Dense Vector

- a dense vector is a numeric repersentation where most (or all) dimentions have non-zero values, it encodes information smoothly across all dimensions, like this: Ex [0.12, -0.37, 0.44, 0.05, ... 0.1]
this is what you get from embeddings or Nural Network layers

- Why NN work better with them? 1) continuity: small input changes lead to small output changes, making it easier to learn gradients 2) information spread: Every dimention carries some information (vs sparse vectors with mostly zeros) 3) efficient learning: Dense representations let NNs capture complex patterns and relasionships through weighted combinations 4) Differentiability: works naturaly with backpropigation since everything is continous and dense

How: models do this automatically since every dim has a non zero continous value
• if you want to ensure it use nerual embeddings (eg transformers), not one-hot or sparse bag of words

(instruction tuned models can fix hallucinations using thinking models)

Base Models vs instruction tuned models (Both still LLM)

- Base models (Mistral, LLaMA, Qwenn etc): are foundational language models trained purely with self-supervised learning on alot of text for ex predicting the next token in asequence, this means the model just learns to model Natural text, not to follow instructions or behave helpfully
Ex "what is Quantum computing" $\xrightarrow{ans}$ "Quantum computing, also known as..." basically just continuing your text (next word predictions) not nessesarily answering Questions
• these models know what they dont know this can be mesured via metrics ie if they have 70% confidence there 70% right this shows they are well calabrated (trails off, says false ans)

- instruction tuned (RLHF) models (GPT, Claude etc): are base models that have been fine-tuned to behave helpfully and safely and conversationally, done is two major steps
1) supervised Fine-tuning (SFT): base model is trained on instruction response pairs usally curated or human made ex: "samarize this climate change paper" $\xrightarrow{ans}$ "climat change refers to long term shifts in temp..." this teaches model format and behaviour (to treat prompts as requests not just text to continue" 2) Rienforcement learning with human feed back (RLHF) after SFT mony model outputs are generated for a given prompt and humans rank them (helpful → unhelpfull, truthfull → misleading, polite → rude, etc) then useing these rankings a reward model is trained to score outputs, finally model is fine tuned using Rienforcement learning (exPPO) to maximize reward, the goal is to make model more helpfull, truthfull, harmless
• these models are rewarded for being helpfull, certian, and human like, RLHF penalizes non-ans (ie idont know) and rewards clear & confident answers even if model uncertian, result: these models dont know what they dont know there always confident even if wrong
this causes Hallucinations: fluent, confident wrong answers