📚 **Resumen de la Clase 1 – Introducción al PLN y Vectorización de Texto**

### **🧩 Contenido Técnico**

#### **1. Introducción al curso y objetivos**

* Se presentó el curso como parte de la especialización en IA, orientado a **entender y construir modelos de lenguaje**.
* Objetivo final: comprender arquitecturas modernas como **Transformers**.
* Se trabajará principalmente con **texto**, una de las modalidades del lenguaje natural.

#### **2. Lenguaje natural: desafíos**

* El lenguaje es:  
  + **Cultural**: cambia según región y época.
  + **Multimodal**: incluye voz, gestos, texto.
  + **Ambiguo**: depende del contexto.
* Se abordó la dificultad de representar computacionalmente algo tan complejo.

#### **3. Línea de tiempo histórica**

* Destacados:  
  + **Test de Turing** (1950): lenguaje como criterio de inteligencia.
  + **John Firth (1957)**: “El significado de una palabra está en la compañía que mantiene”.
  + **Espacios vectoriales** (Salton): analogía entre palabras como vectores.
  + Evolución hacia **Word Embeddings** y **modelos generativos**.

#### **4. Representación de texto: de documentos a vectores**

* Se definieron:  
  + **Corpus**: colección de documentos.
  + **Tokenización**: dividir texto en unidades mínimas (tokens).
  + **Vocabulario**: conjunto de todos los tokens únicos.
* Técnicas de vectorización:  
  + **Frecuencia**: cuenta cuántas veces aparece cada palabra en un documento.
  + **One-Hot Encoding**: binario (aparece/no aparece).
  + Problemas: matrices grandes y esparsas, dominancia de palabras frecuentes.

#### **5. Mejora con TF-IDF**

* Combina:  
  + **TF (Term Frequency)**: frecuencia en el documento.
  + **IDF (Inverse Document Frequency)**: penaliza palabras comunes en todos los documentos.
* Resalta palabras **específicas y relevantes**.
* Normalización mediante **logaritmo** para evitar explosiones numéricas.

#### **6. Similitud de documentos**

* Se introdujo la **similitud coseno** como métrica para comparar textos.  
  + Producto interno entre vectores normalizados.
  + Permite medir **orientación** y no solo magnitud (útil para detectar similitud semántica).
* Ejemplo: comparar dos noticias deportivas con términos similares.

### **📌 Organización del curso**

* Evaluación mediante **desafíos prácticos** (no hay examen final).
* Cada entrega se realiza en un **repositorio personal** (preferentemente en GitHub).
* Se penalizan las entregas tardías: la nota máxima disminuye si se entrega después de ciertas clases.
* Se trabajará con **TensorFlow/Keras** (aunque se permite PyTorch si el estudiante lo prefiere).
* Comunicación vía **Slack**, aunque se considera migrar a **Discord**.