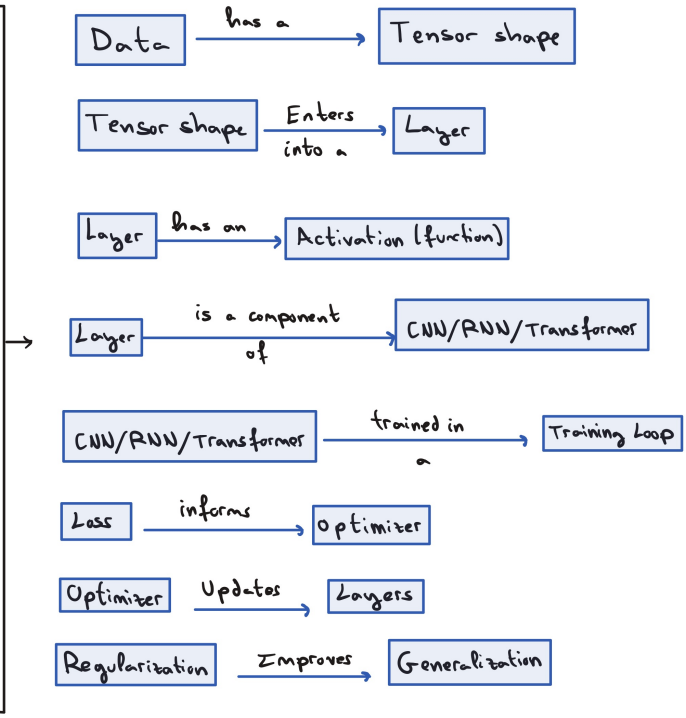
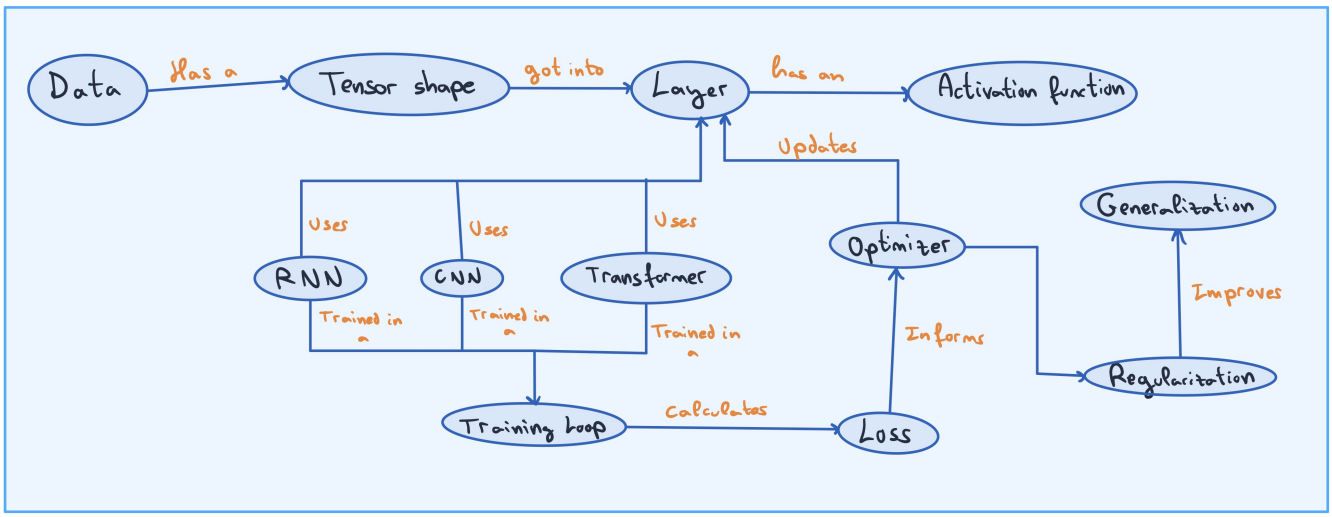
# D1

- **Data**: is the main information (raw).
- **Tensor shape**: the structures of my data (dimensions)
- **Layer**: each layer works over data working on a transformation for the model.
- **Activation**: this is a function inside the layer. The decision filter.
- **Loss**: the error score. It measures how far the results are from the truth.
- **Optimizer**: the optimizer looks the loss and actualize the weighs for a better perform.
- **Training loop**: repetitive process where the model looks the data, makes a guess, and learn from the mistakes
- **Generalization**: this is the final aim. Our model has to perform well not only with our data but with unseen data.
- **Regularization**: techniques to prevent overfitting.
- **CNN**: specialiced in spatial data, like images.
- **RNN**: specialiced in secuencial data, like time.
- **Transformer**: specialiced in all. (Chat GPT)



# D2

| Prompt | URL | Evidence snippet (<= 25 words) | Claim in your own words | Source type & why credible |
|---|---|---|---|---|
| AI vs ML vs DL: define each in 1–2 lines and give one example of each. | https://cloud.google.com/discover/deep-learning-vs-machine-learning | AI: sistemas que imitan la inteligencia humana<br>ML: subcampo de IA donde la máquina aprende de datos sin programar reglas<br>DL: subtipo de ML usando redes neuronales profundas con muchas capas. | El resumen es que la inteligencia artificial engloba al resto. ML y DL son inteligencia artificial, y Dl es un ML aplicando muchas capas de redes neuronales. En el ML se aprende de datos para predecir la etiqueta. | Google Cloud. Son expertos y Google Cloud tiene mucha credibilidad, es mundial y una de las ballenas líderes. |
| What “learning” means in DL: explain the training loop without equations. | https://www.tensorflow.org/guide/basic_training_loops | Learning en DL es el bucle de entrenamiento.<br>1. Forward pass<br>2. Calcular loss<br>3. Backward pass<br>4. Update<br>5. Repetir | Básicamente: learning = bucle. Bucle: datos → predicción → mide error → ajusta pesos → repite hasta mejorar | Documentos oficiales de TensorFlow (líder) |
| Vector/matrix as tensors: explain shape and rank with one example | https://www.geeksforgeeks.org/deep-learning/what-is-tensor-and-tensor-shapes/ | Rank 0: Scalars.<br>Rank 1: Vectors.<br>Rank 2: Matrices.<br>Rank 3: 3D tensors. | Shape representa la dimensión del vector / matriz. Un ejemplo es una matriz 2x3 → shape(2,3)<br>rank = nº de nimensiones (Dimensión 2) | Tutorial de GeeksforGeeks DL, explica los tensores con código. |
| Weights and bias: what do they represent conceptually? | https://www.geeksforgeeks.org/deep-learning/the-role-of-weights-and-bias-in-neural-networks/ | "Weights decide how strongly an input affects output. Bias helps neurons activate even when weighted sum insufficient." | Weights = importancia de cada entrada. Bias = umbral de activación (permite activarse sin entradas) | GeeksforGeeks DL tutorial (explicaciones con código también) |
| Activation: Why do we need non-linearity? (one intuition) | https://stackoverflow.com/questions/9782071/why- | "Without non-linear activation, NN behaves just like single-layer | Muchas capas lineales es como una linea recta. ReLu permite | StackOverflow: expertos en ML |

| | | | | |
|---|---|---|---|---|
| | must-a-nonlinear-activation-function-be-used | perceptron regardless of layers." | curvas complejas. | |
| Loss vs optimizer: what is each one used for? | https://docs.edgeimpulse.com/knowledge/concepts/machine-learning/neural-networks/loss-functions | "Loss: measures difference between prediction and label. Optimizer: adjusts parameters to minimize loss." | Loss mide el error en la predicción (predicho vs real), mientras que optimizer ajusta los pesos para intentar mejorar dicha predicción y reducir el error | Docs Edge Impulse ML, muy profesional y creedibilidad alta. |
| Optimization vs generalization: what is overfitting and how do we detect it? | https://aws.amazon.com/what-is/overfitting/ | "Overfitting: accurate predictions for training data but not for new data." | El modelo se ajusta para los datos disponibles, no para datos nuevos | AWS Machine Learning, muy importante, empresa líder cloud/ML). |
| Course preview: choose ONE (CNN or Transformer) and explain where it is used and what input it expects. | https://www.coursera.org/articles/transformers-vs-convolutional-neural-networks | "Transformers use self-attention to process entire input simultaneously, not local neighborhoods.” | Transformers para texto/visión; input: secuencia tokens, imágen… por ejemplo chat gpt | Coursera, expertos AI |

**He utilizado Perplexity.ai para contrastar la información y como modo de aprendizaje.**