



МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ

МАТЕРИАЛОВ: СОВРЕМЕННЫЕ ИССЛЕДОВАНИЯ

И ПОДГОТОВКА КАДРОВ

20 – 24 ноября 2023 года

«Обзор применения алгоритмов машинного обучения в
области материаловедения

Кольцов С.Н.

НИУ ВШЭ в Санкт-Петербурге, Санкт-Петербургская школа физико-математических и компьютерных наук,
Департамент информатики

Важность применения ML в области материаловедения

Науку с интенсивным использованием данных можно назвать "**четвертой парадигмой**" научного поиска, в то время как первые три парадигмы - это эксперимент, теория и моделирование. Почему важно и полезно использовать алгоритмы машинного обучения на существующих данных в дополнении к эксперименту, теории и классическому мат. моделированию:

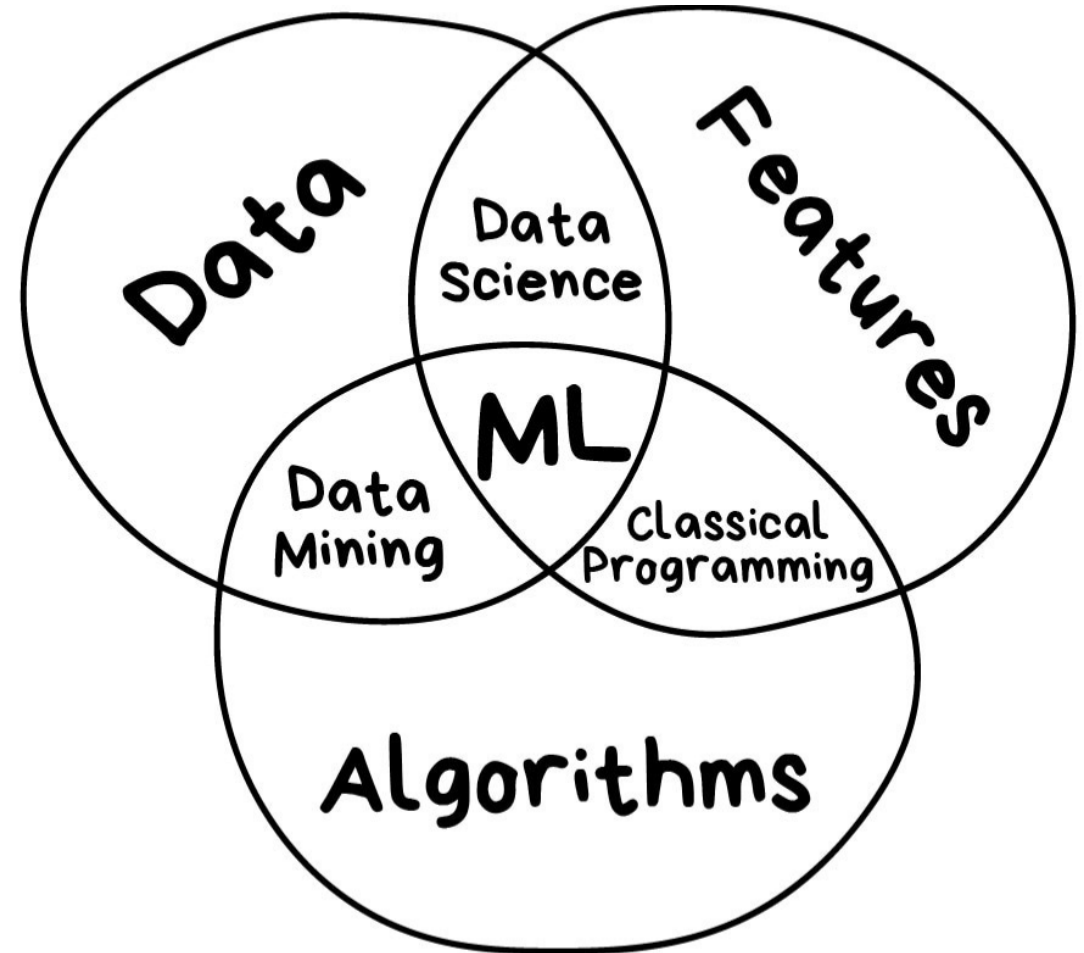
1. Использование машинного обучения в материаловедении может сократить типичный многолетний летний цикл разработки и коммерциализации новых материалов.
2. Упрощения предсказания свойств веществ, сплавов, биологических компонент.
3. Выделение паттернов, характеризующие свойства веществ, что дает дополнительные возможности для теоретического анализа.

Однако, во-первых, получение качественных результатов при помощи алгоритмов машинного обучения существенно зависит от качества исходных данных, на которых производится обучения алгоритмов. **Во-вторых**, необходимо производить тонкую настройку алгоритмов, так как алгоритмы основаны на различных математических предположениях.

Что такое машинное обучение?

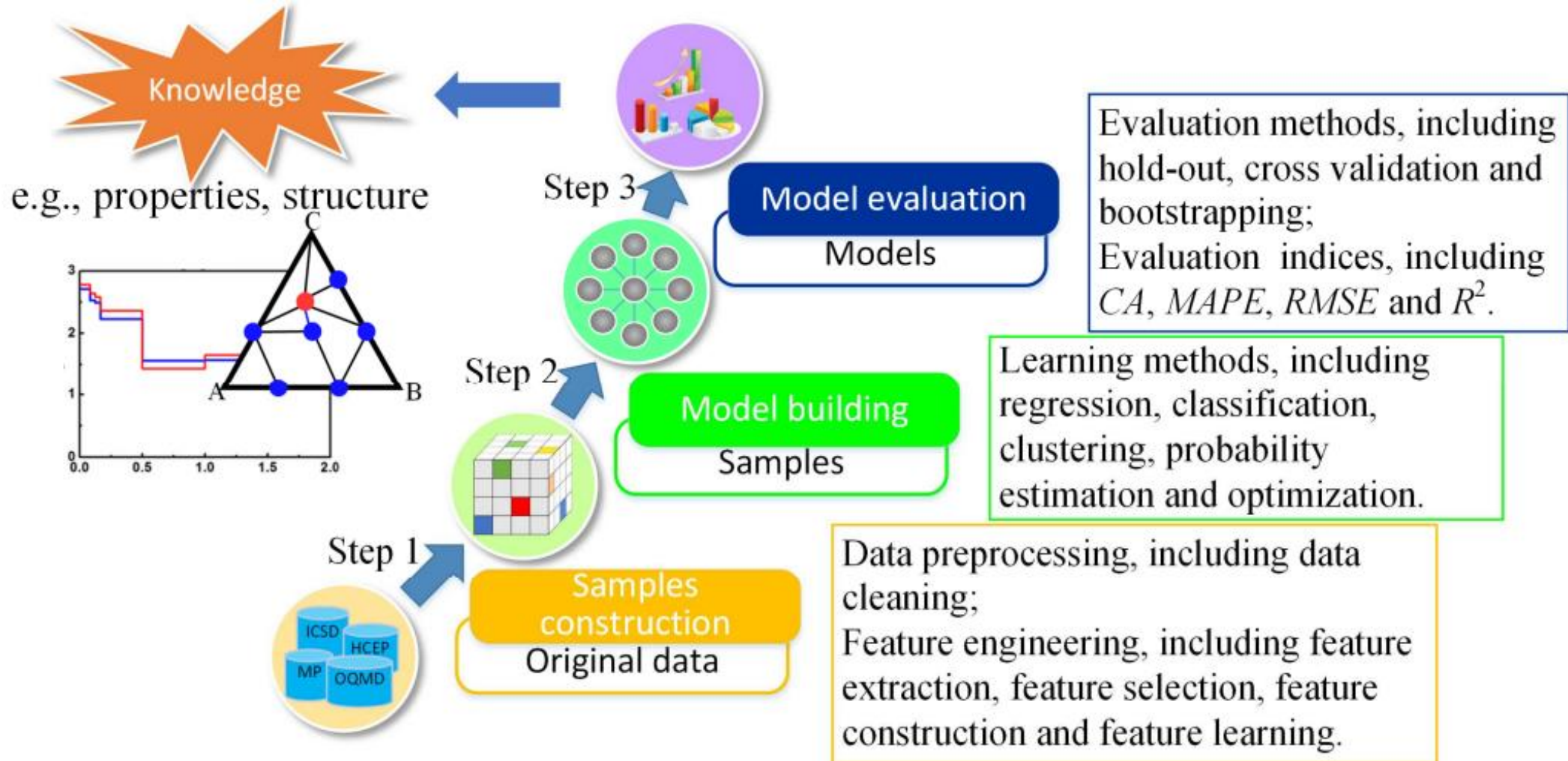
Машинное обучение (ML) - это метод автоматизации построения аналитических/полуаналитических моделей. Используя алгоритмы, которые итеративно обучаются на основе данных, машинное обучение позволяет компьютерам находить скрытые идеи, не будучи явно запрограммированными на то, где их искать. Суть ML - это мощный инструмент поиска закономерностей в высоко размерных данных.

Машинное обучение показывает хорошую применимость в классификации, регрессии и других задачах, связанных с именно высоко размерными данными. Машинное обучение, нацеленное на извлечение знаний и получение информации из массивных баз данных, опирается на результаты предыдущих вычислений (итеративной обучение).





Общий процесс машинного обучения в материаловедении



Step 1. Сбор данных.



Table I. A list of some notable materials-data resources.


Name	URL	Category	Free/Non-Free
3D Materials Atlas	cosmicweb.mse.iastate.edu/wiki/display/home/Materials+Atlas+Home	3D Characterization	Free
AFLOWLIB	aflowlib.org	Computational	Free
AIST Research Information Databases	www.aist.go.jp/aist_e/list/database/riodb	General Materials Data	Free
American Mineralogist Crystal Structure Database	rruff.geo.arizona.edu/AMS/amcsd.php	Minerals	Free
ASM Alloy Center Database	mio.asminternational.org/ac	Alloys	Non-Free
ASM Phase Diagrams	www1.asminternational.org/AsmEnterprise/APD	Thermodynamics	Non-Free
CALPHAD databases (e.g., Thermocalc SGTE)	www.thermocalc.com/products-services/databases/thermodynamic	Thermodynamics	Non-Free
Cambridge Crystallographic Data Centre	www.ccdc.cam.ac.uk/pages/Home.aspx	Crystallography	Non-Free
CatApp	suncat.stanford.edu/catapp	Catalysts	Free
Chemspider	www.chemspider.com	Chemical data	Free
CINDAS High-Performance Alloys Database	cindasdata.com/products/hpad	Alloys	Non-Free
Citration	citration.com	General Materials Data	Free
Computational Materials Repository	cmr.fysik.dtu.dk	Computational	Free

Нет хорошей унифицированной базы данных, но есть масса баз россыпью. У некоторых баз есть API.

Juan, Y., Dai, Y., Yang, Y., & Zhang, J. (2021). *Accelerating materials discovery using machine learning*. *Journal of Materials Science & Technology*, 79, 178–190. doi:10.1016/j.jmst.2020.12.010
40 баз данных



Step 1. Сбор данных.

 aflowlib.org

Aflow - Automatic FLOW for Materials Discovery

Apps and Docs

MendeLIB search



AFLOW database search application

AFLOW-online



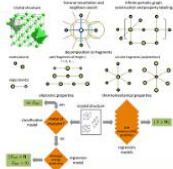
Online interface for AFLOW's symmetry structure comparison, CCE, POCC, and other functionality

Prototype encyclopedia



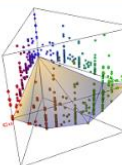
The AFLOW Prototype Encyclopedia with over 1,100 prototypes

AFLOW-ML



Machine Learning application for the PLMF, MDF, and ASC model

AFLOW-CHULL



Convex HULL application for thermodynamic stability and synthesizability

Documentation

AFLOW
REST-API WIKI

Documentation for the AFLOW REST-API and AFLUX

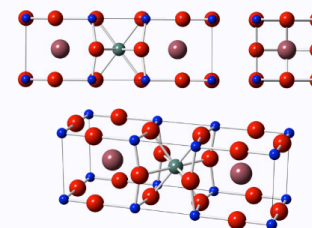


ENCYCLOPEDIA OF CRYSTALLOGRAPHIC PROTOTYPES

M. J. Mehl, D. Hicks, C. Toher, O. Levy, R. M. Hanson, G. L. W. Hart, and S. Curtarolo, The AFLOW Library of Crystallographic Prototypes: Part 1, *Comp. Mat. Sci.* 136, S1-S828 (2017). (doi=10.1016/j.commatsci.2017.01.017)

D. Hicks, M. J. Mehl, E. Gossett, C. Toher, O. Levy, R. M. Hanson, G. L. W. Hart, and S. Curtarolo, The AFLOW Library of Crystallographic Prototypes: Part 2, *Comp. Mat. Sci.* 161, S1-S1011 (2019). (doi=10.1016/j.commatsci.2018.10.043)

D. Hicks, M. J. Mehl, M. Esters, C. Oses, O. Levy, G. L. W. Hart, C. Toher, and S. Curtarolo, The AFLOW Library of Crystallographic Prototypes: Part 3, *Comp. Mat. Sci.* 199, 110450 (2021). (doi=10.1016/j.commatsci.2021.110450)



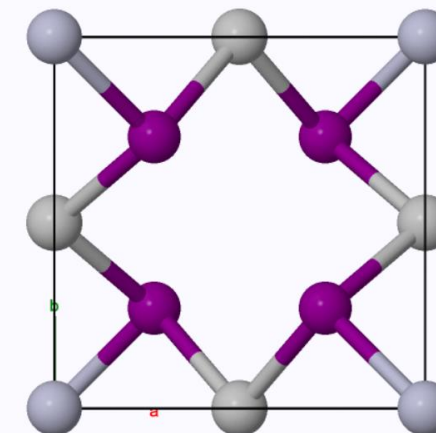
The AFLOW standard encyclopedia of crystallographic prototypes provides a complete description of each structure including formulas for the primitive vectors, basis vectors, and AFLOW commands to

$E3_1$ (β -Ag₂HgI₄) (*obsolete*) Structure : A2BC4_tP7_111_f_a_n

HM: P -4 2 m #111
a=6.340Å
b=6.340Å
c=6.340Å
α=90.000°
β=90.000°
γ=90.000°

● Ag
● Hg
● I

Search by common name or composition...



Построение выборки в виде набора признаков

Информация в базе данных, соответствует конкретным материалам или свойствам материалов. При построении признаков, по которым нужно предсказывать свойство материала идеальным является вариант, когда было выбрано меньшее количество данных при которых ML- модель отработала лучше всего.

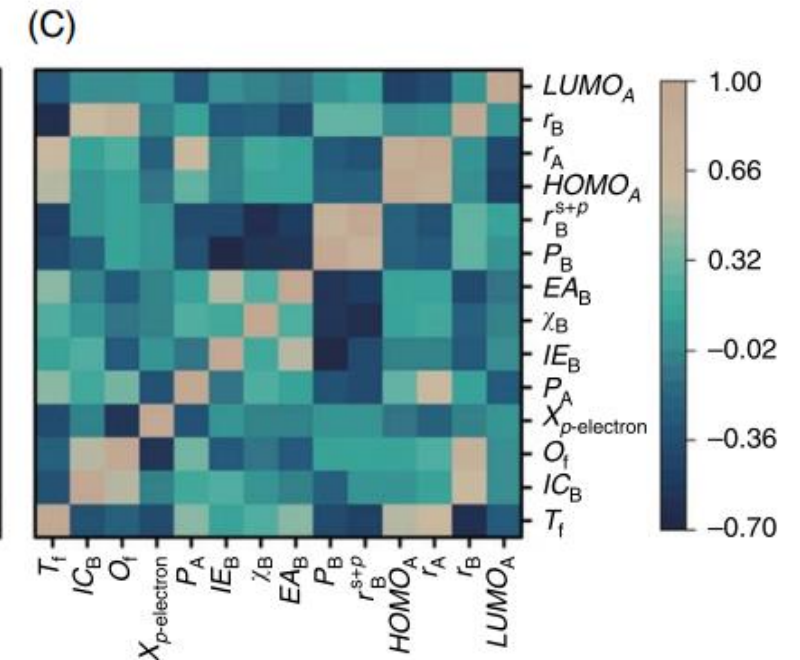
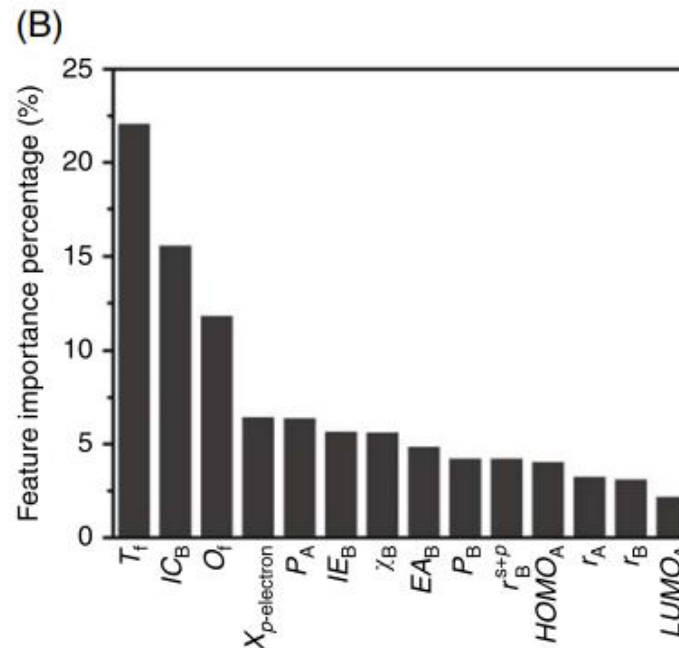
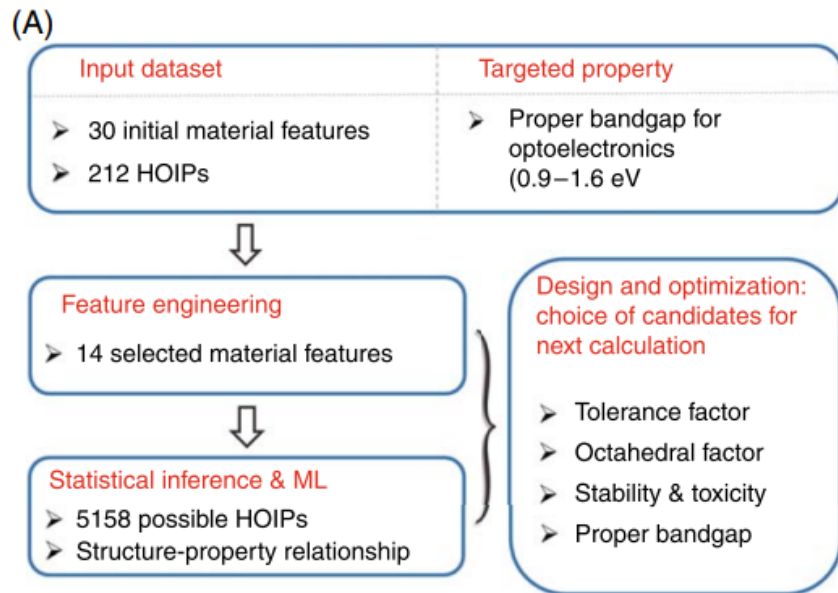
В материаловедении сначала определяются значения характеристик (признаки) материала, затем определяются данные с высокой корреляцией и выбираются в качестве входных образцов ML-моделей, которые не сильно между собой с коррелированы.

На этом этапе решающее значение играет роль как уменьшение размерности (отбор признаков), так и нормализация, для уменьшения эффекта вклада отдельных признаков. Кроме того, необходимо заполнять пропущенные данные. Процедуры заполнения пропущенных данных – отдельная деятельность, зачастую там также используются алгоритмы машинного обучения. Поэтому при формировании списка признаков, чрезвычайно важен опыт людей, непосредственно занятых экспериментальными исследованиями.

	FORMULA	CONDITION: Temperature (K)	PROPERTY: Heat Capacity (J/mol K)
0	B2O3	1400.0	134.306
1	B2O3	1300.0	131.294
2	B2O3	1200.0	128.072
3	B2O3	1100.0	124.516
4	B2O3	1000.0	120.625
5	B2O3	900.0	116.190
6	B2O3	800.0	111.169
7	B2O3	723.0	106.692
8	B2O3	700.0	105.228
9	B2O3	600.0	98.115

Построение выборки в виде набора признаков

Традиционные методы машинного обучения (неглубокое обучение) требуют ручного отбора признаков. Например, Oliyuk et al [1] использовал такие признаки как номер группы элемента В, общее число валентных электронов p , разность радиусов A/B . Другой пример, Lu et al [2] использовали в качестве признаков общий ионный заряд, p -орбитальные электроны и сумма радиусов s - и p -орбиталей, что позволило с помощью модели машинного обучения предсказывать еще не открытые гибридные органо-неорганические перовскиты.



1. Oliyuk AO, Antono E, Sparks TD, et al. High-throughput machine-learning-driven synthesis of full-Heusler compounds. *Chem Mater*. 2016;28:7324-7331
2. Lu S, Zhou Q, Ouyang Y, Guo Y, Li Q, Wang J. Accelerated discovery of stable lead-free hybrid organic-inorganic perovskites via machine learning. *Nat Commun*. 2018;9:3405-3413.

Карта моделей машинного обучения

Модели могут быть поделены (условно) на следующие направления:

Классическое обучение, включает в себя регрессионные модели, модели классификации, которые не включают в себя глубокое обучение. Сюда включены различные виды регрессий (на основе МНК, максимизации логарифма правдоподобия, стохастического градиента), классификаторы: **KNN, SVM, Random Forest Classifier, GradientBoostingClassifier**, Классификация на основе теоремы Байеса: **MultinomialNB, BernoulliNB, ComplementNB, LogisticRegression**.

Модели в виде ансамблей на основе классических моделей, такие как: **StackingClassifier, Voting Ensemble, Adaboost, GradientBoosting, Xboost, CatBoost**

Нейронные сети с глубоким обучением. NN реализованные на различных архитектурах, на базе слоев: 1. Сети прямого распространения (Feed forward neural networks (**Dense layer**)). 2. Рекурсивные нейронные сети (**RNN, LSTM, GRM layers**). 3. Сверточная нейронная сеть — CNN (Convolutional Neural Network, **CN layers**)

Модели типа обучение с подкреплением. Основное отличие обучения с подкреплением (reinforcement learning) от классического машинного обучения заключается в том, что искусственный интеллект обучается в процессе взаимодействия с окружающей средой, а не на исторических данных.



Карта моделей машинного обучения

Модели также могут быть поделены (условно) на следующие направления:

1. Обучение с учителем. В таких моделях (регрессии, классификаторы, нейронные сети), модели обучаются на датасетах с известными ответами. В этом случае, модели обучают предсказывать заданные значения или классы.

2. Модели без учителя. Это модели, в которых производится разделение объектов на группы, где объекты внутри группы похожи друг на друга, а между группами наблюдается максимальное различие. Модели кластеризации (Kmeans, Cmeans, DbSCAN, GMM, topic modeling)

3. Модели, направленные на уменьшение размерности датасета. Это модели которые позволяют отобрать максимально нужные признаки (фичи). Модели: **Method Univariate Selection, Recursive Feature Elimination, Random ForestRegressor feature selection, Principal component analysis(PCA)**).



Карта моделей машинного обучения: Зоопарк нейронные сети Neural Networks

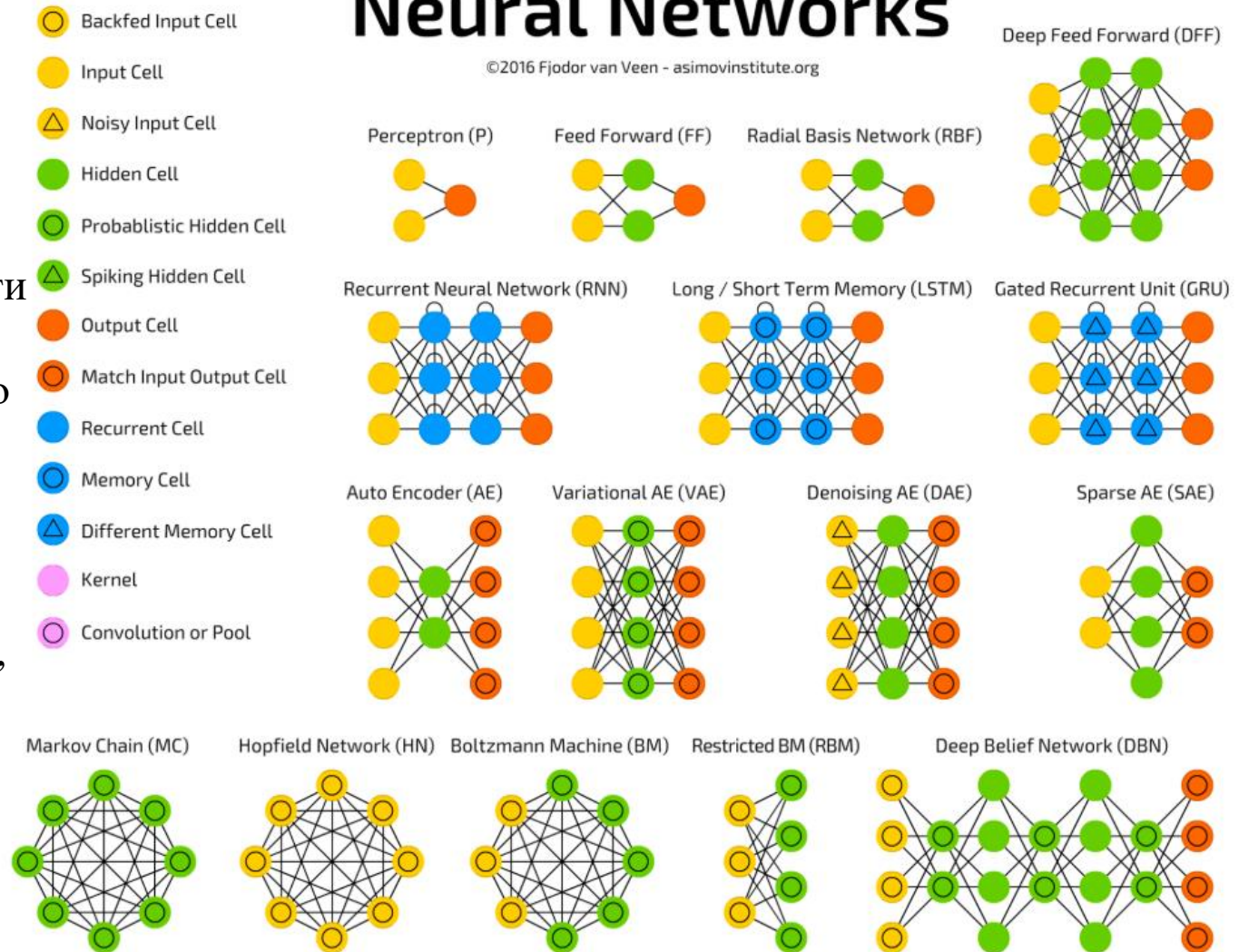
Существует большое количество архитектур NN, реализованных на различных типах слоев:

1. Сети прямого распространения (FFNN).
2. Рекуррентны сети (RNN и ее разновидности).

Часто использовалась в задачах перевода и генерации текста или изображений. — это те же сети прямого распространения, но со смещением во времени: нейроны получают информацию не только от предыдущего слоя, но и от самих себя в результате предыдущего прохода (в таких моделях есть учет внутреннего состояния потока информации).

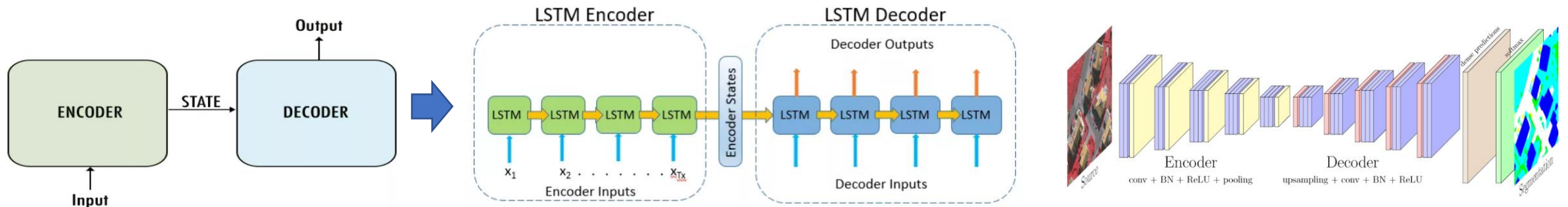
3. Двухнаправленные RNN, LSTM и GRU (BiRNN, BiLSTM и BiGRU). Разница лишь в том, что эти нейросети связаны не только с прошлым, но и с будущим.

4. Ограниченная машина Больцмана (Restricted Boltzmann machine, RBM).



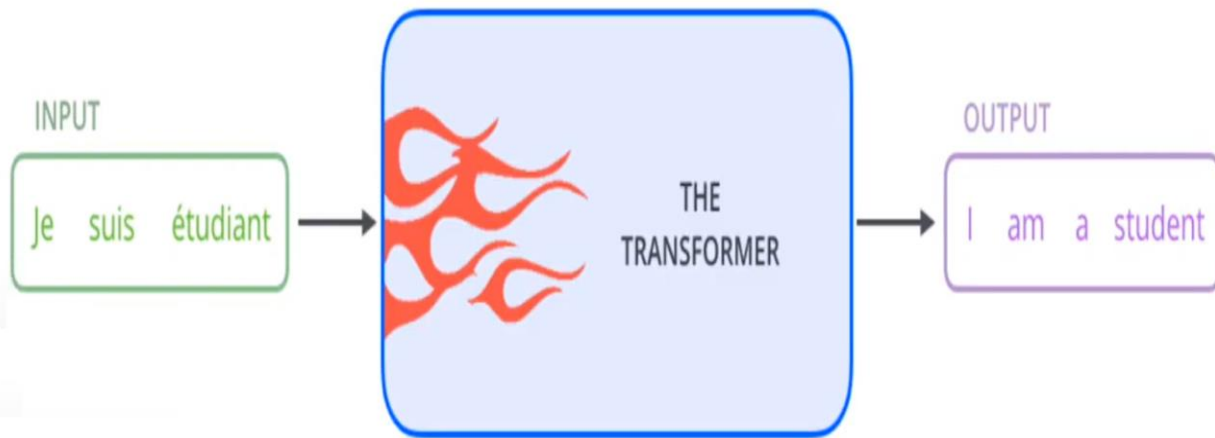
Карта моделей машинного обучения: Зоопарк нейронные сети

5. **Автоэнкодеры (Autoencoders, AE)** — нечто подобное FFNN, Основная идея автоэнкодеров — автоматическое кодирование информации.
6. Архитектура вариационных автоэнкодеров (VAE) такая же, как и у обычных, но обучают их другому — приближительному вероятностному распределению входных образцов. Это в какой-то степени возвращение к истокам, так как VAE немножко ближе к машинам Больцмана.
7. **Шумоподавляющие (помехоустойчивые) автоэнкодеры (Denoising autoencoders, DAE)** — это такое AE, которым на подаем на вход не просто данные, а данные с шумом (например, делая картинку более зернистой).
8. **Свёрточные нейронные сети (convolutional neural networks, CNN)** и глубокие свёрточные нейронные сети (deep convolutional neural networks, DCNN) кардинально отличаются от других сетей. Они используются в основном для обработки изображений, иногда для аудио и других видов входных данных.
9. **Развёртывающие нейронные сети (deconvolutional networks, DN)**, также называемые обратными графическими сетями, — это свёрточные нейронные сети наоборот.

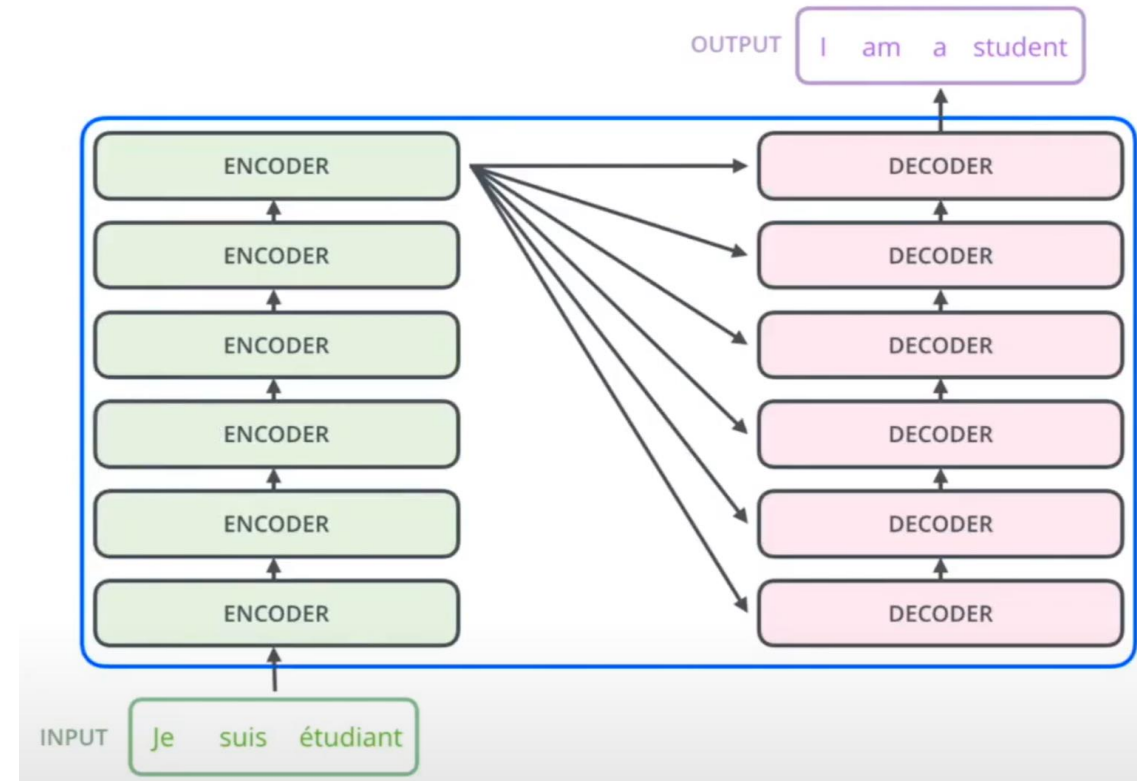


Карта моделей машинного обучения: Transformers

Трансформаторы - это наиболее популярный тип архитектуры нейронных сетей. Трансформеры недавно использовались OpenAI в их языковых моделях, а также недавно использовались DeepMind для AlphaStar - своей программы, направленной на победу над профессиональным игроком в Starcraft. Трансформаторы были разработаны для решения задачи преобразования последовательности или нейронного машинного перевода. Это означает любую задачу, которая преобразует входную последовательность в выходную последовательность. Это включает в себя распознавание речи, преобразование текста в речь и так далее.

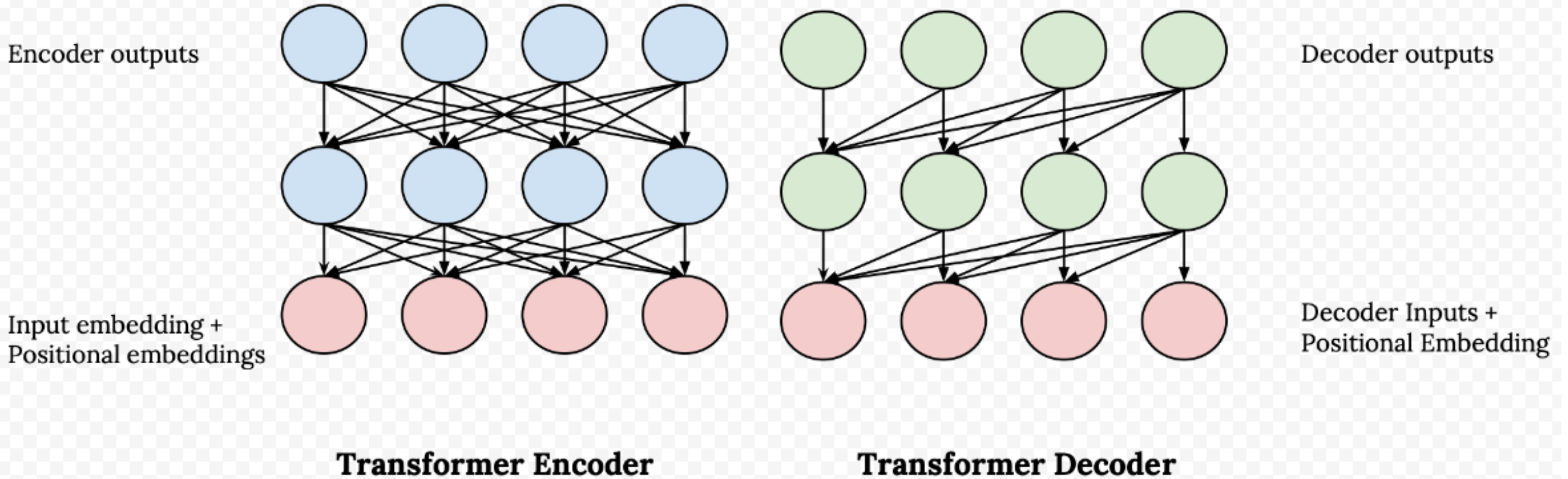


Трансформеры не являются рекуррентными сетями, а построенные на слоях типа self-attention и dense (плотно связанные слои). На данный момент, именно трансформеры задают state of the art больших языковых моделей. Одним из наиболее популярных хранилищ сеток такого типа является Hugging Face.





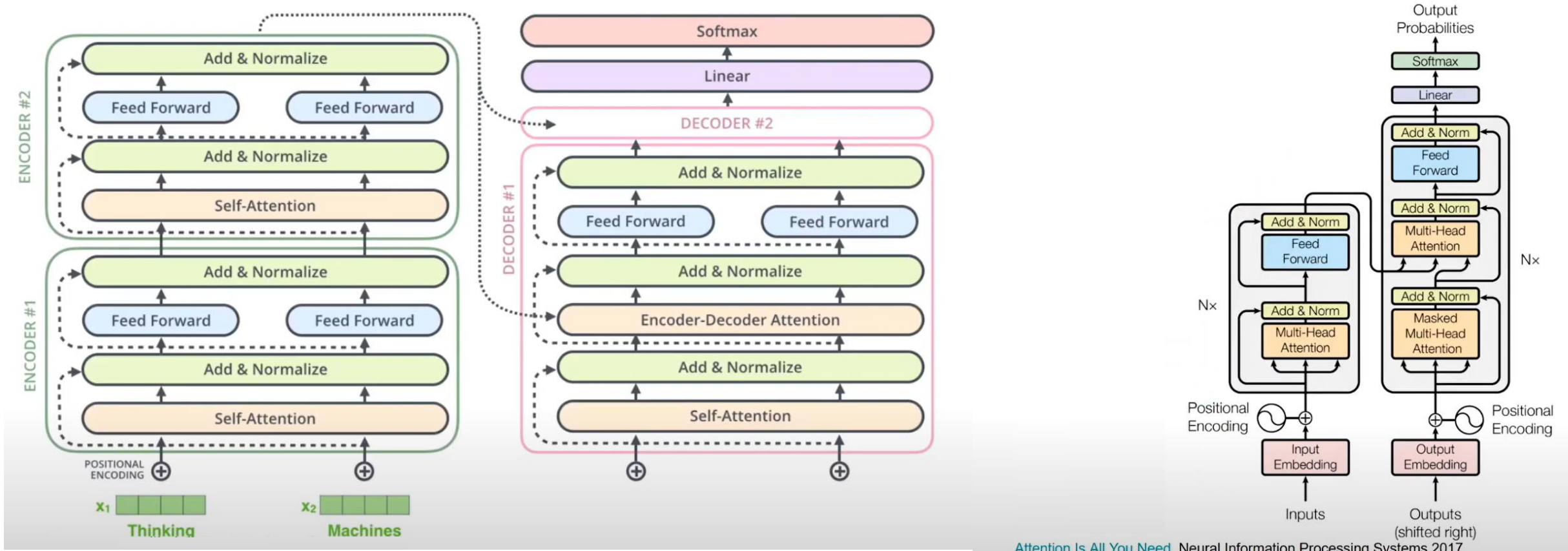
Карта моделей машинного обучения: Transformers



Трансформер это специальная архитектура, которая состоит из слоя эмбединга + слой Multi-head attention layer + LayerNormalization + FFNN. Такая комбинация образует один блок.



Карта моделей машинного обучения: Transformers





Примеры ML Модели в материаловедении

Для каждой пары взяли энергию связи (известную из эксперимента). Таким образом каждая молекула была представлена вектором чисел (Bag – of - Bonds), где каждое число определяет попарное взаимодействие. Далее, соответственно полная энергия была определена в виде суммы:

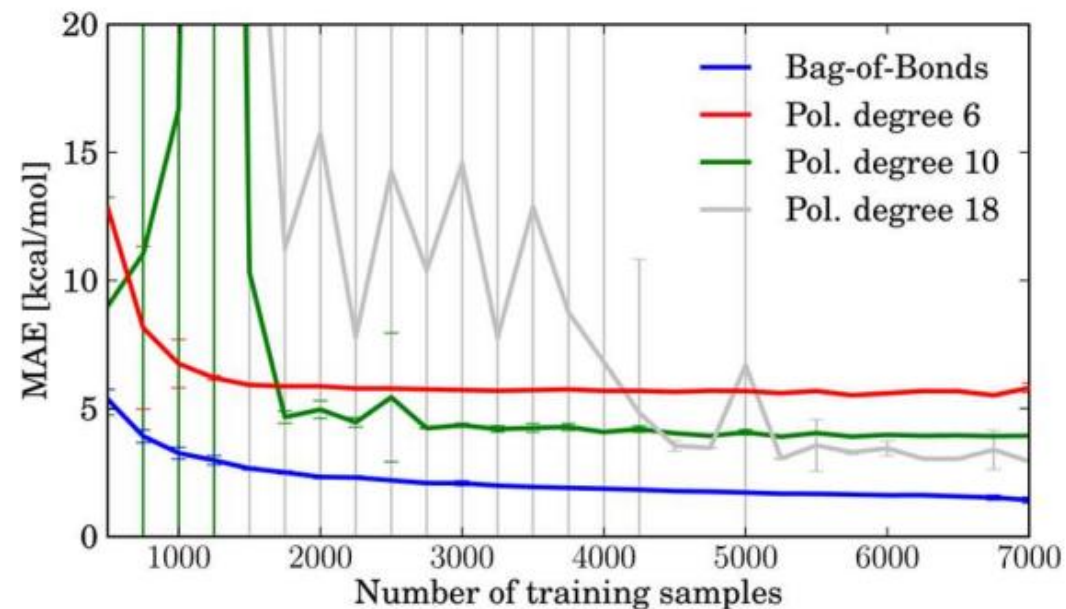
$$\hat{E}_{\text{BoB}}(\mathbf{M}) = \sum_{I=1}^N \alpha_I \exp(-d(\mathbf{M}, \mathbf{M}_I)/\sigma)$$

Так как размеры молекул различны, то все вектора были выравнены по максимальной длине, недостающие данные были заполнены нулями.

В итоге получили матрицу чисел и колонку с правильными ответами. Далее авторы обучили ridge regression для предсказания энергию всей молекулы.

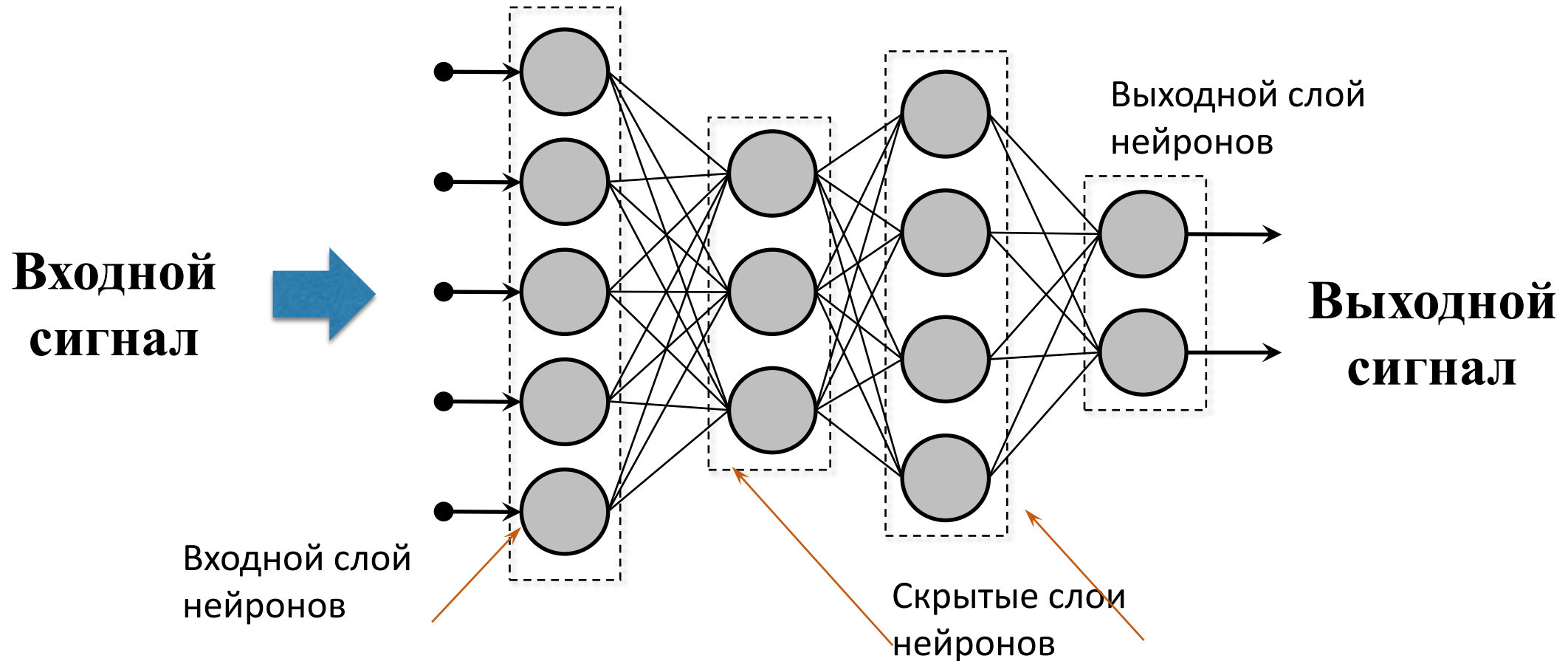
Кроме того, авторы сравнивали свою модель с другими подходами.

Mean absolute error (MAE in kcal/mol)





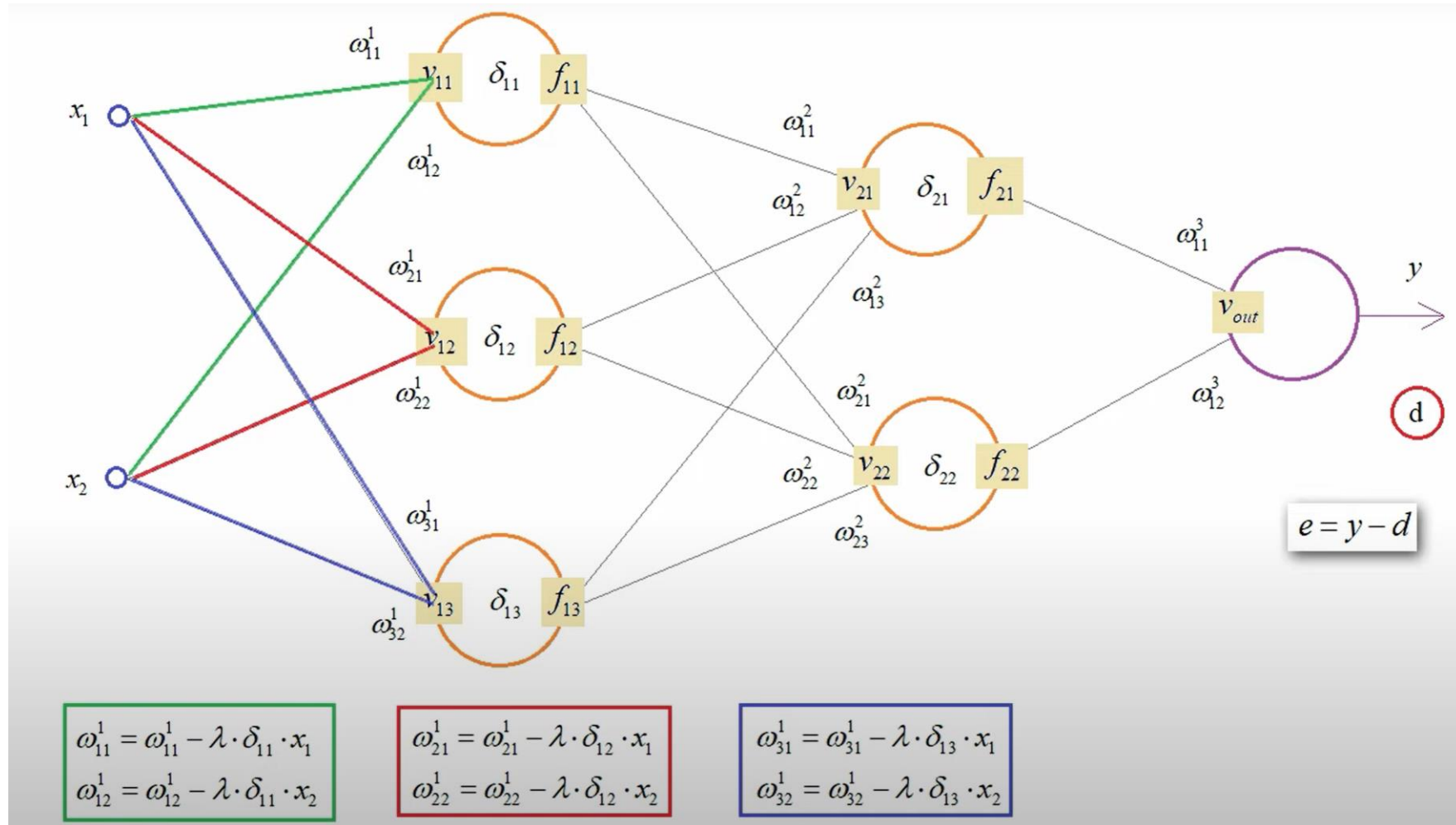
Как работает нейронная сеть (на примере FFNN)



Вычисление выходного вектора \mathbf{Y} , компонентами которого являются выходы **OUT** нейронов, сводится к матричному умножению $\mathbf{Y} = \mathbf{XW}$, где M – число входов, N – число слоев, и \mathbf{X} - векторы-строки.



ML Модели в материаловедении



Метод обратного распространения (Backpropagation)



ML Модели в материаловедении

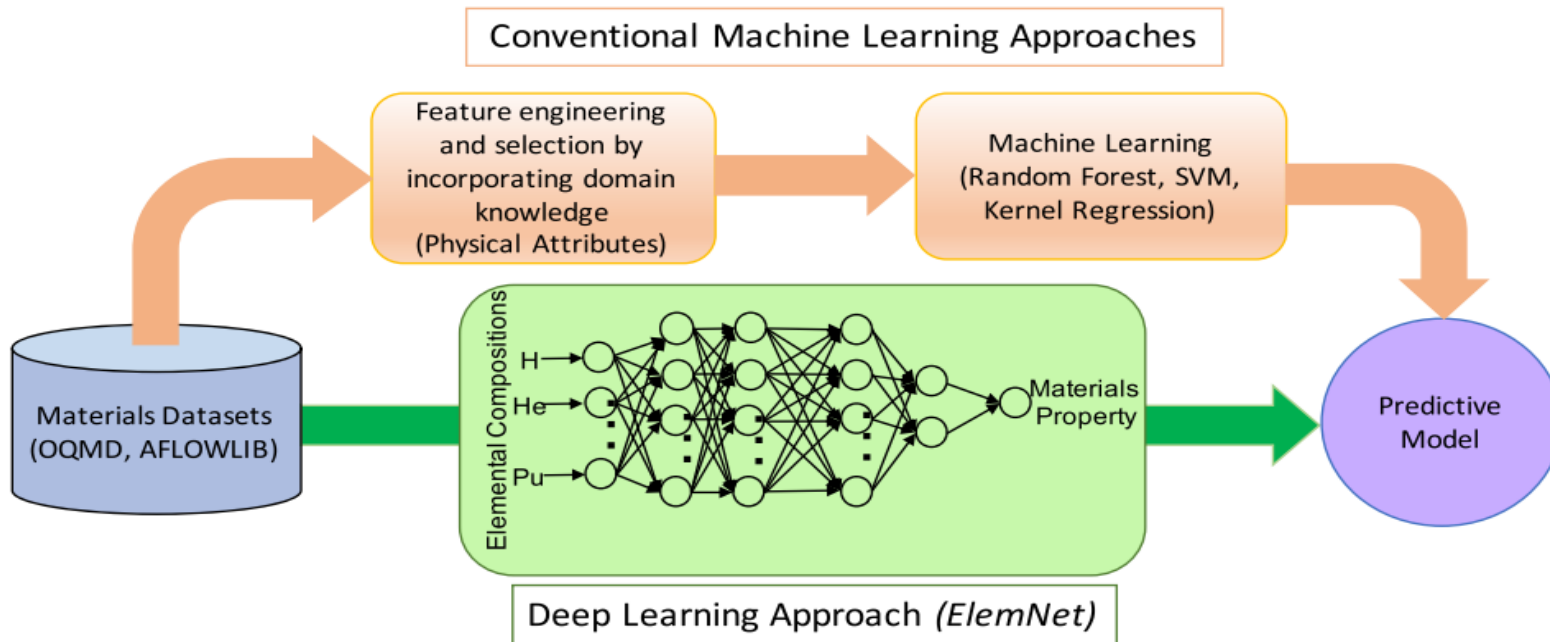
Для обучения и тестирования предложенной нами модели глубокого обучения был использован датасет OQMD10, в состав которой входят DFT-расчеты кристаллографических параметров и энтальпий образования экспериментально наблюдаемых соединений, взятых из базы данных неорганических кристаллических структур (ICSD), и гипотетических структур, созданных путем декорирования прототипных структур из ICSD различными способами. База данных OQMD постоянно расширяется и на момент написания статьи содержит 506 115 соединений с 275 778 уникальными составами. Авторы обучаем ML модели по наименьшим энтальпиям образования при каждом составе так как они представляют собой наиболее стабильные соединения, что позволяет модели предсказывать энергию основного состояния структуры в зависимости от состава.

Авторы построили сеть **ElemNet** и сравнивали нейронную сеть с другими алгоритмами машинного обучения. В состав признаков вошли 145 физических атрибутов, которые делятся на четыре различные категории - стехиометрические атрибуты, статистика свойств элементов, атрибуты электронной структуры и атрибуты ионных соединений.

Layer Types	No. of units	Activation	Layer Positions
Fully-connected Layer	1024	ReLU	First to 4th
Drop-out (0.8)	1024		After 4th
Fully-connected Layer	512	ReLU	5th to 7th
Drop-out (0.9)	512		After 7th
Fully-connected Layer	256	ReLU	8th to 10th
Drop-out (0.7)	256		After 10th
Fully-connected Layer	128	ReLU	11th to 13th
Drop-out (0.8)	128		After 13th
Fully-connected Layer	64	ReLU	14th to 15th
Fully-connected Layer	32	ReLU	16th
Fully-connected Layer	1	Linear	17th



Нейронные сети для решения дифференциальных уравнений



Model	Input Type	MAE (eV/atom)	Training time (hour)	Prediction time (sec)
RandomForest	Physical Attributes	0.071 ± 0.0006	1.5	14.80
RandomForest	Elemental Compositions	0.157 ± 0.0012	1.5	2.87
<i>ElemNet</i>	Elemental Compositions	0.050 ± 0.0007	7 (GPU)	9.28 (CPU) & 0.08 (GPU)



Комбинаторный скрининг кандидатов на получение новых материалов

Авторы **ElemNet** провели комбинаторный скрининг с использованием данной модели в огромном пространстве составов, включающем около полумиллиарда соединений, чтобы выяснить, может ли сеть идентифицировать стабильные соединения, которые отсутствуют в нашем обучающем наборе. Сначала авторы сформировали список из примерно 450 млн. гипотетических соединений вида **A_wB_xC_yD_z** где элементы (**A-D**) могут быть любыми из 86 элементов в OQMD, кроме He, Ne и Ar, а w-z - положительные целые числа, целые числа, где $w+x+y+z \leq 10$. Затем авторы оценивали ΔH_f этих композиций с помощью ElemNet. Поскольку ElemNet на два порядка быстрее чем другие модели, она позволяет чрезвычайно быстро сканировать для обнаружения новых материалов по сравнению с моделями, применяемыми на практике. Авторы выявляли составы, в которых возможно образование нового соединения, определяя составы, для которых ElemNet предсказывает энтальпию образования значительно ниже, определенного порога.

В частности, авторы вычисляли разницу между ΔH_f предсказанным ElemNet для каждого состава, и ΔH_f из базы данных OQMD. Всего было обнаружено 232 бинарных, 14 366 троичных и 353 352 четвертичных химических систем из 4,3 млн, в которых ΔH_f по данным ElemNet ниже текущего значения OQMD.

Category	Binary		Ternary	
	Count	Examples	Count	Examples
[Li,K,Na]-Containing	4	KF ₆ NaF ₈	707	NaY ₂ F ₇ KY ₂ F ₇
Chalco-/oxyhalides	5	OF ₉ SeF ₉	522	Y ₂ OF ₆ Sc ₂ OF ₇
Metal Oxides	1	Cu₂O	81	KTi ₄ O ₅ ReAu ₂ O ₅
3d Metal Oxides	1	Cu₂O	3	Zn ₂ (CuO) ₃ Ti ₅ CuO ₂
Intermetallics	11	Nb ₅ Sn ₃ Al ₅ Ir ₃	123	HfAl ₅ Ir ₃ YAl ₄ Ir ₃
Intermetallics $\Delta H_f < 2500$	0		0	

NN для решения дифференциальных уравнений - формулировка

Нейронные сети могут быть применены для решения дифференциальных уравнений благодаря их способности аппроксимировать любые функции.

1. Формулировка задачи: Предположим, что у нас есть дифференциальное уравнение, которое мы хотим решить. Например, дифференциальное уравнение вида $dy/dx = f(x, y)$, где f - известная функция. Мы хотим найти функцию $y(x)$, которая удовлетворяет этому уравнению и начальному условию $y(x_0) = y_0$.

2. Аппроксимация решения: Мы представляем искомую функцию $y(x)$ как выход нейронной сети с весами w и входом x : $y(x) = NN(x; w)$.

3. Формулировка функции потерь: функция потерь формулируется так, чтобы минимизировать разницу между левой и правой частями дифференциального уравнения, вычисленными в точках обучающей выборки, а также удовлетворить начальным условиям. Например, для уравнения $dy/dx = f(x, y)$ функция потерь может быть такой: $L(w) = \sum (dy/dx - f(x, NN(x; w)))^2 + (NN(x_0; w) - y_0)^2$.

3. Обучение нейронной сети: Нейронная сеть обучается минимизировать функцию потерь с помощью алгоритма оптимизации, градиентного спуска. В процессе обучения веса w нейронной сети подстраиваются так, чтобы функция $NN(x; w)$ все лучше и лучше удовлетворяла дифференциальному уравнению и начальным условиям.

4. Использование обученной сети: После обучения нейронная сеть может быть использована для приближенного решения дифференциального уравнения в любых точках. Выход нейронной сети для данного входа x дает приближенное значение искомой функции $y(x)$ в этой точке.

NN для решения дифференциальных уравнений: + и -

Достоинства применения нейронных сетей для решения дифференциальных уравнений:

- 1. Универсальность:** Нейронные сети могут приближать практически любые функции, что делает их подходящими для решения широкого спектра дифференциальных уравнений.
- 2. Параллелизм:** Обучение нейронных сетей может быть эффективно распараллелено, что позволяет использовать современные вычислительные ресурсы для ускорения решения.
- 3. Работа с высокоразмерными пространствами:** Нейронные сети могут быть эффективно использованы для решения дифференциальных уравнений в высокоразмерных пространствах, где традиционные методы могут столкнуться с "проклятием размерности".

Недостатки применения нейронных сетей для решения дифференциальных уравнений:

- 1. Сложность настройки:** Нейронные сети требуют тонкой настройки гиперпараметров, включая архитектуру сети, скорость обучения и другие параметры оптимизатора.
 - 2. Недостаток гарантий сходимости:** В отличие от некоторых традиционных численных методов, нейронные сети не всегда могут гарантировать сходимость к истинному решению.
 - 3. Требовательность к вычислительным ресурсам:** Обучение нейронной сети может потребовать значительных вычислительных ресурсов и времени, особенно для сложных и высокоразмерных задач.
- Сложность интерпретации:** Результаты, полученные с помощью нейронных сетей, могут быть сложны для интерпретации, поскольку нейронные сети являются "черными ящиками".



Open source projects in materials science

<https://www.opensourceagenda.com/tags/materials-science>

Deepchem

Democratizing Deep-Learning for Drug Discovery, Quantum Chemistry, Mater...

Deepmd Kit

A deep learning package for many-body potential energy representation an...

Pymatgen

Python Materials Genomics (pymatgen) is a robust materials analysis code...

Megnet

Graph Networks as a Universal Machine Learning Framework for Molecules a...

Fipy

FiPy is a Finite Volume PDE solver written in Python

Nequip

NequIP is a code for building E(3)-equivariant interatomic potentials

Matminer

Data mining for materials science

DFTK.jl

Density-functional toolkit

Dscribe

Dscribe is a python package for creating machine learning descriptors fo...

Tomviz

Cross platform, open source application for the processing, visualizatio...

Pycalphad

CALPHAD tools for designing thermodynamic models, calculating phase diag...

Pycroscopy

Scientific analysis of nanoscale materials imaging data

Open source projects in materials science

1. Проект **DeepChem** содержит обширную коллекцию учебных пособий. Все учебные пособия предназначены для работы в Google colab (или локально, если вы предпочитаете). Учебники расположены в предлагаемой последовательности обучения, которая позволит вам пройти путь от новичка до специалиста в области молекулярного машинного обучения и вычислительной биологии в целом. Links: <https://www.opensourceagenda.com/projects/deepchem>
Python, tensorflow

2. Целью проекта **Deep Potential** является использование методов глубокого обучения и реализация общей, точной, вычислительно эффективной и масштабируемой модели межатомной потенциальной энергии. Ключевым компонентом является соблюдение экстенсивных и симметрично-инвариантных свойств модели потенциальной энергии путем назначения локальной системы отсчета и локального окружения для каждого атома. Каждое окружение содержит конечное число атомов, локальные координаты которых расположены с сохранением симметрии. Затем эти локальные координаты преобразуются через подсеть в так называемую атомную энергию. Суммирование всех атомных энергий дает потенциальную энергию системы.

Links: <https://www.opensourceagenda.com/projects/deepmd-kit>

C++, Python



Open source projects in materials science

3. Pymatgen (Python Materials Genomics) - это библиотека Python с открытым исходным кодом для анализа материалов. Вот некоторые из ее основных особенностей: Высокогибкие классы для представления объектов Element, Site, Molecule и Structure. Широкая поддержка ввода/вывода данных, включая поддержку форматов файлов VASP, ABINIT, CIF, Gaussian, XYZ и многих других. Мощные средства анализа, включая построение фазовых диаграмм, диаграмм Пурбе, диффузионных анализов, реакций и т.д. Анализ электронной структуры, такой как плотность состояний и полосовая структура. Интеграция с REST API проекта Materials Project.

Links: <https://www.opensourceagenda.com/projects/pymatgen>

python

4. Data Resources For Materials Science

Данная коллекция включает список онлайн-овых и оффлайн-овых ресурсов по физическим, химическим, механическим и всем другим свойствам материалов.

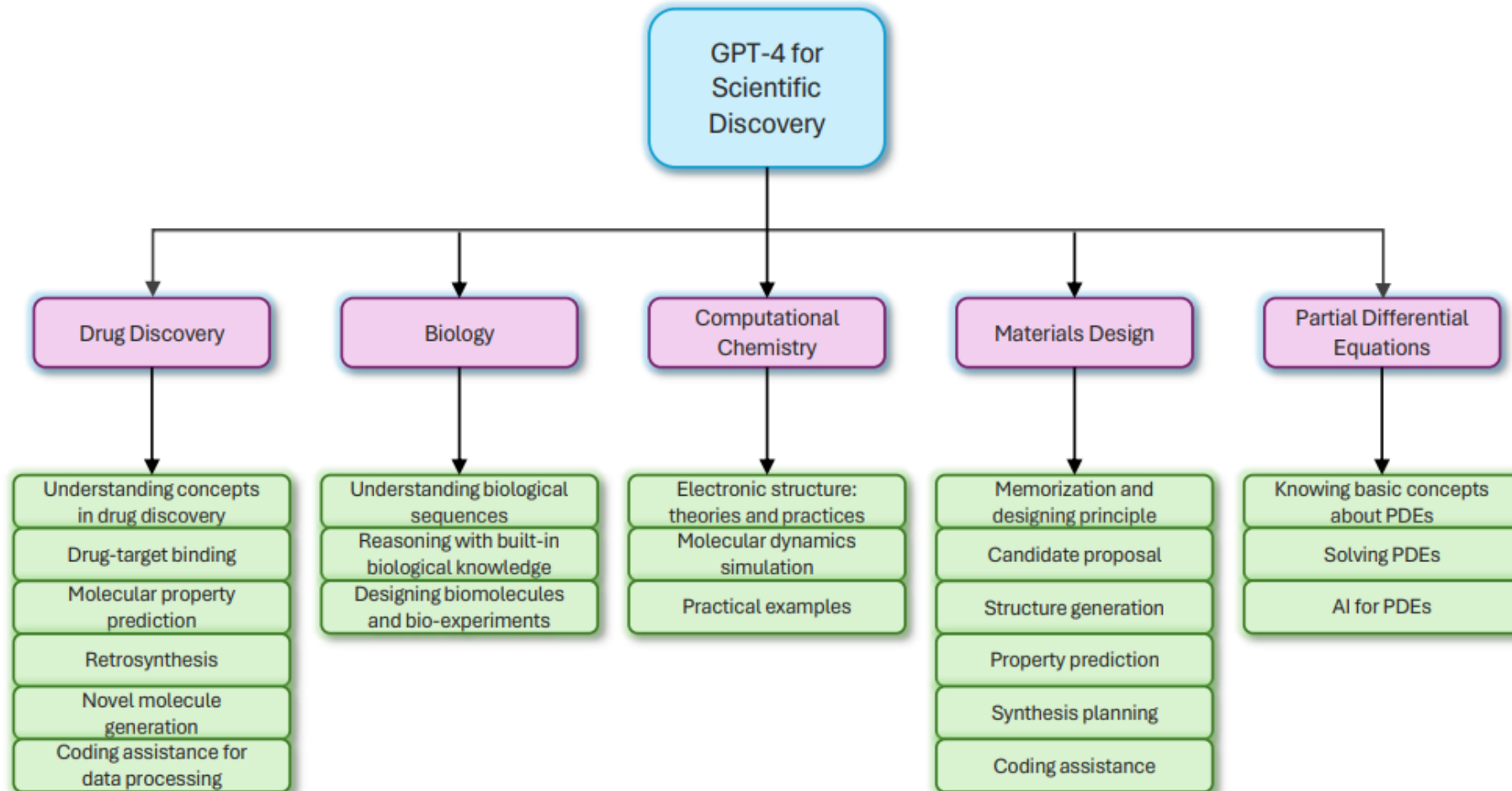
<https://www.opensourceagenda.com/tags/materials-science>

Databases in Materials Science

Database Name	Description
NIST Materials Genome Initiative (MGI)	Several databases for different material classes
NIST Materials Data Curation System (Part of MGI)	Phase transformation temperatures (e.g. melting, solidus, solvus), lattice parameters, thermal expansion, elastic constants, and diffusion coefficients and compositions profiles
NIST Materials Data	



Применение LLM в области Materials Science: 13.November.2023





Применение LLM в области Materials Science: 13.November.2023

Промпт инжиниринг

GPT-4

Prompt:

For inorganic solid electrolytes, can you categorize them into families based on the anion chemistry?

GPT-4:

Certainly! Inorganic solid electrolytes can also be categorized into families based on the anion chemistry. Here are some of the main categories:

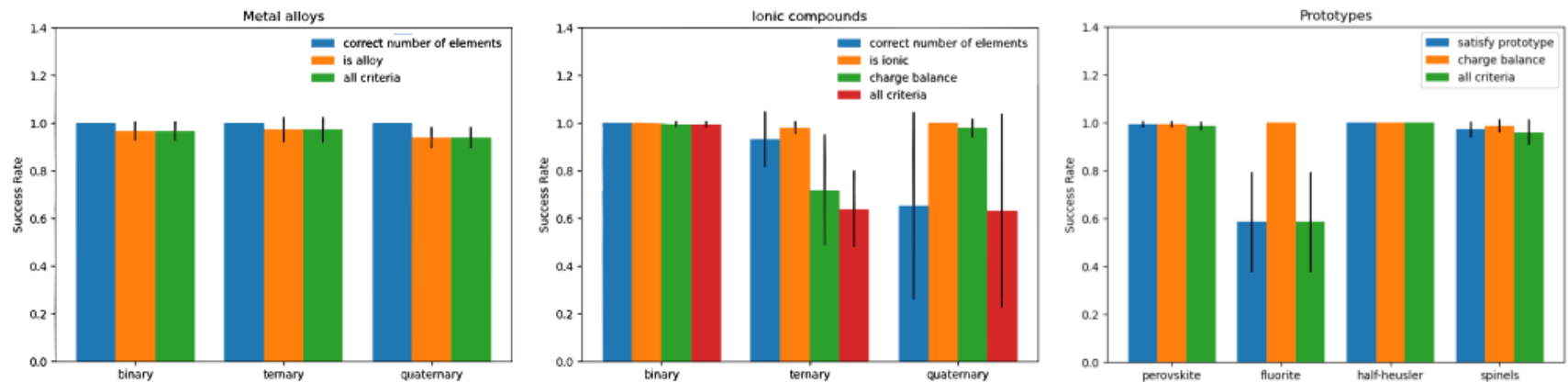
1. Oxide-based electrolytes: These electrolytes have oxide anions (O^{2-}) in their structure. Examples include:
 - a. Perovskite-type: e.g., $La_{0.67}Sr_{0.33}Al_{0.67}Ta_{0.33}O_3$ (LSAT) for sodium-ion conductors.
 - b. NASICON (Na Super Ionic CONductor)-type: e.g., $Li_{1.5}Al_{0.5}Ge_{1.5}(PO_4)_3$ (LAGP) for lithium-ion conductors.
 - c. Garnet-type: e.g., $Li_7La_3Zr_2O_{12}$ (LLZO) for lithium-ion conductors.
2. Sulfide-based electrolytes: These electrolytes have sulfide anions (S^{2-}) in their structure. Examples include:
 - a. Thio-LISICON (Li Super Ionic CONductor)-type: e.g., Li_4GeS_4 and Li_4SiS_4 .
 - b. Argyroditite-type: e.g., Li_6PS_5Cl and Li_6PS_5Br .
 - c. Thiophosphate-based: e.g., $Li_{10}GeP_2S_{12}$ (LGPS).



Применение LLM в области Materials Science: 13.November.2023

Prompt

- You are a materials scientist assistant and should be able to help with proposing new chemical composition of materials.
- You are asked to propose a list of chemical compositions given the requirements.
- The format of the chemical composition is $A_xB_yC_z$, where A, B, and C are elements in the periodic table, and x, y, and z are the number of atoms of each element.
- The answer should be only a list of chemical compositions separated by a comma.
- The answer should not contain any other information.
- Propose 30 requirements.



Left: the success rate of generating chemical composition of metal alloys. **Middle:** the success rate of generating the chemical position of ionic compounds. **Right:** the success rate of generating the chemical composition of given prototypes.

Применение LLM в области Materials Science: 13.November.2023

Table 10: Prediction of atomic coordinates with GPT-4.

Formula	Element	Correct CN	GPT-4 CN
BaAl ₂ O ₄	Ba	9	(provided as example)
BaAl ₂ O ₄	Al	4	(provided as example)
BaAl ₂ O ₄	O	2	(provided as example)
Be ₂ SiO ₄	Be	4	4
Be ₂ SiO ₄	Si	4	4
Be ₂ SiO ₄	O	3	2
Ca(BO ₂) ₂	Ca	8	7
Ca(BO ₂) ₂	B	3	3
Ca(FeO ₂) ₂	Ca	8	6
Ca(FeO ₂) ₂	Fe	6	4
Ca(FeO ₂) ₂	O	5	2
Fe ₂ SiO ₄	Fe	6	6
...

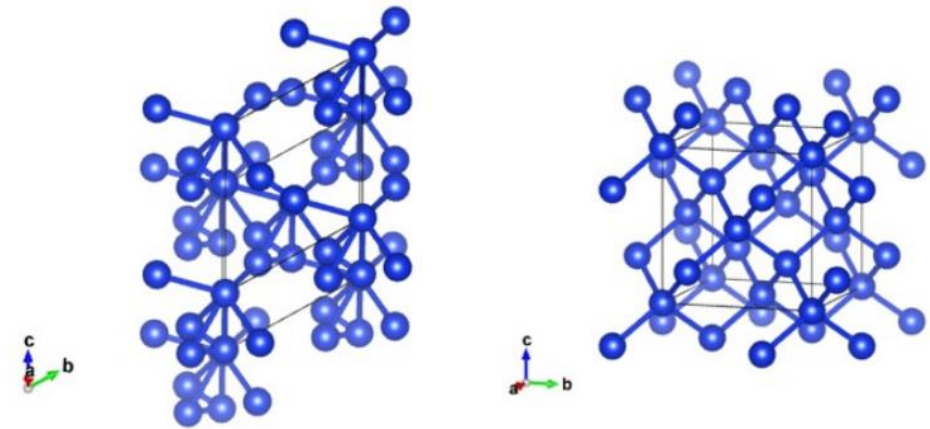


Figure 5.6: crystal structure of silicon predicted by GPT-4 and from materials project.



Применение LLM в области дифф. уравнений: 13.November.2023

GPT-4

Prompt:
please provide MATLAB code to solve the following equation

$$u_t - \Delta(u^\gamma) = 0 \quad (x, t) \in [0, 1] \times [0, 1],$$

$$u(x, 0) = \lambda^{\frac{1}{1-\gamma}} x^\alpha,$$

$$u(0, t) = 0,$$

$$u(1, t) = ((1 - \gamma)\mu t + \lambda)^{\frac{1}{1-\gamma}}$$

where $u \geq 0$ and $\gamma = 2$, $\lambda = 100$, $\alpha = \frac{2}{\gamma-1}$ and $\mu = \alpha\gamma(\alpha\gamma - 1) > 0$. Further, the analytical solution is $u(x, t) = ((1 - \gamma)\mu t + \lambda)^{\frac{1}{1-\gamma}} x^\alpha$, please plot the analytical solution, numerical result and their comparison in different figures

GPT-4:
Here's a MATLAB code that solves the given equation numerically, plots the analytical solution, numerical result, and their comparison in different figures:

```
% Parameters
gamma = 2;
lambda = 100;
alpha = 2 / (gamma - 1);
mu = alpha * gamma * (alpha * gamma - 1);
% Discretization
N = 100; M = 100;
dx = 1 / (N - 1); dt = 1 / (M - 1); x = linspace(0, 1, N); t = linspace(0, 1, M);
% Initialize solution matrix
U = zeros(N, M);
% Boundary and initial conditions
U(:, 1) = lambda^(1 / (1 - gamma)) * x.^alpha;
U(1, :) = 0;
U(end, :) = ((1 - gamma) * mu * t + lambda).^(1 / (1 - gamma));
% Numerical solution using finite difference method
for j = 1:M-1
```



Применение LLM в области дифф. уравнений: 13.November.2023

GPT-4

Prompt:

Some works leverage deep learning to approximate the solution of the PDE. Give me some representative works and introduce them with one sentence.

GPT-4:

1. Raissi, Maziar, Paris Perdikaris, and George Em Karniadakis. "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations." *Journal of Computational Physics* 378 (2019): 686-707. This work presents physics-informed neural networks (PINNs) as a deep learning framework for solving forward and inverse problems involving nonlinear PDEs by encoding the governing equations into the loss function.
2. Sirignano, Justin, and Konstantinos Spiliopoulos. "DGM: A deep learning algorithm for solving partial differential equations." *Journal of Computational Physics* 375 (2018): 1339-1364. This paper introduces the Deep Galerkin Method (DGM), a deep learning-based algorithm for solving high-dimensional PDEs, with applications in finance and physics.
3. Han, Jingtian (Correction: Jiequn, Han), et al. "Solving high-dimensional partial differential equations using deep learning." *Proceedings of the National Academy of Sciences* 115.34 (2018): 8505-8510. The authors propose a deep learning-based method to solve high-dimensional PDEs using feedforward neural networks and demonstrate its efficiency in solving a variety of PDEs, including the Black-Scholes and Hamilton-Jacobi-Bellman equations.
4. Long, Zichao, Yiping Lu, and Bin Dong. "PDE-Net: Learning PDEs from data." *International Conference on Machine Learning*. PMLR, 2018. This work introduces PDE-Net, a deep learning framework that learns the dynamics of PDEs from data by using a trainable numerical integration scheme and convolutional neural networks to approximate the spatial derivatives.