



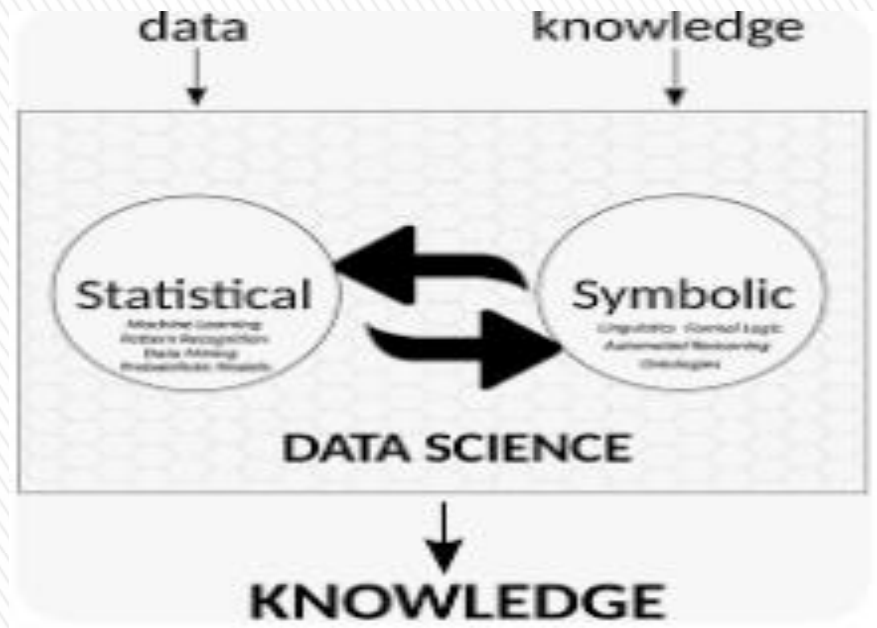
# » Лекционен курс

## » Въведение в генеративния ИИ



**Увод** >

# Изкуствен интелект (ИИ)



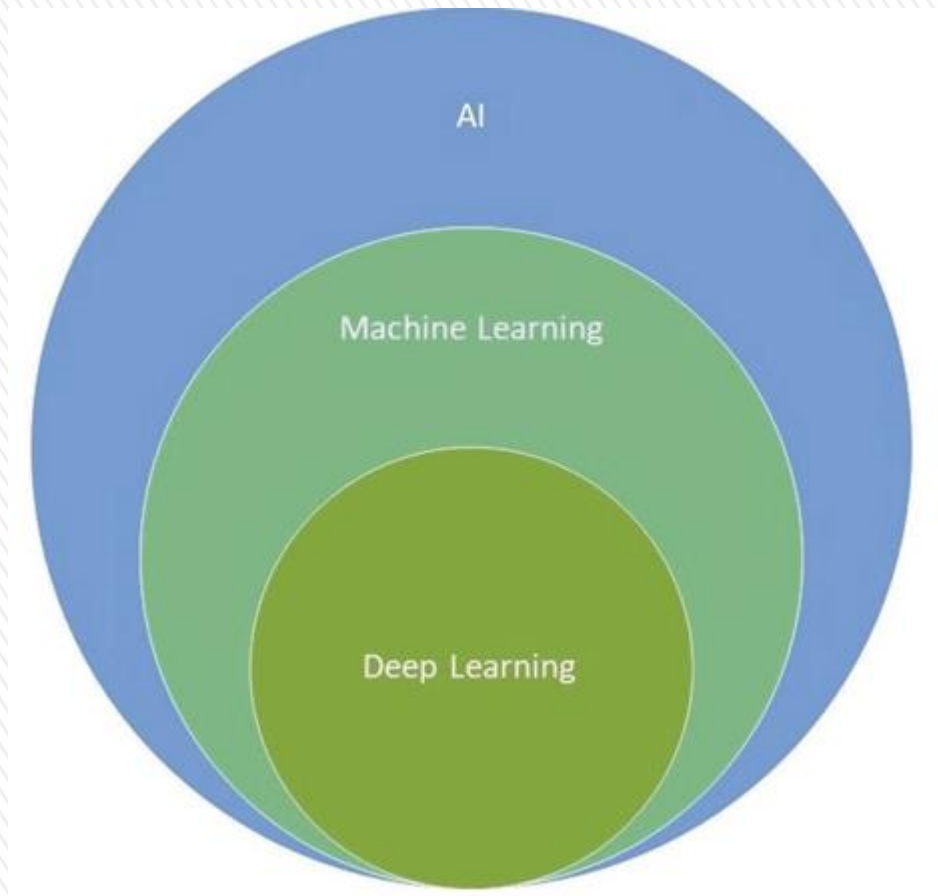
Два основни насоки:

- ✓ Символен ИИ
- ✓ Статистически ИИ

Взаимно се допълват



# Машинно учене (МУ)



Изкуствен интелект (ИИ)  
Машинно учене (МУ)  
Дълбоко учене (ДУ)



# Какво е невронна мрежа?

- » Най-общо, една **невронна мрежа** е съвкупност от взаимосвързани възли (наречени **неврони**), които **работят заедно** за да **обработват и анализират данни**.
- » Всеки неврон в невронна мрежа получава **входове** от други неврони и произвежда **изход**, който се предава на следващи неврони в мрежата.
- » Входовете към всеки неврон са **претеглени**, което означава, че някои входове са **по-важни** от други при определяне на изхода на неврона.



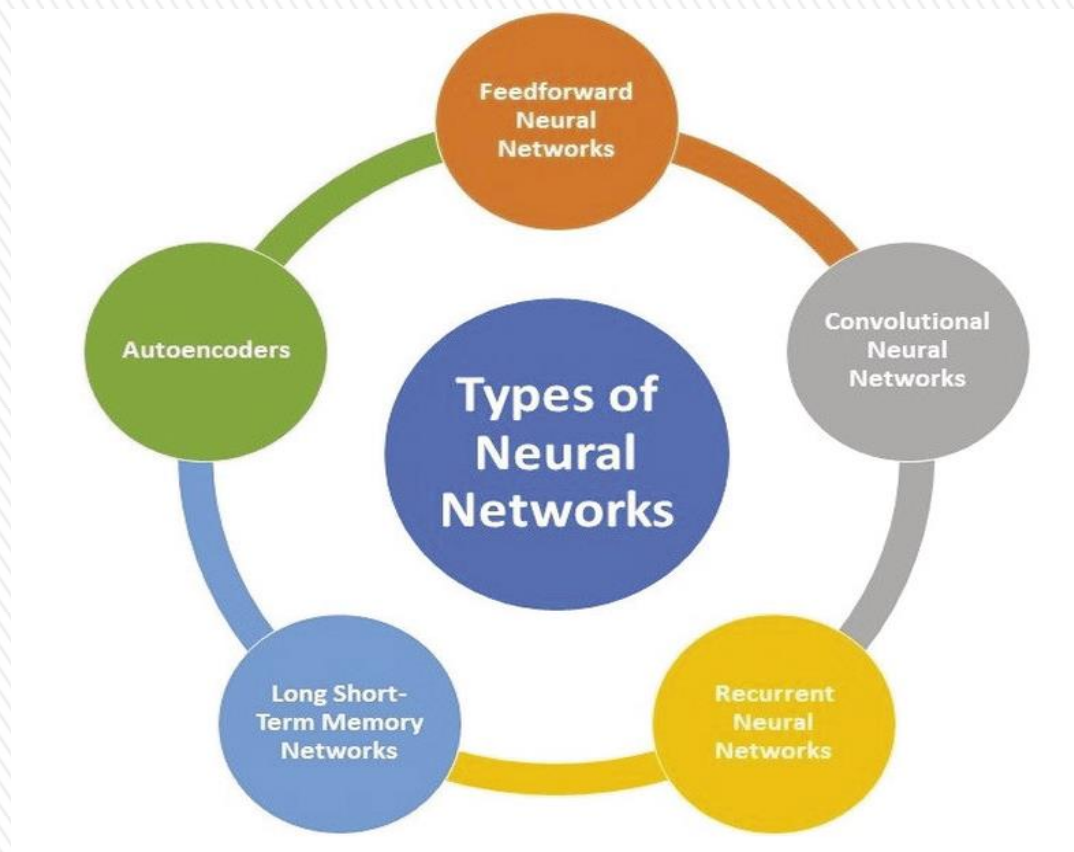
# Какво е невронна мрежа?

- » Невроните в невронната мрежа са организирани в **слоеве**, като всеки слой има **специфична функция** при обработката на данните.
- » **Входният слой** е мястото, където данните първоначално се въвеждат в мрежата.
- » **Изходният слой** произвежда крайния резултат от обработката на мрежата.
- » Между входния и изходния слой има един или повече **скрити слоеве**, които извършват по-голямата част от обработката в мрежата.
- » Съществуват **различни типове** невронни мрежи.





# Невронните мрежи с връзка напред



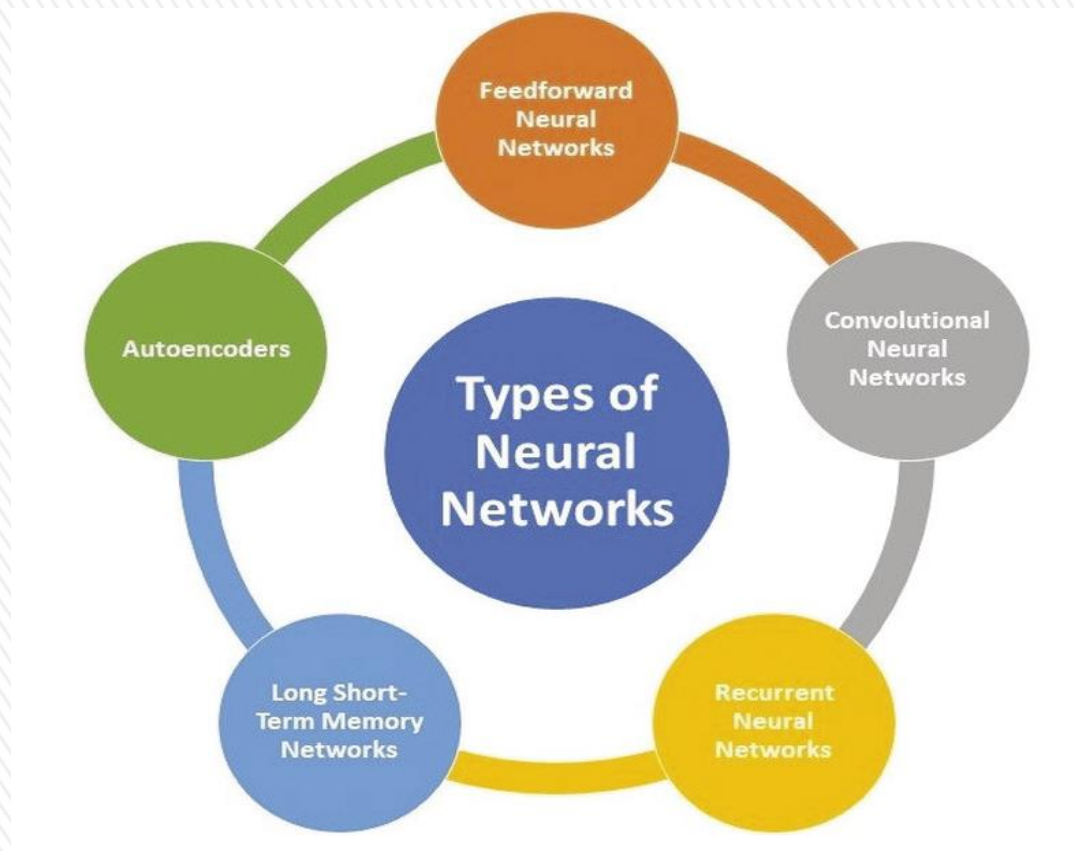
Най-основният тип невронни мрежи.

Състоят се от един или повече слоеве от неврони, които са свързани по начин на предаване напред, което означава, че **изходът на всеки неврон се предава като вход към следващия слой** от неврони.

Тези невронни мрежи обикновено се използват за задачи за **класификация** или **регресия**, където изходът е **единична стойност** или **етикет на клас**.



# Конволюционни невронни мрежи (CNN)



Обикновено се използват в задачи за **разпознаване** на изображения и видео.

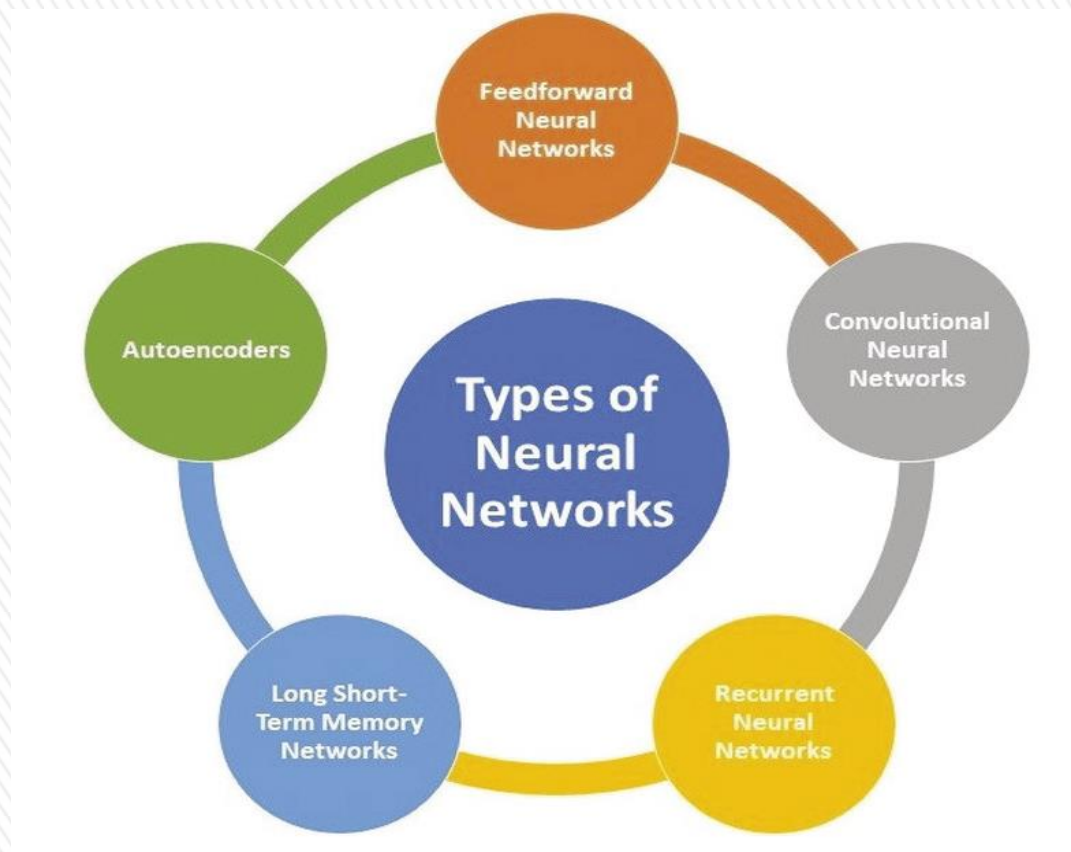
Състоят се от няколко слоя неврони, които са проектирани да **обработват изображения или видеоклипове по йерархичен начин**.

Първият слой на CNN обикновено е **конволюционен слой**, който прилага серия от **филтри** към входното изображение, за да **извлече характеристики** като ръбове или текстури.

След това изходът от конволюционния слой се предава на един или повече напълно свързани слоеве за по-нататъшна обработка и класификация.



# Повтарящите се невронни мрежи (RNN)



Проектирани да обработват **последователни данни**, като **реч** или **текст**.

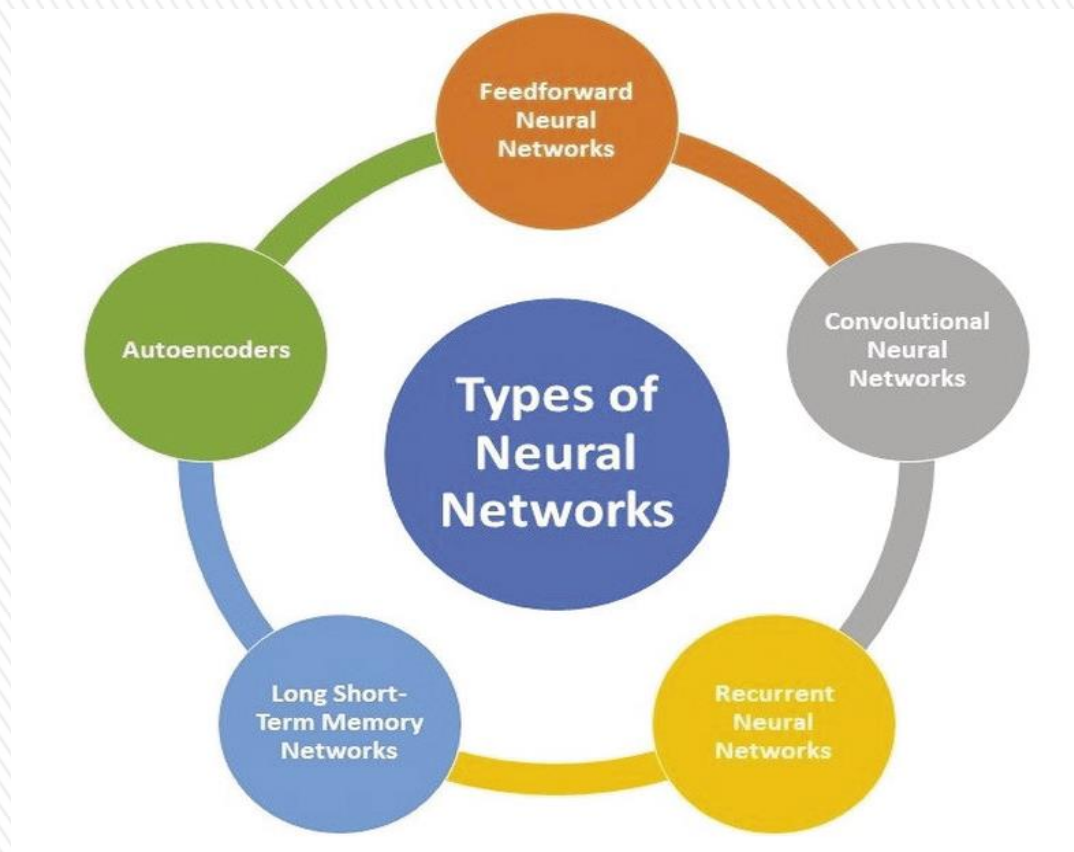
Те се състоят от **поредица от взаимосвързани неврони**, като всеки неврон в мрежата получава входни данни както от **предишния неврон** в последователността, така и от **текущия вход**.

RNN са способни да научават **дългосрочни зависимости** във входните данни, което ги прави подходящи за задачи като **езиков превод** или **разпознаване на реч**.





# Мрежите с дълга краткосрочна памет (LSTM)



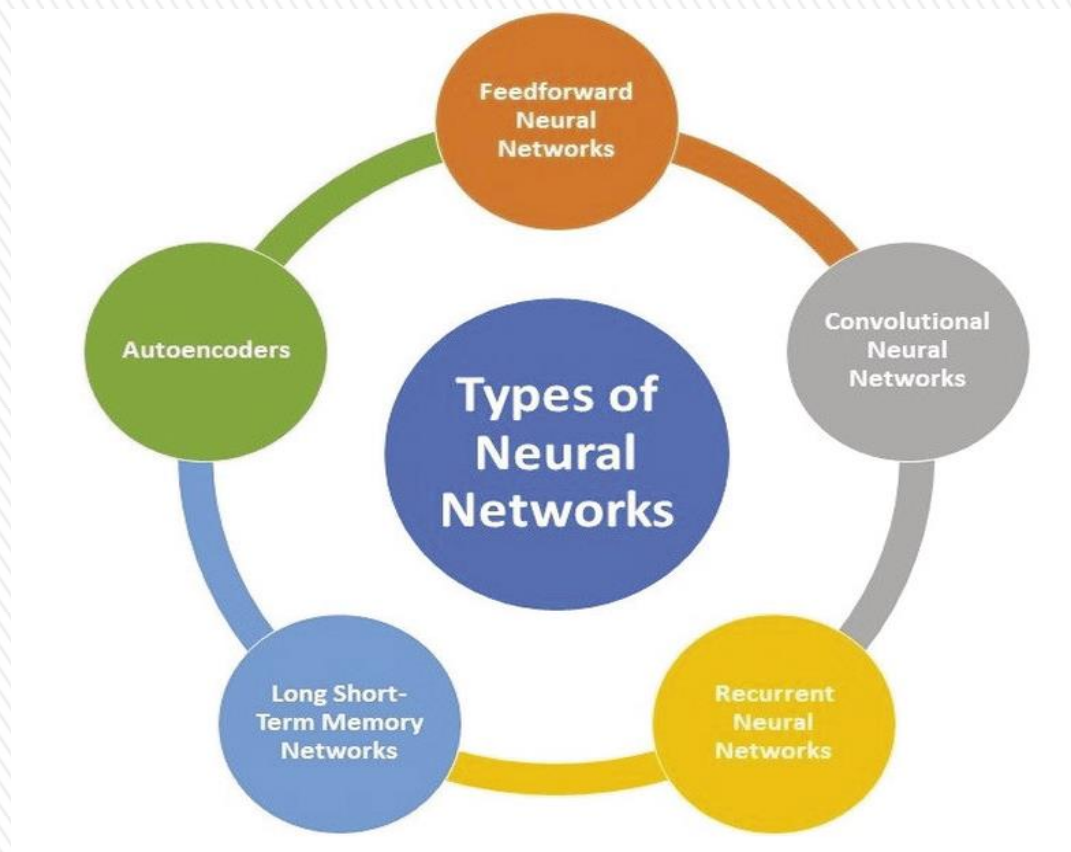
Тип RNN, който е специално проектиран да обработва **дългосрочни зависимости** в последователни данни.

Те се състоят от поредица от **клетки с памет**, които могат да бъдат **избирателно** четени, записвани или изтривани въз основа на входните данни.

LSTM са много подходящи за задачи като **разпознаване на реч** или **разпознаване на ръкописен текст**, където входните данни може да имат **дългосрочни зависимости**, които трябва да бъдат уловени.



# Автоенкодери



Използват се за задачи за **неконтролирано учене**, като **извличане на характеристики** или **компресиране на данни**.

Те се състоят от:

- ✓ Енкодерна мрежа, която взема входните данни и ги преобразува в **по-нискоизмерно латентно пространство**
- ✓ Декодерна мрежа, която **реконструира оригиналните входни данни от латентното пространство**.

Автоматичните енкодери често се използват за задачи като **генериране на изображение или текст**, където изходът се генерира въз основа на научените характеристики на входните данни.

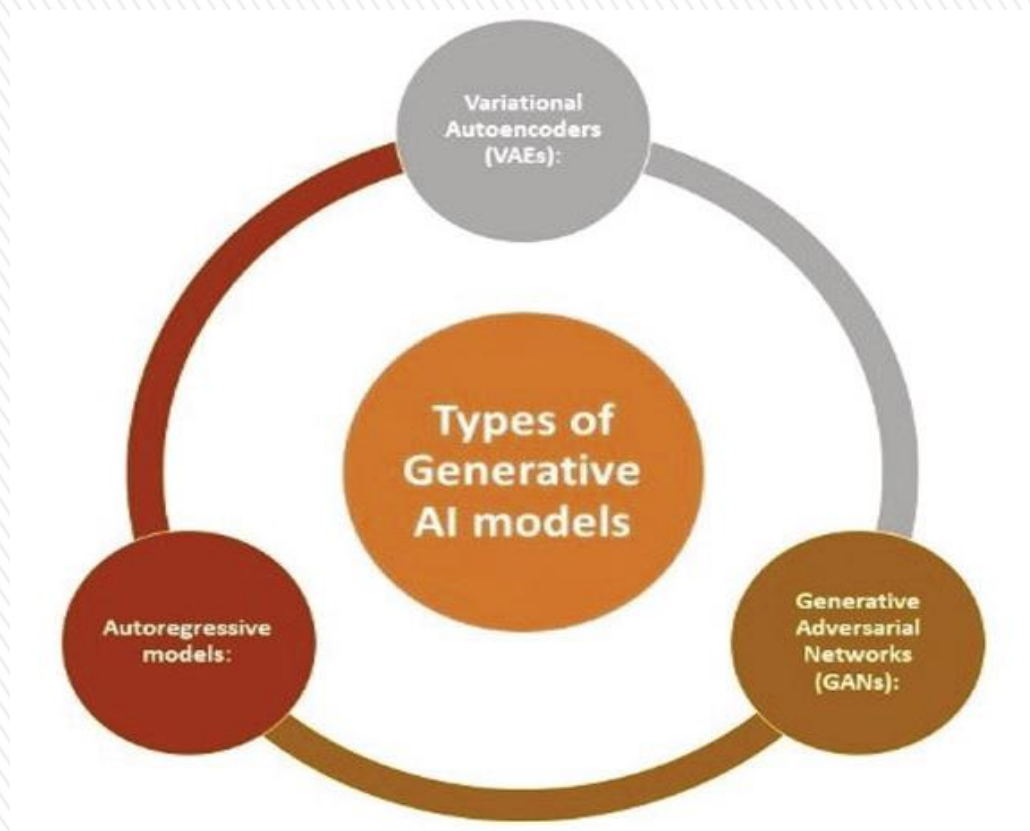


# Какво е генеративен ИИ (ГИИ)

- » ГИИ е клас техники за МУ, които имат за цел да генерират нови данни, които са подобни (но не идентични) на данните, с които са обучени.
- » С други думи, ГИИ моделите се научават да създават нови проби от данни, които имат сходни статистически свойства с данните за обучение, което им позволява да създават ново съдържание като изображения, видеоклипове, аудио или текст, които никога не са виждани преди.
- » Съществуват различни ГИИ модели.



# Вариационни автоенкодери (VAE)



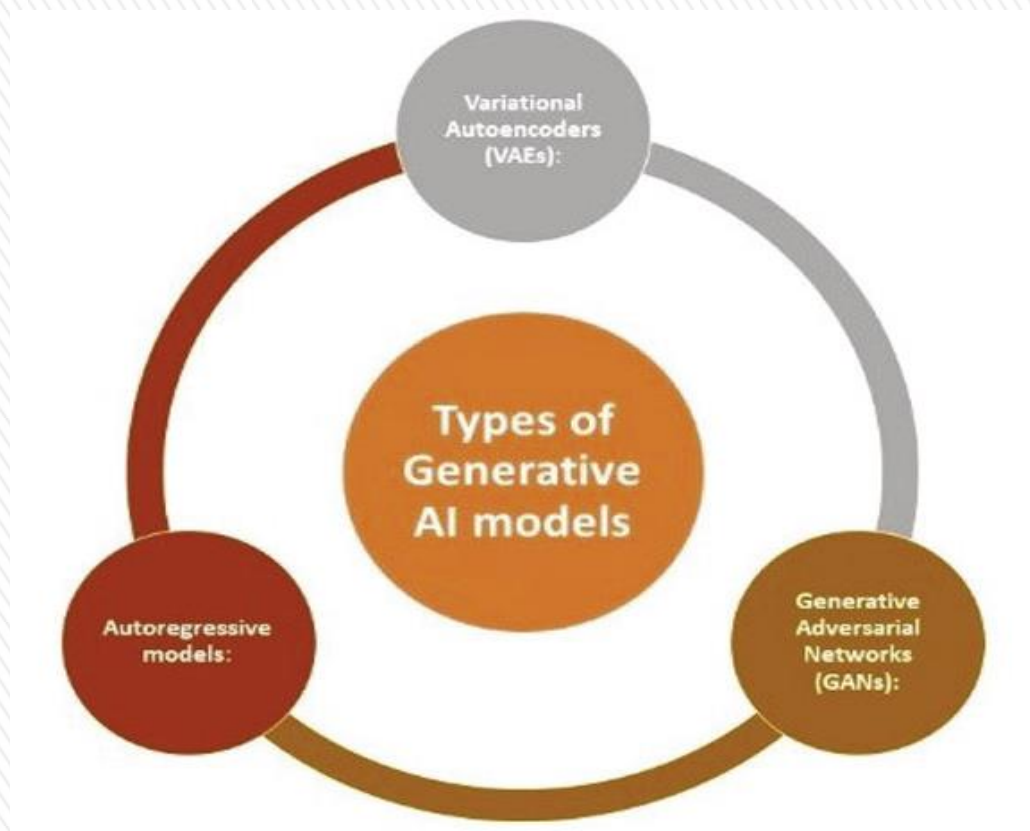
Тип генеративен модел, който се научава да **кодира** входни данни в **по-нискоизмерно латентно пространство**, след което **декодира** латентното пространство обратно в **изходно пространство** за да генерира нови данни, които са подобни на оригиналните входни данни.

Обикновено се използват за генериране на изображения и видео.





# Генеративни противопоставящи се мрежи (GANs)



Вид генеративен модел, който се научава да генерира нови данни чрез **противопоставяне на две невронни мрежи** – генератор и дискриминатор.

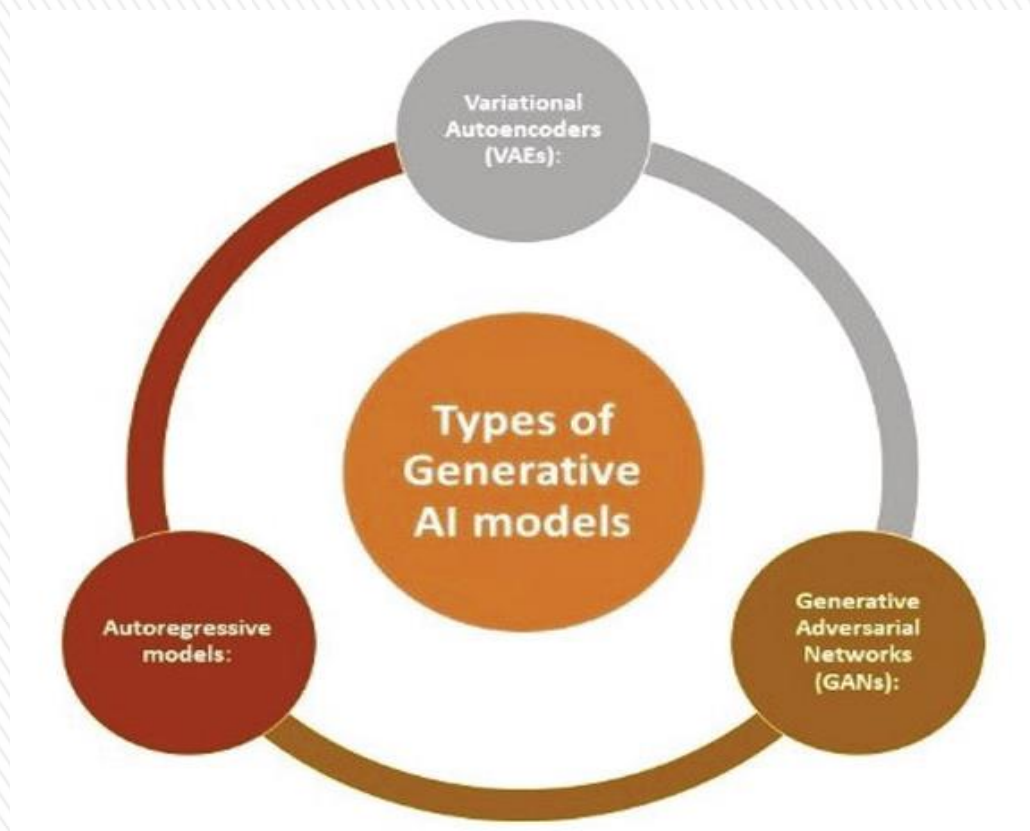
Генераторът се научава да **създава нови проби** от данни, които **могат да заблудят дискриминатора**.

Докато дискриминаторът се научава да **прави разлика между истински и фалшиви проби** от данни.

GAN обикновено се използват за **генериране на изображения, видео и аудио**.



# Авторегресивни модели



Вид генеративен модел, който се научава да генерира нови данни чрез **прогнозиране на вероятностното разпределение на следващата точка** от данни, дадена на предишните точки от данни.

Тези модели обикновено се използват за **генериране на текст**.



# Възможности на ГИИ



**Генериране на креативно съдържание:** Една от най-вълнуващите възможности в ГИИ е способността да се **създава ново и уникално съдържание в различни области** като изкуство, музика, литература и дизайн.

ГИИ може да помогне на артисти и дизайнери да създават нови и уникални произведения.



# Възможности на ГИИ



**Подобрена персонализация:** ГИИ може да помогне на бизнеса да предостави **по-персонализирано** изживяване на своите клиенти.

Например, може да се използва за генериране на **персонализирани препоръки**, **продуктов дизайн** или съдържание за потребителите въз основа на **техните предпочитания**.





# Възможности на ГИИ



**Подобрена поверителност на данните:** ГИИ може да се използва за **генериране на синтетични данни**, които **имитират** статистическите свойства на реалните данни, които могат да се използват за **защита на поверителността на потребителите**.

Това може да бъде особено полезно в **здравеопазването**, където чувствителните медицински данни трябва да бъдат защитени.



# Възможности на ГИИ

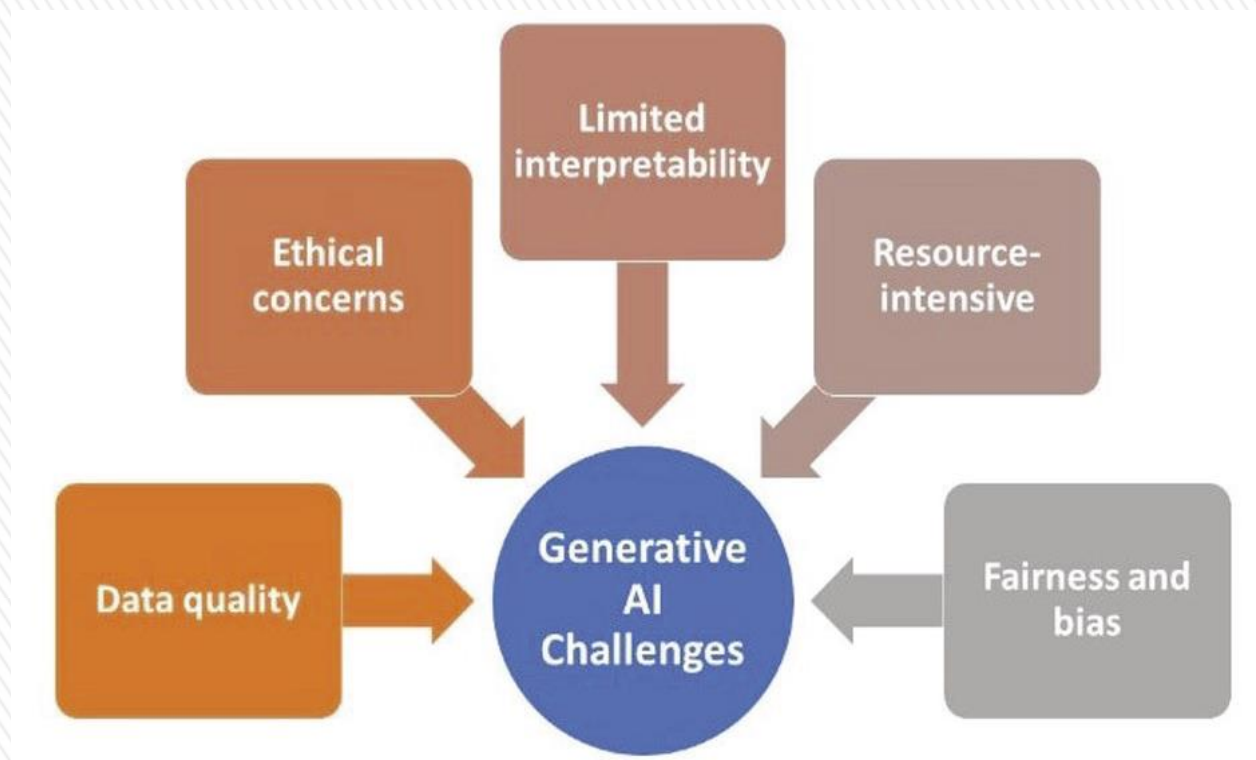


**По-добро вземане на решения:** ГИИ може да се използва за **генериране на алтернативни сценарии** за да помогне на вземащите решения да вземат по-добре информирани решения.

Например, може да се използва за симулация различни **сценарии във финанси, прогнозиране на времето** или **управление на трафика**.



# Предизвикателства на ГИИ

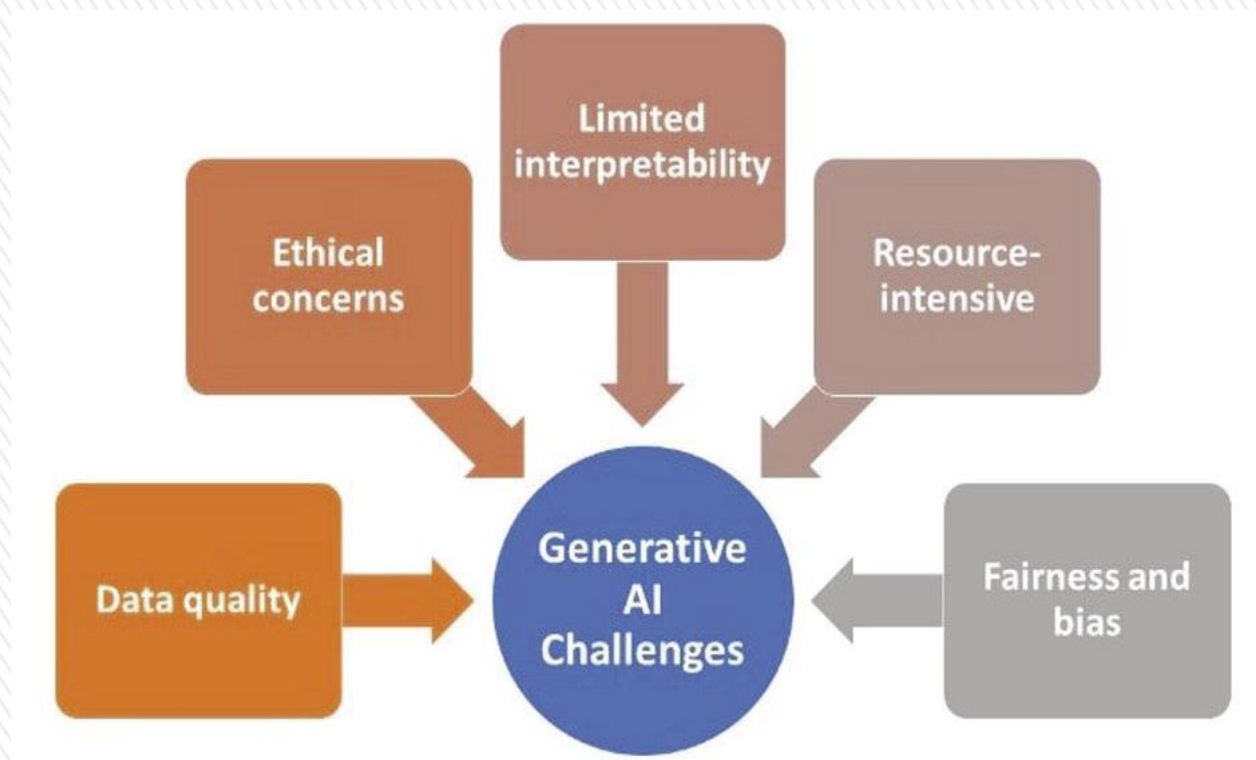


**Качество на данните:** ГИИ моделите силно разчитат на **качеството и количеството на данните**, използвани за тяхното обучение.

Данните с лошо качество могат да доведат до модели, които генерират **резултати с ниско качество**, което може да повлияе на тяхната използваемост и ефективност.



# Предизвикателства на ГИИ



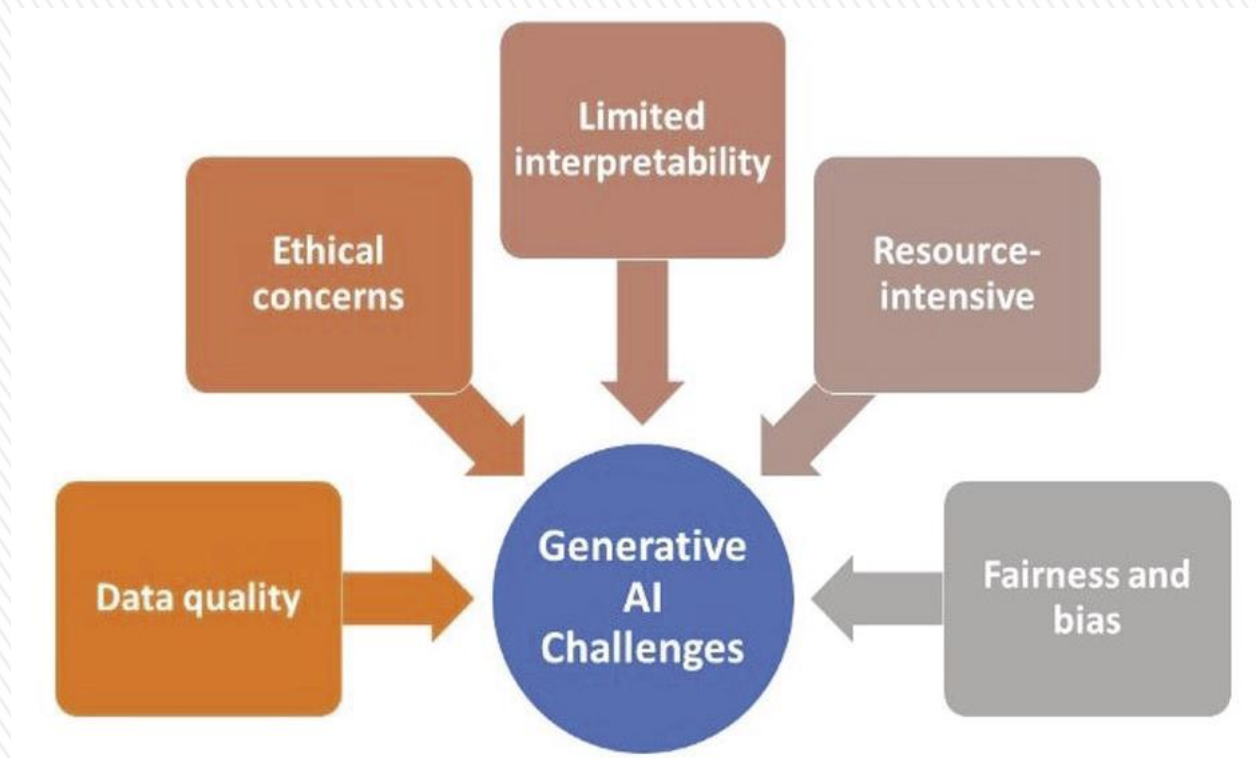
**Етични съображения:** ГИИ може да предизвика **етични опасения** относно използването на синтезирани данни, особено в области като **здравеопазването**, където синтетичните данни може да не отразяват точно данните от реалния свят.

Освен това ГИИ може да се използва за създаване на **фалшиви медии**, което може да има отрицателни последици, ако се използва неправилно.





# Предизвикателства на ГИИ

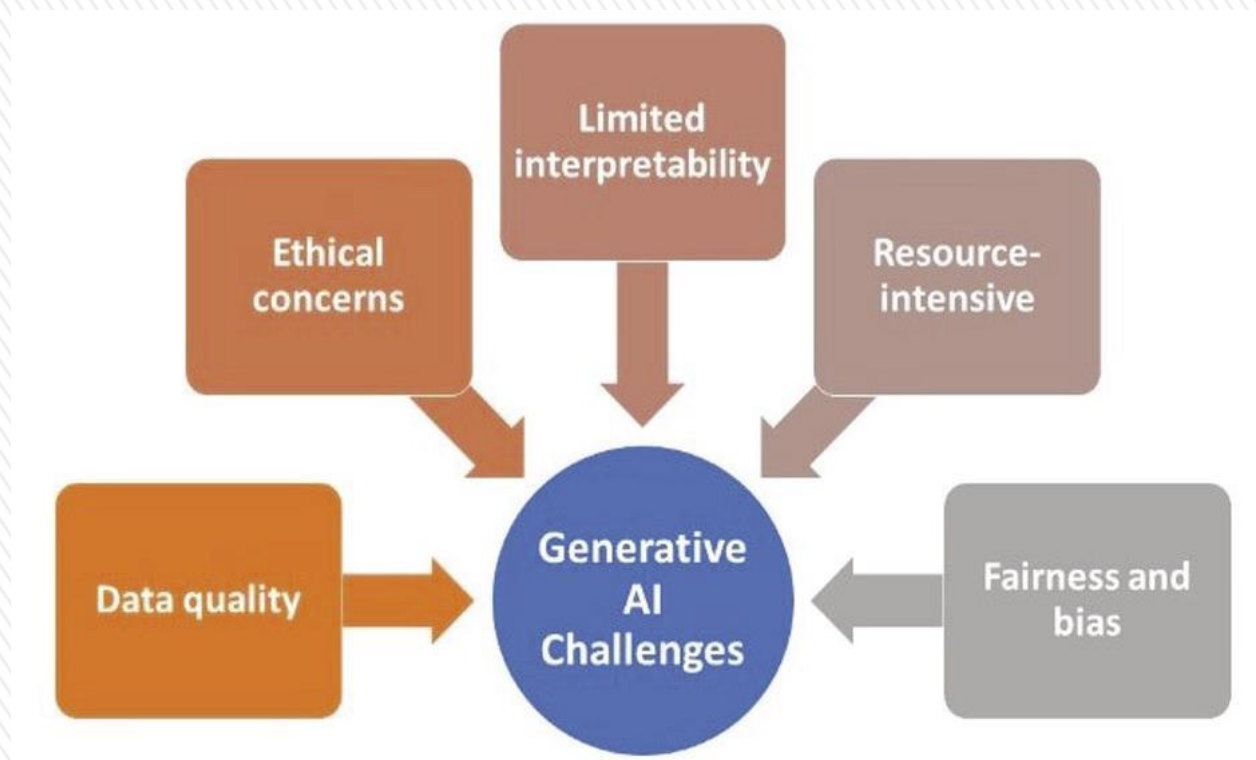


**Ограничена интерпретируемост:** ГИИ моделите могат да бъдат **сложни и трудни за интерпретиране**, което затруднява разбирането как те генерират своите резултати.

Това може да затрудни диагностицирането и коригирането на грешки или отклонения в моделите.



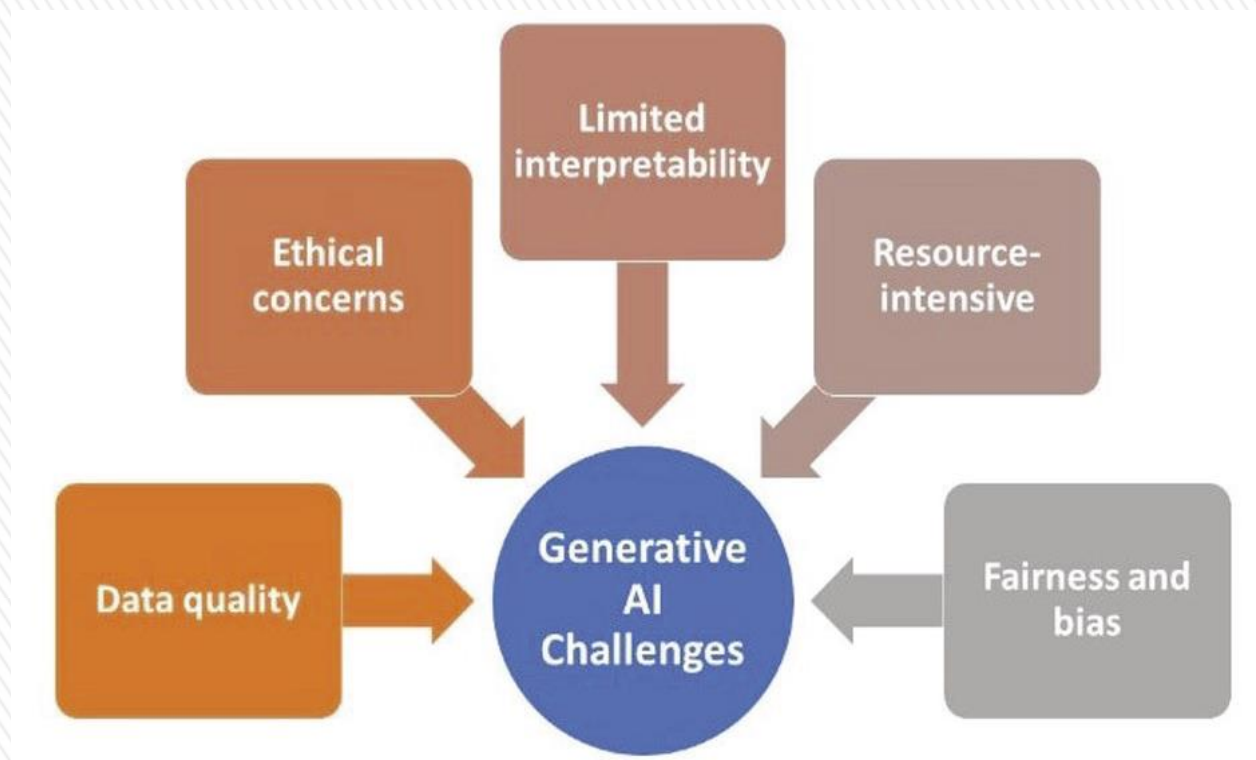
# Предизвикателства на ГИИ



**Интензивни ресурси:** ГИИ модели изискват **значителна изчислителна мощност и време за обучение**, което прави предизвикателство мащабирането им за големи набори от данни или приложения в реално време.



# Предизвикателства на ГИИ



**Справедливост и пристрастия:** ГИИ моделите могат да поддържат пристрастия, присъстващи в данните за обучение, което води до резултати, които са **дискриминационни** или **несправедливи** към определени групи.

Гарантирането на справедливост и смекчаването на пристрастията в ГИИ моделите е продължаващо предизвикателство.



# **Еволюция на невронните мрежи до големи езикови модели**





- » През последните няколко десетилетия езиковите модели претърпяха **значителен напредък**.
- » Първоначално основните езикови модели бяха използвани за задачи като **разпознаване на реч, машинен превод и извличане на информация**.
- » Тези ранни модели са конструирани с помощта на **статистически методи**, като n-грами и скрити модели на Марков.
- » Въпреки тяхната полезност, тези модели имаха **ограничения** по отношение на **точност** и **мащабируемост**.



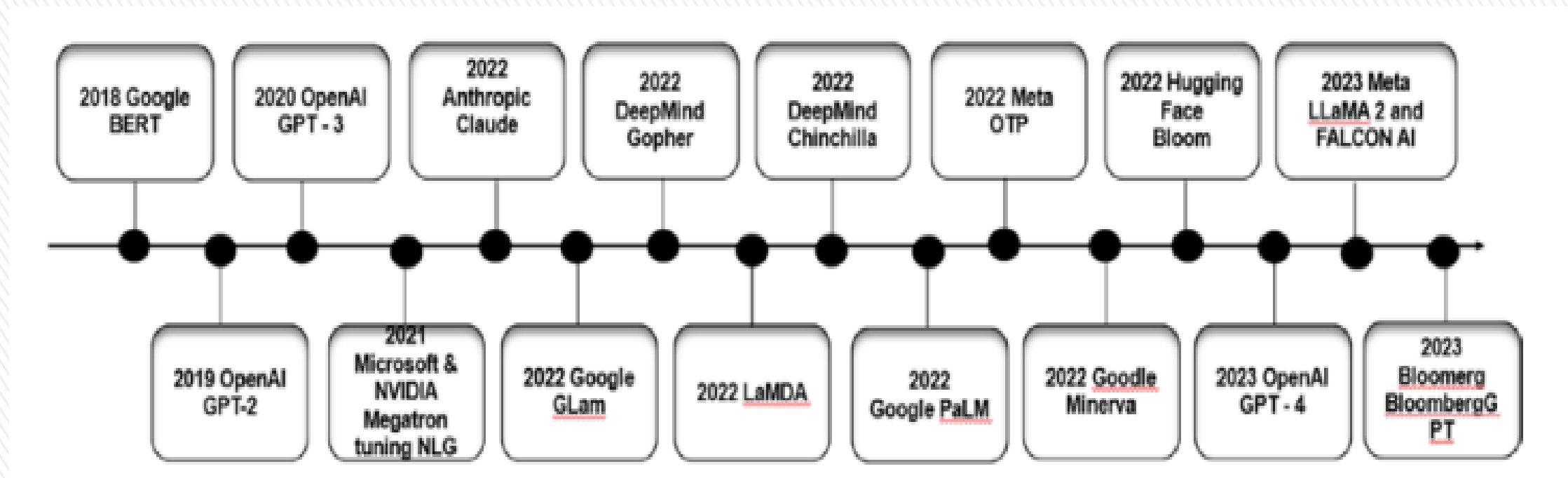
- » С въвеждането на **дълбокото учене**, невронните мрежи станаха **по-популярни** за задачи за езиково моделиране.
- » Сред тях, RNN и LSTM се очертаха като **особено ефективни избори**.
- » Тези модели превъзхождат при улавяне на последователни връзки в лингвистични данни и генериране кохерентен изход.



- » В последно време подходите, базирани на вниманието, илюстрирани от **трансформаторната архитектура** привлякоха значително внимание.
- » Тези модели произвеждат изход чрез **фокусиране върху специфични сегменти** от входната последователност, използвайки техники за **самонасочване**.
- » Техният успех е демонстриран в различни задачи за обработка на естествен език, включително езиково моделиране.



# Схема на вволюцията на езиковите модели





**Обработка на естествен език**



# Обработка на естествен език

- » Обработката на естествен език (ОЕЕ) е подобласт на ИИ и компютърната лингвистика, което се фокусира върху това да позволи на компютрите да **разбират, интерпретират и генерират човешки език**.
- » ОЕЕ има за цел да преодолее пропастта между човешката комуникация и машинното разбиране, позволявайки на компютрите да **обработват и извличат значение от текстови данни**.
- » Играе решаваща роля в различни приложения, включително **езиков превод, анализ на настроението, чатботове, гласови асистенти, обобщаване на текст** и др.



# Обработка на естествен език

- » Последните постижения в ОЕЕ се дължат на техники за **дълбоко учене**, особено чрез използване на **трансформаторни модели**, като:
  - > **BERT** (двупосочни енкодерни представяния от трансформатори)
  - > **GPT** (генеративен предварително обучен трансформатор).
- » Тези модели използват **широкомащабно** предварително обучение върху **огромни количества текстови данни** и могат да бъдат **фино настроени** за конкретни ОЕЕ задачи, постигайки най-съвременна производителност в широк спектър от приложения.



# Обработка на естествен език

- » ОЕЕ продължава да бъде бързо развиваща се област с непрекъснати изследвания и разработки, целящи да подобрят разбирането на езика, генерирането и взаимодействието между машини и хора.
- » Тъй като възможностите на ОЕЕ се подобряват, то има потенциала да **революционизира** начина, по който взаимодействаме с технологиите и да позволи по-естествена и безпроблемна **комуникация между хората и компютъра**.





# Токенизация

- » Токенизацията е процесът на **разделяне на текста на отделни думи или токени**.
- » Помага при **сегментирането** на текста и **анализирането** му на **по-подробно ниво**.
- » Пример:
  - > Вход: „I love to code in python“
  - > Токенизация: [„I“, „Love“, „to“, „code“, „in“, „python“].



# N-грами

- » В ОЕЕ n-грамите са мощна и широко използвана техника за извличане на контекстна информация от текстови данни.
- » По същество N-грамите са съседни последователности от n елемента, където елементите могат да бъдат думи, знаци или дори фонemi в зависимост от контекста.
- » Стойността на „n“ в n-грамове определя броя на последователните елементи в последователността.
- » Често използваните n-грами включват:
  - > униграми (1-грам)
  - > биграми (2-грама)
  - > триграми (3-грама) и т.н.



# Униграми (1-грами)

- » Униграмите са **отделни думи** в текст.
- » Те представляват отделни **лексеми** или **смислови единици** в текста.
- » Пример:
  - > Вход: „I love natural language processing.“
  - > Униграми: [“I”, “love”, “natural”, “language”, “processing”, “.”]



# Биграми (2 грами)

- » Биграмите се състоят от **две последователни думи** в текст.
- » Те осигуряват **смисъл на двойки думи и връзката между съседни думи**.
- » Пример:
  - > Вход: „I love natural language processing.“
  - > Биграми: [(“I”, “love”), (“love”, “natural”), (“natural”, “language”), (“language”, “processing”), (“processing”, “.”)]





# Триграми (3 грами)

- » Триграмите са **три последователни думи** в текст.
- » Те улавят **повече контекст** и дават представа за триплетите думи.
- » Пример:
  - > Вход: „I love natural language processing.“
  - > Триграми: [(“I”, “love”, “natural”), (“love”, “natural”, “language”), (“natural”, “language”, “processing”), (“language”, “processing”, “.”)]



# N-грами в езиковото моделиране

- » В задачите за езиково моделиране n-грамите се използват за **оценка на вероятността за дума, дадена в нейния контекст.**
- » Например, с биграми можем да преценим вероятността за дума въз основа на **предходната дума.**



# N-грами в класификация на текст

- » N-грами са полезни при **задачи за класификация на текст**, като например анализ на настроението.
- » Отчитайки честотите на n-грамите в положителни и отрицателни текстове, класификаторът може да научи **отличителните характеристики** на всеки клас.



# Ограничения на n-грамите

- » Въпреки че n-грамите са мощни при улавяне на локален контекст, те могат да **загубят глобалния контекст**.
- » Например биграмите може да **не са достатъчни**, за да разберат **значението на изречение**, ако някои думи имат **силна зависимост от други, разположени по-далеч**.





# Боравене с думи извън речниковия запас

- » Когато използваме  $n$ -грами, важно е да боравим с думи **извън речниковия запас** (думи, които не се виждат по време на обучение).
- » Могат да се използват техники като добавяне на **специален знак за непознати думи** или използване на  **$n$ -грами на ниво СИМВОЛ**.



# Изглаждане

- » N-грамовите модели може да страдат от **оскъдни данни**, особено когато се работи с n-грами от по-висок ред.
- » Техники за изглаждане като изглаждане на Лаплас (добавено) или изглаждане на Гуд-Тюринг могат да помогнат за справяне с този проблем.



- » N-грамите са ценен инструмент в ОЕЕ за улавяне на **локален контекст** и извличане на **значими характеристики** от текстови данни.
- » Те имат различни приложения в езиковото моделиране, класифицирането на текст, извличането на информация и др.
- » Въпреки че n-грамите предоставят ценна представа за структурата и контекста на текста, те трябва да се използват **заедно с други ОЕЕ техники** за изграждане на стабилни и точни модели.



# Езиково представяне и вграждания

- » Езиковото представяне и вграждане са основни концепции в ОЕЕ
  - > Включват трансформиране на думи или изречения в числови вектори.
- » Тези числени представяния позволяват на компютрите да разбират и обработват човешкия език, което улеснява прилагането на алгоритми за МУ към ОЕЕ задачи.





# Техники за вграждане

- » Word2Vec и GloVe са популярни техники за вграждане на думи - процес на представяне на думи като **плътни вектори** във **високомерно векторно пространство**.
- » Тези вграждания на думи улавят **семантичните връзки** между думите и се използват широко в задачи за обработка на естествен език.



# Word2Vec

- » Word2Vec е семейство модели за вграждане на думи, въведени през 2013 г.
- » Състои се от две основни архитектури:
  - > **CBOW**: предвижда целева дума въз основа на нейните контекстни думи:
    - + Приема набор от контекстни думи като вход и се опитва да предвиди целевата дума в средата на контекста.
    - + Ефективна и може да обработва множество контекстни думи в един кадър.
  - > **Skip-gram**: прави обратното на CBOW:
    - + Приема целева дума като вход и се опитва да предвиди контекстните думи около нея.
    - + Полезна за улавяне на връзки между думите
    - + Представя се по-добре при редки думи.



# Word2Vec

- » Word2Vec използва **плитка невронна мрежа** с един скрит слой, за да научи вграждането на думи.
- » Научените вграждания поставят **семантично подобни думи по-близо една до друга във векторното пространство.**



# GloVe

- » GloVe е друга популярна техника за вграждане на думи, въведена през 2014 г.
- » За разлика от Word2Vec, GloVe използва **матрица за съвместно срещане на двойки думи** за да научи вграждане на думи.
- » Матрицата на съвместно срещане представя **колко често две думи се появяват заедно в даден корпус**.





# GloVe

- » GloVe има за цел да **факторизира тази матрица** на съвместни събития за да получи вграждане на думи, които улавят **глобалните връзки** дума-към-дума в целия корпус.
- » Той използва както глобална, така и локална контекстна информация, за да създаде по-смислени представяния на думи.



# **Вероятностни модели**



# Вероятностни модели

- » Вероятностният модел на  $n$ -грам е прост и широко използван подход за езиково моделиране при ОЕЕ.
- » Той оценява вероятността за дума въз основа на **предходните  $n-1$  думи** в последователност.
- » „ $n$ “ в  $n$ -грама представлява **броя на думите**, разглеждани заедно като единица.
- » Моделът на  $n$ -грамата е изграден върху предположението на Марков, което предполага, че **вероятността** за дадена дума **зависи само от фиксиран прозорец на предишните думи**.



# N-грамно представяне

- » Входящият текст е разделен на последователни последователности от  $n$  думи.
- » Всяка последователност от  $n$  думи се третира като единица или  $n$ -грам.
- » Например, в биграмен модел ( $n=2$ ), всяка двойка последователни думи се превръща в  $n$ -грама.





# Броене на честотата

- » Моделът **брои** появляванията на всеки  $n$ -грама в данните за обучение.
- » Той следи колко често всеки специфична последователност от думи се появява в корпуса.



# Изчисляване на вероятности

- » За да предскаже **вероятността за следващата дума** в последователност, моделът използва **броя на n-грамите**.
- » Например, в биграмен модел, вероятността за дума се оценява въз основа на **честотата на предходната дума** (униграма).
- » Вероятността се изчислява като съотношението на броя на биграмата към броя на униграмата.



# Изглаждане

- » На практика  $n$ -грамният модел може да срещне **невидими  $n$ -грами** (последователности, които не присъстват в данните за обучение).
- » За справяне с този проблем се прилагат **техники за изглаждане**, за да се присвоят **малки вероятности** на невидими  $n$ -грами.



# Генериране на език

- » След като  $n$ -грамният модел бъде обучен, той може да се използва за **генериране на език**.
- » Започвайки с начална дума, моделът предвижда следващата дума въз основа на най-високите вероятности от наличните  $n$ -грами.
- » Този процес може да се **повтаря итеративно**, за да се **генерират изречения**.



# Скрит модел на Марков

- » Скритият модел на Марков (СММ) е друг важен вероятностен модел в езиковата обработка.
- » Използва се за моделиране на последователности от данни, които следват маркова структура, където основна последователност от скрити състояния генерира наблюдавани събития.
- » Терминът „скрити“ се отнася до факта, че не можем директно да наблюдаваме състоянията, но можем да ги направим изводи от наблюдаваните събития.
- » СММ се използват в различни задачи, като разпознаване на реч, маркиране на част от речта и машинен превод.





# Ограничения

- » Моделът на n-gram има ограничен контекст, като се вземат предвид **само предходните n-1 думи**, които може да не уловят **дългосрочни зависимости**.
- » Може да не улови ефективно семантичното значение или синтактичните структури в езика.



# Обобщение

- » Въпреки своята простота и ограничения, **n-грамният вероятностен модел осигурява полезна основа** за задачи за езиково моделиране
- » **Основополагаща концепция** за по-сложни езикови модели като **RNN** и **трансформаторни модели**.



**Езикови модели, базирани на невронни  
мрежи**



- » Базираните на невронни мрежи езикови модели направиха **значителен пробив** в ОЕЕ напоследък.
- » Тези модели използват **невронни мрежи**, които са изчислителни структури, вдъхновени от човешкия мозък, за обработка и разбиране на езика.



- » Основната идея зад тези модели е да се обучи невронна мрежа **да предсказва следващата дума** в изречение въз основа на **думите, които го предхождат**.
- » Като представя на мрежата голямо количество текстови данни и я обучава да разпознава модели и връзки между думите, тя се научава да прави **вероятностни прогнози** за това, коя дума е вероятно да последва.





» След като невронната мрежа бъде обучена на огромен набор от данни, тя може да използва научените модели за да **генерира текст**, да завърши изречения или дори да отговори на въпроси въз основа на контекста, който е научила по време на обучението.



» Чрез ефективно улавяне на връзките и зависимостите между думите в едно изречение, тези езикови модели **драстично подобриха способността на компютрите** да разбират и генерират човешки език, което доведе до значителен напредък в различни ОЕЕ приложения като машинен превод, анализ на настроението, чатботове и много други .



Input Layer ( $n_1, n_2, \dots, n_{\text{input}}$ )



Hidden Layer ( $n_3, n_4, \dots, n_{\text{hidden}}$ )



Output Layer ( $n_5, n_6, \dots, n_{\text{output}}$ )

„ $n_{\text{input}}$ “ представлява броя на входните неврони, всеки от които съответства на функция във входните данни.

„ $n_{\text{hidden}}$ “ представлява броя на невроните в скрития слой. Скритият слой може да има множество неврони, което обикновено води до по-сложни представяния на входните данни.

„ $n_{\text{output}}$ “ представлява броя на невроните в изходния слой. Броят на изходните неврони зависи от естеството на проблема - може да бъде двоичен (един неврон) или многокласов (множество неврони).



**Повтарящи се невронни мрежи (RNN) >**

- » RNN са вид изкуствена невронна мрежа, предназначена да обработва последователни данни поелементно, като същевременно поддържа **вътрешно състояние**, което обобщава **историята** на предишни входове.
- » Те имат уникалната способност да обработват входни и изходни последователности с променлива дължина, което ги прави подходящи за задачи за обработка на естествен език като синтез на език, машинен превод и разпознаване на реч.





- » Ключовата характеристика, която отличава RNN, е способността им да улавят **времеви зависимости чрез вериги за обратна връзка**.
- » Тези цикли позволяват на мрежата да използва информация от предишни изходи като входове за бъдещи прогнози.
- » Тази подобна на паметта способност позволява на RNN да **запазят контекст и информация от по-ранни елементи в последователността, влияейки върху генерирането на последващи изходи**.



# Предизвикателства

- » RNN обаче се сблъскват с някои предизвикателства.
- » Проблемът с **изчезващия градиент** е важен проблем, при който градиентите, използвани за актуализиране на теглата на мрежата, стават **много малки** по време на обучение, което затруднява ефективното изучаване на дългосрочни зависимости.
- » Обратно, проблемът с **експлодиращия градиент** може да възникне, когато градиентите станат **твърде големи**, което води до **нестабилни актуализации** на теглата.



# Ограничения

- » Освен това, RNN по своята същност са **последователни**, обработващи елементи един по един, което може да бъде **изчислително скъпо** и **предизвикателно** за паралелизиране.
- » Това ограничение може да **възпрепятства** тяхната **мащабируемост** при работа с големи набори от данни.



# Усъвършенстване

- » За справяне с някои от тези проблеми са разработени по-усъвършенствани варианти на RNN, като дълга краткосрочна памет (LSTM) и затворена повтаряща се единица (GRU).
- » Тези варианти се оказаха по-ефективни при улавяне на дългосрочни зависимости и смекчаване на проблема с изчезващия градиент.



# **Дълга краткосрочна памет (LSTM)**





- » Мрежите с дълга краткосрочна памет (LSTM) са **специализиран тип архитектура** на повтаряща се невронна мрежа (RNN), предназначена да се справи с проблема с изчезващия градиент и да улови дългосрочни зависимости в последователни данни.
- » Въведени през 1997 г. и оттогава придобиха популярност за моделиране на последователни данни в различни приложения.



- » Ключовата характеристика, която отличава LSTM от традиционните RNN, е способността му да включва **клетка с памет**, която може избирателно да запазва или забравя информация във времето.
- » Тази клетка с памет се управлява от три порти: входна врата, забравена врата и изходна врата:
  - > Входната врата регулира потока от нови данни в клетката на паметта, позволявайки ѝ да реши коя нова информация е важна за съхраняване.
  - > Портата за забравяне контролира запазването на текущи данни в клетката на паметта, позволявайки ѝ да забрави неподходяща или остаряла информация от предишни времеви стъпки.
  - > Изходният гейт регулира потока информация от паметта клетка към изхода на мрежата, като се гарантира, че съответната информация се използва при генериране на прогнози.



- » Този механизъм позволява на LSTM да улавя дългосрочни зависимости в последователни данни, което го прави особено ефективен за задачи, включващи обработка на естествен език, като езиково моделиране, машинен превод и анализ на настроението.
- » Освен това, LSTM са успешно приложени в други задачи като гласово разпознаване и надписи на изображения.



- » Чрез справяне с проблема с изчезващия градиент и осигуряване на по-добър начин за запазване и използване на важна информация във времето, LSTM мрежите се превърнаха в мощен инструмент за обработка на последователни данни и значително подобриха производителността на различни приложения в областта на МУ и ИИ.



# **Gated Recurrent Unit (GRU)**





- » GRU (gated recurrent unit) мрежите са тип архитектура на невронни мрежи, често използвани в ДУ и ОЕЕ.
- » Те са предназначени да се справят с проблема с **изчезващия градиент**, точно като LSTM мрежите.



- » Подобно на LSTM, GRU също включват механизъм, позволяващ на мрежата **избирателно да актуализира и забравя информацията с течение на времето.**
- » Този механизъм за стробиране е от решаващо значение за улавяне на дългосрочни зависимости в последователни данни и прави GRU ефективни за задачи, включващи език и последователни данни.



- » Основното предимство на GRU пред LSTM се крие в техния **по-прост дизайн** и **по-малко параметри**.
- » Тази простота прави GRU **по-бързи за обучение** и **по-лесни за внедряване**, което ги прави популярен избор в различни приложения.



- » Докато GRU и LSTM имат подобен механизъм, ключовата разлика е в броя на портовете, използвани за регулиране на потока от информация.
- » LSTM използват три порти: входен, за забравяне и изходен.
- » За разлика от тях GRU използват само два порта: за нулиране и за актуализиране.



- » Портът за нулиране контролира коя информация да се отхвърли от предишната времева стъпка, докато портът за актуализиране определя колко от новата информация да се добави към клетката с памет.
- » Тези два порта позволяват на GRU да контролират ефективно потока от информация без сложността на наличието на изходен порт.





- » Мрежите на GRU са ценно допълнение към семейството на повтарящи се невронни мрежи.
- » Техният по-прост дизайн и ефективно обучение ги правят практичен избор за различни задачи, свързани с последователност, и те са доказали своята висока ефективност при обработка на естествен език, разпознаване на реч и други приложения за последователен анализ на данни.



# **Мрежи енкодер-декодер**



# Архитектура

- » Архитектурата на енкодер-декодер е вид невронна мрежа, използвана за обработка на **последователни задачи** като **езиков превод, чатбот, аудио разпознаване и надписи на изображения**.
- » Състои се от два основни компонента:
  - > Енкодерна мрежа
  - > Декодерна мрежа.



# Контекстен вектор

- » По време на езиков превод, например, мрежата на енкодера обработва входното изречение на изходния език.
- » Преминава през изречението дума по дума, генерирайки представяне с фиксирана дължина, наречено **контекстен вектор**.
- » Този контекстен вектор съдържа важна информация за входното изречение и служи като **съкратена версия на оригиналното изречение**.



# Изход

- » След това контекстният вектор се подава в мрежата на декодера.
- » Мрежата на декодера използва контекстния вектор заедно с неговите вътрешни състояния, за да започне **генерирането на изходната последователност**, която в този случай е **преводът на целевия език**.
- » Декодерът генерира една дума наведнъж, като използва вектора на контекста и предварително генерираните думи, за да предвиди следващата дума в превода.





**Seq2Seq модели**



# Архитектура

- » Моделите от последователност към последователност (Seq2Seq) са вид архитектура за ДУ, предназначена да обработва входни последователности с **променлива дължина** и да генерира изходни последователности с **променлива дължина**.
- » Те станаха популярни в задачи за ОЕЕ като машинен превод, обобщаване на текст, чатботове и др.
- » Архитектурата се състои от:
  - > Енкодер
  - > Декодер
  - > И двата са RNN или трансформаторни модели.



# Енкодер

- » Енкодерът взема входната последователност и я обработва дума по дума, създавайки представяне с фиксиран размер (контекстен вектор), което кодира цялата входна последователност.
- » Контекстният вектор улавя съществената информация от входната последователност и служи като **първоначално скрито състояние** за декодера.



# Декодер

- » Декодерът приема контекстния вектор като първоначално скрито състояние и **генерира изходната последователност** дума по дума.
- » На всяка стъпка той **предвижда следващата дума** в последователността въз основа на контекстния вектор и предишните генерирани думи.
- » Декодерът е обусловен от входа на енкодера, което му позволява да произвежда значими изходи.



# Механизъм на внимание

- » В стандартната енкодер-декодер архитектура процесът започва с кодиране на входната последователност във векторно представяне с фиксирана дължина.
- » Тази стъпка на кодиране **кондензира цялата информация от входната последователност в единичен вектор с фиксиран размер, известен като контекстен вектор.**





# Механизъм на внимание

- » След това декодерът приема този контекстен вектор като вход и **генерира изходната последователност** стъпка по стъпка.
- » Декодерът използва контекстния вектор и неговите вътрешни състояния, за да **предвиди всеки елемент** от изходната последователност.



# Механизъм на внимание

- » Докато този подход работи добре за **по-къси входни последователности**, той може да се сблъска с предизвикателства при работа с **дълги входни последователности**.
- » Кодирането с фиксирана дължина може да доведе до **загуба на информация**, тъй като контекстният вектор има ограничен капацитет да улови всички нюанси и детайли, присъстващи в по-дълги последователности.



# Механизъм на внимание

- » По същество, когато входните последователности са дълги, кодирането с фиксирана дължина може да се затрудни да запази цялата релевантна информация, което потенциално води до **по-малко точна или непълна изходна последователност**.



# Механизъм на внимание

- » За справяне с този проблем са разработени по-усъвършенствани техники, като например използване на механизми за внимание в архитектурата на енкодер-декодер.
- » Механизмите за внимание позволяват на модела да се фокусира върху конкретни части от входната последователност, докато генерира всеки елемент от изходната последователност.
- » По този начин моделът може ефективно да обработва дълги входни последователности и да избягва загуба на информация, което води до подобрена производителност и по-точни резултати.



# Механизъм на внимание

- » Механизмът за внимание **изчислява резултатите за внимание** между скритото състояние на декодера (заявка) и скритото състояние на всеки енкодер (ключ).
- » Тези оценки на вниманието определят **важността на различни части от входната последователност** и след това контекстният вектор се формира като **претеглена сума** от скритите състояния на енкодера, с тегла, определени от оценките на вниманието.





# Обобщение

- » Архитектурата Seq2Seq, със или без внимание, позволява на модела да обработва последователности с променлива дължина и да генерира значими изходни последователности, което го прави подходящ за различни ОЕЕ задачи, които включват последователни данни.



**Трансформатори**



# Архитектура

- » Трансформаторната архитектура е въведена през 2017 г. като **новаторски дизайн на невронни мрежи**, широко използван в задачи за обработка на естествен език като категоризиране на текст, езиково моделиране и машинен превод.
- » В основата си архитектурата **прилича на модел енкодер-декодер**.
- » Процесът започва с енкодера, който взема входната последователност и генерира нейно скрито представяне.
- » Това скрито представяне съдържа съществена информация за входната последователност и служи като контекстуализирано представяне.



# Архитектура

- » След това скритото представяне се предава на декодера, който го използва за генериране на изходната последователност.
- » Както енкодерът, така и декодерът се състоят от множество слоеве на невронни мрежи за самонасочване и подаване напред.
- » Слой за самовнимание изчислява теглата на вниманието между всички двойки входни компоненти, позволявайки на модела да се фокусира върху различни части от входната последователност, ако е необходимо.
- » Теглата на вниманието се използват за изчисляване на претеглена сума на входните елементи, като предоставят на модела начин за селективно включване на подходяща информация от цялата входна последователност.



# Архитектура

- » Слой за подаване допълнително обработва изхода на слоя за самовнимание с нелинейни трансформации, подобрявайки способността на модела да улавя сложни модели и връзки в данните.
- » Дизайнът на трансформатора предлага няколко предимства пред предишните архитектури на невронни мрежи.





# Ефективност

- » Позволява **паралелна обработка** на входната последователност, което я прави по-бърза и по-ефективна в изчислително отношение в сравнение с традиционните последователни модели.



# Интерпретируемост

- » Теглата на вниманието могат да бъдат **визуализирани**, което ни позволява да видим върху кои части от входната последователност се фокусира моделът по време на обработката, което улеснява разбирането и тълкуването на поведението на модела.



# Глобален контекст

- » Трансформаторите могат да разглеждат цялата входна последователност **едновременно**, което им позволява да улавя дългосрочни зависимости и да подобрява производителността на задачи като машинен превод, където контекстът от цялото изречение е от решаващо значение.



# Обобщение

- » Трансформаторната архитектура се превърна в доминиращ подход при обработката на естествения език и значително напредна в съвременните технологии в различни задачи, свързани с езика, благодарение на своята ефективност, интерпретируемост и способност да улавя глобалния контекст в данните.



**Големи езикови модели (LLM)**





# Особености

- » Големите езикови модели (LLM) се отнасят до клас усъвършенствани модели за ИИ, специално проектирани да обработват и разбират човешкия език в широк мащаб.
- » Тези модели обикновено се изграждат с помощта на техники за ДУ, особено базирани на трансформаторни архитектури
- » Обучават се върху **огромни количества** текстови данни от Интернет.



# Характеристики

- » Ключовата характеристика на големите езикови модели е способността им да научават сложни модели, семантични представяния и контекстуални връзки в естествения език.
- » Те могат да генерират човешки текст, да превеждат между езици, да отговарят на въпроси, да извършват анализ на настроението и да изпълняват широк набор от задачи за обработка на естествен език.



# Известен пример

- » Един от най-известните примери за големи езикови модели е серията GPT (Generative Pre-trained Transformer) на OpenAI, която включва модели като GPT-3.
- » Тези модели са предварително обучени върху масивни масиви от данни и могат да бъдат фино настроени за конкретни приложения, което им позволява да се адаптират и да се отличават с различни задачи, свързани с езика.



- » Възможностите на големите езикови модели донесоха значителен напредък в обработката на естествения език, правейки ги инструментални в различни индустрии, включително поддръжка на клиенти, генериране на съдържание, превод на език и др.
- » Въпреки това, те също повдигат важни опасения относно етиката, пристрастията и злоупотребата поради техния потенциал да генерират човешки текст и да разпространяват дезинформация, ако не се използват отговорно.
- » Някои забележителни примери за LLM включват:



# GPT

- » **GPT** е четвъртата версия на серията Generative Pretrained Transformer на OpenAI.
- » Той е известен със способността си да генерира човешки текст и е демонстрирал умение да отговаря на въпроси, да създава поезия и дори да пише код.





# BERT (Двупосочни енкодерни представления от Transformers)

- » Разработени от Google, **BERT** е основен LLM, който улавя контекста от двете посоки на въведения текст, което го прави умело в разбирането на езиковите нюанси и взаимоотношения.
- » Той се превърна в основен модел за широк спектър от ОЕЕ задачи.



# T5 (Text-to-Text Transfer Transformer)

- » Също така разработен от Google, T5 подхожда към всички ОЕЕ задачи като проблеми от текст към текст.
- » Тази обединяваща рамка показва изключителна производителност при задачи като превод, обобщение и отговаряне на въпроси.



# RoBERTa на Facebook

- » Оптимизирана версия на BERT, която е постигнала най-съвременни резултати в различни OEE бенчмаркове.
- » Той се основава на архитектурата и процеса на обучение на BERT, като допълнително подобрява възможностите за разбиране на езика.



# Заклучение

- » Тези LLM демонстрират напредък в обработката на естествения език, разширявайки границите на това, което ИИ моделите могат да постигнат в задачи като генериране на език, разбиране и превод.
- » Тяхната гъвкавост и най-съвременна производителност ги направиха ценни активи в приложения, вариращи от чатботове и езиков превод до анализ на настроенятия и генериране на съдържание.
- » С напредването на изследванията в тази област можем да очакваме да се появят още по-сложни и способни LLMs, които продължават да революционизират областта на ОЕЕ.





Благодаря за вниманието!

