What is generative AI?

→ Generative AI is the class of Artificial Intelligence where these AI do not use to predict According to the patterns from the data that they have learnt what we are actually doing, we were feeding our models with lots of data, and they are using that model for our predictions, but generative AI is not such a case. generative AI is the class of AI where we feed the data but that model actually used to create new data, earlier using machine learning learning, deep learning or NLP we were not be able to create new content or new data or new images. we were only able to do classifications, prediction, sentiment Analysis.

Generative AI such a ~~case~~ class of AI where it is use to create new data, new images, new text, new videos.

How it produces those kind of things?

→ It produces those kind of things, just by learning the patterns of the data.

Generative AI:→ It contents two words Generative & AI

Generative → creating/generating new content

AI → using Artificial Intelligence

Generative → create new content (Audio, code, images, text video)

Artificial Intelligence → automatically using a computer program.

→ Generative AI encompasses various approaches, including generative Adversarial networks (GAN), Auto regressive models, variational autoencoders (VAEs).

When we talked about generative AI. that does not mean the LLM one or large language model or chatgpt or the mixture.

generative AI means Any model which can generate new content, new thing or new data. and in this process we have 3-4 kinds of things inside a generative AI.

Autoregressive model:→ This is a decoder model which is use by gpt, most of the models which generates next-text.

Generative Adversarial Networks:→

Generative means which is a generator which generate new content

Adversarial means there is some discreminator, Descriminator means there will be some neural network which will check generated content of generator. There will be two neural network, one will be for generator and other will be for descriminator.

Generator ek aisa neural network hai, jisko agar kuchh input data diya jaayega aur input data dene ke baad wo kuchh naya cheez generate karega, jab wo naya cheez generate kar lega tab yaha par discriminator aayega jisko (adversorial) bhi bolte hai, wo ye check karega ki jo bhi naya data generator ne release kiya hai wo real data se align kar raha hai ya kitna ka diff. aa raha hai. according to that it will give some scores and in this process it tries to learn. generating ⟺ Discreaminating

Jaise hi generator (Neural Network) naya data generate karta hai according to input data, aur discreaminator ne usko pahchaan liya, jo data generate hua hai, wo real ke jaisa hi hai, tab to ~~generat~~ generator fail ho gaya kyuki discreminator ne kaise pahchaan liya.

Actually generator ko aisa cheez generate karna hai jo ho to real hi, lekin discreminator ko lage ki real nahi hai, fake hai. ess tarike se generator ek naya aur naye-2 cheez aur real type ka cheez generate kar payega

Variational autoencoders (VAEs) :→ ye variational hai aur autoencoders hai.

autoencoders :→ Eske paas kuchh encoded input honge jo ki embeddings me honge ya vectors me honge. jisme kuchh information chhipa hoga aur ye ek neural network use karega aur kuchh decoder architecture use karega kis, uss embeddings me, uss vectors me jo bhi information chhipa hua hai usko wo decode karke ek data output kar sake.

Autoencoders data ko as it is generate karta hai lekin, variational encoders uske jaisa sara output ko learn kar jayega → 2, 2', 2'', 2''' → (jitne bhi uske variations ho sakte hain, wo saare cheez ko learn karega).

usecase :→ Drug discovery

=) Generative AI is not a new concept, we have google translation. It was, day before LLM come into existance. This awas also an example of gen AI. uss time jo method use hua usko LLM nahi kahte uska naam (statistical machine Translation (SMT)) hai

SMT :→ Statistical method use karte the aur uske a/c jo bhi language hai usko translate karte the.

Jo statistical Translation model hain esme 3 types ke model use hote the

(i) Alignment model
(ii) Translation model
(iii) Language model.

i) Alignment model :- Harek ~~model~~ Word ko wo corresponding English Word se usko create karta tha. aligned karta tha.

eg:- le → The, chat → cat, noir → black ek

ye alignment huge data par trained tha jisme ek side koi aur language. koi aur language the dusre side koi aur language.

ii) Translation model :- esme ek sequence of character lete hain aur usko english me convert kar deta hai.

eg-
le chat noir → the black cat.

iii) Language model :- ye aise output generate karta hai jo ki natural sounding ki taraf ho aur ye sentence aur semantic ko capture kar paaye.

Earlier we are also having, Siri, Cortana, Alexa, Bixby, all are the example of generative AI → They were using Hidden Markov's models (HMMs) for speech recognition.

Hidden Markov models :- Hidden markov models are used for task like speech recognition where the goal is to understand spoken words.

iske alawa esme supervised learning Algorithm bhi use hua

Supervised learning :→ Eska matlab ye hai ki jaise hi user ne koi query puchha to wo model jo hai ess tarike se train tha ki wo classify kar pata ki ye query kis category ka hai

NLP
Feature Engg.

Working of GAN's

There is a real face
This is a generator neural network, this neural Network will give some input some data inside the generator so generator will create face or new data according to the data which has been feed upon once the face has been generated now, it will be given to Discriminator neural network and descriminator neural network will compare Real face and generated face. and it will give some score according to the things weather it is a real or fake.

~~Sumar~~
Summary:-

GANs me hota ye hai ki ek neural Network hai jisko generator bolte hain. esko humlog kuchh data dete hain uske baad ye naya data generate karta hai ab waha par Descreatrinator ka kaam hota hai ki uska jo generated data hai wo diya jata hai real data hai wo diya jata hai. ab descriminator esme kya karta hai ki esko descriminate karne ka koshish karta hai generated data real hai ki fake hai agar wo generated data ko jaisa usne generate kiya hai agar wo real pahchaan liya to usko -1 assign karega, agar fake pahchana to zero (0) value dega. esi duration me training hota hai

Jyada koshish ye kiya jata hai ki generator ess tarah ka data generate kare ki descriminator usko pahchaan naa sake tab generator ko award milega aur descriminator ko penalty.

Auto encoders:- Auto encoders kya karta hai ki eske paas 2 neural network hote the ek hai encoder dusra decoder encoder me humlog data dete hain data dene ke baad, ek ye possibility hai ki data higher dimension ka ho, eske baad data ka dimension reduce karne ke baad ye ek latent state banata tha

Latent State:- Jo bhi data input me numbers me diya ~~jata~~ jayega, agar wo higher dimension ka hai, to uske features ko retain karte hue usko low dimension me lekar aata hai, latent state esi ko bolte hain.

uske baad uss jo reduced dimension hain jiske andar wo saare features hain, usne koi bhi features loss nahi kiye hain koi bhi information loose nahi hua hai during the dimensionality reduction usko usne phir decoder me diya, phir decoder kya karta hai, jo cheez usne encoded tha wo usko decode kar ke de deta hai.

eg:-

Input me hamne M.S. Dhoni ka image diya uske baad hum esko no. me convert karenge uske baad uska latent state banega, esme uske dimensions ko reduced kar ke hum uske features ko provide karte hue usko latent state me store karwa liye. phir usko ek decoder me denge jo ~~bhi~~ khud bhi ek neural network hai, wo decoder usko wahi cheez decode kar ke, jaisa bhi hamara input tha as it is return kar dega.

Variational AutoEncoders:→

eg→ Jaise hi maine input me M.S.Dhoni ka image paas kiya esne usko no.'s me convert kiya. After conversion Dimension Reduction hua waha par uske features ko retain karte hue latent state banaya fir usi reduced dimension waale features ko decoder me diya. ab ye variational auto encoders kya karega ki ye sirf M.S Dhoni ke features ko hi learn nahi karega, balki waisa jitna bhi images hain, usi variety ke, usse saare variations ko learn karega aur wo ess tarike ka ability khud me produce kar lega, ki sirf M.S Dhoni jaisa image nahi, balki uss se diff-2 kind of variation create karne ke kaabil ho jayega. that is variational Autoencoder.

Architecture:

Encoder Network ⟶ Latent state ⟶ Decoder Network.

Autoregressive Models

ye two words ka combination hain Auto + Regressive Regressive means - jo bhi past data hain usi past data ko regress kar dega / learn kar lega, automatically ek new text generate karega. ye bhi ek statistical model ka ek type hai, eska use time series analysis me bhi ho raha hai, aur mostly abhi ke time par sequence generation task me use kar rahe hain. esi ke upar based hai large language model.

Large language model:→ Aisa model jo language ko ess tarike se represent kare / generate kare jo ki coherent bhi ho aur contextualy releavent bhi ho esiliye esko language model bola jata hai.

Basically, LLM ek aisa AI hai jo Ke Neural Network ya Deep learning technique ko use karta hai taaki wo human type text generate kar sake.
Actually LLM ko ess tarike se trained kiya gaya hai ki usme usko next word predict karane ka capacity bhi diya gaya hai. Sabhi LLM aise trained hain ki wo next word predict kar sakta hai, looking over the past data. But aaj ke llm's, Q&A, summarizations machine translations, bhi karta hai. Lekin esko trained kiya gaya tha sirf next word prediction ke liye. So ye downstreamed task kaise karta hai. for example machine translation, Q&A?

Jab hamara model ye samajh leta hai ki how to produce next word, then usko fine-tuned kiya jata hai down-streamed task ke liye. fine-tuned ka matlab jaise hum apne llm ko train kar diya, uske baad wo next word predict kar raha ho, Q&A bhi kar raha hai summarize bhi kar raha hai ess sab ke liye jo technique use hua usko kahte hai supervised-fine tuning

Supervised fine tuning → esme large corpus of data liya jata hai jaha par Q&A means ek side question & ek side Answer aise train kiya jata hai, jab ess tarah ke data par train kiya jata hai to eske andar Q&A ka ability bhi aa jata hai
Similarly machine translation ke data par bhi train kiya jata hai.

So downstreamed task ke liye jo bhi aapka base model tha (pretrained model) jo ki next word predict kar raha tha usko phir aapne fine-tuned kiya downstreamed task ke liye like Q&A, MT, etc.

eg: gpt ye sabhi kaam kar raha hai

Today & Till date transformer is the base model of any llm.

So understanding the basic architecture is v. imp to use the llm in your use cases. once you don't know the basic architecture of transformers you can use the Llm in your use cases but you will not have a great catch / knowledg may be you can say that may be you can't be able to fine-tune model according to your use cases.

Transformer Architecture me 2 cheeze hain. ek encoder aur ek decoder.

Encoder:→ Jo bhi hum textual data input me dete hain vsko ye vector me convert karta hai. aur vector me convert karne ke baad jitne bhi vske meanings & information hain vsko collect kar ke ek final content vector return kar deta hai.

Sabse pahle hamne input me kuchh pass kiya → phir vska embedding banaya, embedding means (No. me convert kiya) → eske baad positional encoding kiya gaya mean Kon sa word Kis word ke baad aaya hai. (ye bahut matter karta hai kyuki humlog sequential data par dealing kar rahe hain. agar hum word ka position change kar denge to complete meaning bhi change ho jayega). esi wajah se positional Encoding kiya jata hai

prob:→ Ham kaise pata karenge ki agar two similar type ke word two diff scenario me use ho raha hai

eg→
The robber went to bank and rob the bank.
The robber went to the bank of river and looking at a tree

Yah par hame positional encoding se meaning define nahi ho payega

sol^n → Selb Attention Mechanism / Attension mechanism

Attention mechanism → Harek word ka relation harek word se khojta hai define karta hai aur ek value / output generate karta hai

eske baad Add & Norms aata hai Add & Norm ka matlab ye hai ki jo bhi hame output aaya Attention se usme kuchh cheez add kar ke normalize karte hain for better output. eske baad Feed Forward Neural Network aata hai (ye ek bahut bara neural Network hai jo ki data ke pattern ko samjhta hai). eske baad phir Add & Norm (kuchh cheeze add kiya pichhle output par phir usko normalize kar diya). Finally encoder yaha par aapko ek context vector de dega. ab ess context vetor ko decode karna hai aur next word predict karna hai decoder ka bhi same architecture hai but decoder encoder se ess tarah se diff. hai ki decoder me naya word predict hota hai. Encoder me naya word predict nahi hoga. encoder me aapne jo text diya hai usika embedding hoga usika pos. encoding hoga usika context samjha jayega usika pattern samjha jayega finally usika vector generate hoga. jo uske meaning ko rakha hoga. jisko humlog context vector bolte hain agar usko naya data/text generate karna hai to uss Context vector ko hame use Decoder me dena hoga.

ek Nx layer me multi-head Attention Add Norm — Feed Forward (Add & Norm), aisa Nx layer open AI ke GPT me 96 hain. harek layer me feed forward network hai. aur uss harek feed forward network me bahut saare hidden layer hain.

Encoder: is Context Veetor generate karega jo ki uske meaning ko rakha hoga. aur sabse multi-head attention hota hai multihead attention context ko samjhata hai, jaise ek hi word do context me aa gaya to multihead attention kaam aata hai. feed forward network kya hai → neural Network hai large no. of hidden layers jo ki data ke pattern ko samajhta hai.

Decoder: is New text generate karega.

Neural Network ka first layer input layer kahlata hai jo last layer hai wo output layer kahlata hai bich bale layer ko hidden layer kahte hai

eg→ ek house price prediction model banate hain. jisme kuchh features ke saath hame price predict karna hai

Ham saare features ko input layer ke perceptron me bheja $(x_1, x_2, x_3, \ldots, x_n)$ sabka relation next hidden layer se hoga aur wo hidden layer ke harek perceptron se karega jis duration me ye sabki se relate karega uss time me weight initialize kiya jayega. uss time me (x_1) feature tha input layer ka uska kitna importance hai jab wo dusre (hidden layer) perceptron me jaa raha hai ye weight (w_1) batayega. agar uska weightage jyada hai to matlab wo feature jyada imp. hai for price prediction. jab saare weightage assign ho jaate hain fab hame ye pata chal jata hai ki kon sa column kitna imp. tha hamare price prediction ke liye. eske baad ek weighted sum next function me jata hai $(w_1x_1 + w_2x_2 \ldots)$ uske baad esme ek activation function hota hai ye activation function ek value/output generate karega jo ye predict karega ki house ka ess features aur ess weight ke a/c kitna hona chahiye esme ek bias bhi add kiya jata hai. bias already saare perceptron me already included rahta hai:

Bias term provide neural networks with the flexibility to model complex relationships in data by shifts & adjustments in the activation functions and intercepts

Jo bhi hamne weight initialize kiya tha ye jaruri nahi hai ki sahi ho ho sakta hai ki no of bedrooms ko imp. feature maan ke jyada weightage de diya ho lekin actual me price ~~no~~ of bedroom ke badle size of bedroom par depend kar raha ho. esi cheez ko learn karta hai during back propogation. aur phir weight change karta hai

Activation function:- ⊙ Jo bhi hamara data hai uske andar ke non-linearity ko capture karta hai

Multihead Attention:- multi head me jo multi head hai esme ek head kya kaam karege subject(cat) uska content dekhega aur uske meaning ko dhundhega at a same time dusra head dusre word (sits) verb/action wo mat se kaise related hai kya content hai eska. ess tarike se saare head apna apna words ke reation ko identify karta hai to sab apna ek score generate karta hai. sabki head ka apna ek score hota hai. finally uska eka avg. calculate kar ke aage forward kiya jata hai

eg:- The cat sits on mat.

Masked multi head attention:- masking kya karta hai ki jab self attention kaam ho raha hota hai to wo sirf previous waale token ko consider karne deta hai jo token aap attention ke liye use kar rahe ho vss token se aage wale token ko wo mask kar deta hai

output layer me jo probability hai usko bhi select karne ke two methods hain. ek hai greedy decoding and one is beam search

Greedy decoding:- Aapko kuchh words milenge next word ke taur par uski probability value milegi aur aapko select karna hai jiski probability value jyada hai aapko usi word ko as a next word predict karna hai esiko greedy decoding bolte hain.

(11) Beam Search :- esme updated sequence banata hai phir uske baad ek probability ka multiply karta hai aur ek combine probability ki value nikal ke ek final decision leta hai ki kon se sentence to reuchna hai

Tokenization & positional Embedding

Tokenization

Tokenization ek process hai jisme aap apne input sequence ko divide karte ho by the basis of character, word, sentence. aap apne rules/need ke hisaab se karte ho

Training of GPT

(1) Unsupervised pre training :- Hamne simply data ko liya aur feed kar diya. ab model khud se bina human labelled ke/bina human annotation ke model khud se data ke pattern ko samjhega, usko structure karega relationship ko samjhega, without any guidance, without any external sources aur phir uske pattern ko samajh ke ek apna jo model hai wo ready karega. esi process ko un-supervised pre training kahte hain.

(11) Self Supervised learning :- esme ham koi sentence input me diye, ess sentence ke kisi bhi word ko ab truncate kar denge ab usko predict karne ka try karenge. ESS process ko bolte hain self supervised learning. khud ke supervision me bina kisi external efforts ke. khud ke supervision me jab model train hota hai ya model learn karta hai to esko kahte hain self supervised learning.

Abhi tak ye model bahut achha nahi hua hai. abhi tak hum next word prediction kar paa rahe hain. masking ke baad bhi prediction kar paa rahe hain. esme kuchh error bhi aayga hamne error ko minimize kiya ab performance sahi hai, lekin eske aage kuchh downstreamed task the eg- Q&A, summarization, machine Translation ess sab ke liye model ko train karna hoga.

abhi tak hamara model/machine unsupervised se Data ke pattern ko samjha hai supervised se data ke next word prediction ko samjha hai masking ko billing karna samjha hai. Q&A, MT -ess sab ke liye ab apne model ko fine tune karna hoga. ab eska fine-tuning supervised type se hoga.

ab hamare paas jo data hoga wo labelled hoga jisme ek side questions honge aur ek side Answers honge.

Same machine translation me bhi hamare paas data hoga jisme ek side koi lang hoga dusre side phir koi ek aur language. hoga

GPT Parameters

→ n_{params} – 125m (matlab ye hai ki esme kitne weights & bias hain.)

→ n_{layers} → Decoder Architecture me kitna layer hain. harek layer me → ek feed forward neural network hoga aur ek attention mechanism hoga.).

→ d_{model} → Embedding ka dimension kitna hai?

→ n_{head} → ek attention mechanism me kitne head hain (multihead)

→ d_{head} → This refers to the dimensionalities of the queries keys & values used in each attention head. Typically d_heads is calculated as d_model divided by n_heads

batch_size:- This refers to the no. of input examples (sequences) processed in parallel during each training iteration. A larger batch size can lead to faster training but may require more memory.

GPT Playground

Temperature:- Temp. ye batata hai ki hamara jo model hai wo kitna random output generate kar sakta hai (output me randomness lata hai:)

eg → hamne koi question input kiya aur temp:=0 assign kiya hua hai to answer aayega kuchh output me agar again same question aur temp:=0 o/p me same answer aayega. lekin agar temp ka value change kar denge to hamesha output me variation aata rahega.

max= length:- max= hum kitna token generate karna chahte hain.

model:- Specifies which version of the gpt model you are using/want to use.

stop sequences:- wo kitna token generate karne ke baad rok dega ye parameter hum esme pass karte hain.

Imp Top P :- ye ek nucleas sampling data hai

hamare paas harek word ka prob. aata hai for next word. eske baad hum uska cumulative prob. nikalte hain. aur hamne apna top p ka parameter set kar diya hai=x jaise hi cumulative prob. ka value top p ke parameter ko cross karega usko wo nucleus word select kar leta hai