.* 25 (1985) 64–73. <https://doi.org/10.1021/ci00046a002>.
- [155] P. Gedeck, B. Rohde, C. Bartels, QSAR - How good is it in practice? Comparison of descriptor sets on an unbiased cross section of corporate data sets, *J. Chem. Inf. Model.* 46 (2006) 1924–1936. <https://doi.org/10.1021/ci050413p>.
- [156] D. Rogers, M. Hahn, Extended-connectivity fingerprints, *J. Chem. Inf. Model.* 50 (2010) 742–754.
- [157] N. Tsuji, P. Sidorov, C. Zhu, Y. Nagata, T. Gimadiev, A. Varnek, B. List, Predicting Highly Enantioselective Catalysts Using Tunable Fragment Descriptors, (2022). <https://chemrxiv.org/engage/chemrxiv/article-details/62e376ed7f3aa6012ffc2e12>.
- [158] J. Bergstra, D. Yamins, D.D. Cox, Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures, in: 30th Int. Conf. Mach. Learn. ICML 2013, 2013: pp. 115–123.
- [159] J. Bergstra, R. Bardenet, Y. Bengio, B. Kégl, Algorithms for hyper-parameter optimization, *Adv. Neural Inf. Process. Syst.* 24 (2011).
- [160] R. Popa, Genetic algorithms in applications, BoD--Books on Demand, 2012.
- [161] G. Marcou, V.P. Solov'ev, D. Horvath, A. Varnek, ISIDA Fragmentor - User Manual, 2017. <http://infochim.u-strasbg.fr/recherche/Download/>.
- [162] A. Varnek, D. Fourches, D. Horvath, O. Klimchuk, C. Gaudin, P. Vayer, V. Solov'ev, F.

- Hoonakker, I. Tetko, G. Marcou, ISIDA - Platform for Virtual Screening Based on Fragment and Pharmacophoric Descriptors, *Curr. Comput. Aided-Drug Des.* 4 (2008) 191–198. <https://doi.org/10.2174/157340908785747465>.
- [163] T. Gimadiev, T. Madzhidov, I. Tetko, R. Nugmanov, I. Casciuc, O. Klimchuk, A. Bodrov, P. Polishchuk, I. Antipin, A. Varnek, Bimolecular Nucleophilic Substitution Reactions: Predictive Models for Rate Constants and Molecular Reaction Pairs Analysis, *Mol. Inform.* 38 (2019) minf.201800104. <https://doi.org/10.1002/minf.201800104>.
- [164] T.I. Madzhidov, T.R. Gimadiev, D.A. Malakhova, R.I. Nugmanov, I.I. Baskin, I.S. Antipin, A.A. Varnek, Structure–reactivity relationship in Diels–Alder reactions obtained using the condensed reaction graph approach, *J. Struct. Chem.* 58 (2017) 650–656. <https://doi.org/10.1134/S0022476617040023>.
- [165] D. V. Zankov, T.I. Madzhidov, A. Rakhimbekova, T.R. Gimadiev, R.I. Nugmanov, M.A. Kazymova, I.I. Baskin, A. Varnek, Conjugated Quantitative Structure-Property Relationship Models: Application to Simultaneous Prediction of Tautomeric Equilibrium Constants and Acidity of Molecules, *J. Chem. Inf. Model.* 59 (2019) 4569–4576. <https://doi.org/10.1021/acs.jcim.9b00722>.
- [166] M. Glavatskikh, T. Madzhidov, D. Horvath, R. Nugmanov, T. Gimadiev, D. Malakhova, G. Marcou, A. Varnek, Predictive Models for Kinetic Parameters of Cycloaddition Reactions, *Mol. Inform.* 38 (2019) e1800077. <https://doi.org/10.1002/minf.201800077>.
- [167] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimselshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, PyTorch: An imperative style, high-performance deep learning library, *Adv. Neural Inf. Process. Syst.* 32 (2019).
- [168] J. Szegezdi, F. Csizmadia, Tautomer generation. pKa based dominance conditions for generating dominant tautomers., in: 234th ACS Natl. Meet. Boston, MA, August 19-23, 2007, Boston, 2007.
- [169] F. Milletti, L. Storch, G. Sfoma, S. Cross, G. Cruciani, Tautomer enumeration and stability prediction for virtual screening on large chemical databases, *J. Chem. Inf. Model.* 49 (2009) 68–75. <https://doi.org/10.1021/ci800340j>.
- [170] T.R. Gimadiev, T.I. Madzhidov, R.I. Nugmanov, I.I. Baskin, I.S. Antipin, A. Varnek, Assessment of tautomer distribution using the condensed reaction graph approach, *J. Comput. Aided. Mol. Des.* 32 (2018) 401–414. <https://doi.org/10.1007/s10822-018-0101-1>

- [171] M. Glavatskikh, T. Madzhidov, I.I. Baskin, D. Horvath, R. Nugmanov, T. Gimadiev, G. Marcou, A. Varnek, Visualization and Analysis of Complex Reaction Data: The Case of Tautomeric Equilibria, *Mol. Inform.* 37 (2018) 1800056. <https://doi.org/10.1002/minf.201800056>.
- [172] W.A. Warr, A Short Review of Chemical Reaction Database Systems, Computer-Aided Synthesis Design, Reaction Prediction, and Synthetic Feasibility, *Mol. Inform.* 33 (2014) 469–476. <https://doi.org/10.1002/minf.201400052>.
- [173] I.I. Baskin, T.I. Madzhidov, I.S. Antipin, A.A. Varnek, Artificial intelligence in synthetic chemistry: achievements and prospects, *Russ. Chem. Rev.* 86 (2017) 1127–1156. <https://doi.org/10.1070/rcr4746>.
- [174] A. Fernández-Ramos, J.A. Miller, S.J. Klippenstein, D.G. Truhlar, Modeling the kinetics of bimolecular reactions, *Chem. Rev.* 106 (2006) 4518–4584. <https://doi.org/10.1021/cr050205w>.
- [175] C.A. Grambow, L. Pattanaik, W.H. Green, Reactants, products, and transition states of elementary chemical reactions based on quantum chemistry, *Sci. Data.* 7 (2020) 137. <https://doi.org/10.1038/s41597-020-0460-4>.
- [176] Y. Zhao, D.G. Truhlar, Density Functionals with Broad Applicability in Chemistry, *Acc. Chem. Res.* 41 (2008) 157–167. <https://doi.org/10.1021/ar700111a>.
- [177] R.A. Friesner, Ab initio quantum chemistry: Methodology and applications, *Proc. Natl. Acad. Sci.* 102 (2005) 6648–6653. <https://doi.org/10.1073/pnas.0408036102>.
- [178] P.-L. Kang, Z.-P. Liu, Reaction prediction via atomistic simulation: from quantum mechanics to machine learning, *IScience.* 24 (2021) 102013. <https://doi.org/10.1016/j.isci.2020.102013>.
- [179] T.I. Madzhidov, A. Rakhimbekova, V.A. Afonina, T.R. Gimadiev, R.N. Mukhametgaleev, R.I. Nugmanov, I.I. Baskin, A. Varnek, Machine learning modelling of chemical reaction characteristics: yesterday, today, tomorrow, *Mendeleev Commun.* 31 (2021) 769–780. <https://doi.org/10.1016/j.mencom.2021.11.003>.
- [180] P.R. Wells, Linear Free Energy Relationships., *Chem. Rev.* 63 (1963) 171–219. <https://doi.org/10.1021/cr60222a005>.
- [181] C. Hansch, A. Leo, R.W. Taft, A survey of Hammett substituent constants and resonance

- and field parameters, *Chem. Rev.* 91 (1991) 165–195. <https://doi.org/10.1021/cr00002a004>.
- [182] K. Jorner, T. Brinck, P.-O.O. Norrby, D. Buttar, Machine learning meets mechanistic modelling for accurate prediction of experimental activation energies, *Chem. Sci.* 12 (2021) 1163–1175. <https://doi.org/10.1039/D0SC04896H>.
- [183] R.I. Nugmanov, T.I. Madzhidov, G.R. Khaliullina, I.I. Baskin, I.S. Antipin, A.A. Varnek, Development of “structure-property” models in nucleophilic substitution reactions involving azides, *J. Struct. Chem.* 55 (2014) 1026–1032. <https://doi.org/10.1134/S0022476614060043>.
- [184] T.I. Madzhidov, P.G. Polishchuk, R.I. Nugmanov, A. V. Bodrov, A.I. Lin, I.I. Baskin, A.A. Varnek, I.S. Antipin, Structure-reactivity relationships in terms of the condensed graphs of reactions, *Russ. J. Org. Chem.* 50 (2014) 459–463. <https://doi.org/10.1134/S1070428014040010>.
- [185] P. Polishchuk, T. Madzhidov, T. Gimadiev, A. Bodrov, R. Nugmanov, A. Varnek, Structure–reactivity modeling using mixture-based representation of chemical reactions, *J. Comput. Aided. Mol. Des.* 31 (2017) 829–839. <https://doi.org/10.1007/s10822-017-0044-3>.
- [186] T.I. Madzhidov, A. V. Bodrov, T.R. Gimadiev, R.I. Nugmanov, I.S. Antipin, A.A. Varnek, Structure–reactivity relationship in bimolecular elimination reactions based on the condensed graph of a reaction, *J. Struct. Chem.* 56 (2015) 1227–1234. <https://doi.org/10.1134/S002247661507001X>.
- [187] A.R. Singh, B.A. Rohr, J.A. Gauthier, J.K. Nørskov, Predicting Chemical Reaction Barriers with a Machine Learning Model, *Catal. Letters.* 149 (2019) 2347–2354. <https://doi.org/10.1007/s10562-019-02705-x>.
- [188] C.A. Grambow, L. Pattanaik, W.H. Green, Deep Learning of Activation Energies, *J. Phys. Chem. Lett.* 11 (2020) 2992–2997. <https://doi.org/10.1021/acs.jpcllett.0c00500>.
- [189] M.H.J. Gruber, Improving efficiency by shrinkage: the James-Stein and ridge regression estimators, Routledge, 2017.
- [190] A. Rakhimbekova, T.N. Akhmetshin, G.I. Minibaeva, R.I. Nugmanov, T.R. Gimadiev, T.I. Madzhidov, I.I. Baskin, A. Varnek, Cross-validation strategies in QSPR modelling of chemical reactions, *SAR QSAR Environ. Res.* 32 (2021) 207–219. <https://doi.org/10.1080/1062936X.2021.1883107>.

Modélisation structure-propriété avec des techniques avancées d'apprentissage automatique

Résumé

Cette thèse est consacrée au développement de techniques avancées d'apprentissage automatique pour la modélisation des propriétés des molécules et des réactions. Le couplage de la méthode d'apprentissage automatique multi-instances (MIL) avec les descripteurs 3D pharmacophoriques a permis de construire des modèles prédictifs prenant en compte l'ensemble des conformations moléculaires. Cette approche 3D ne nécessite pas de sélection et d'alignement de conformères et a été validée dans les études de (i) la bioactivité des composés et (ii) l'énantiosélectivité des catalyseurs organiques chiraux. Dans de nombreux cas, les modèles MIL multi-conformationnelles 3D ont surpassé les approches classiques impliquant des descripteurs 2D populaires. Dans la deuxième partie, un concept d'apprentissage automatique conjugué a été introduit et appliqué à la modélisation des caractéristiques thermodynamiques et cinétiques des réactions chimiques. L'apprentissage automatique conjugué intègre des équations fondamentales avec des algorithmes d'apprentissage automatique, ce qui le distingue de l'apprentissage multitâche traditionnel ne capturant que la relation statistique entre les tâches

Mots-clés : apprentissage multi-instances, modèles conjugués

Résumé en Anglais

This Ph.D. thesis is devoted to the development of advanced machine learning techniques for the modeling of properties of molecules and reactions. Coupling the Multi-Instance machine Learning (MIL) method with the pharmacophoric 3D descriptors enabled the construction of predictive models accounting for an ensemble of molecular conformations. This 3D approach does not require the selection and alignment of conformers and was validated in the case studies of (i) the bioactivity of compounds and (ii) the enantioselectivity of chiral organic catalysts. In many cases, 3D multi-conformation MIL models overperformed classical approaches involving popular 2D descriptors. In the second part, a concept of conjugated machine learning was introduced and applied to the modeling of thermodynamic and kinetic characteristics of reactions. Conjugated machine learning integrates fundamental equations with machine learning algorithms, which distinguishes it from traditional multi-task learning capturing only the statistical relationship between the tasks.

Keywords: multi-instance learning, conjugated machine learning