Instituto Tecnológico y de Estudios Superiores de Monterrey

Campus Querétaro

Actividad 4.5: Similitud en textos mediante Cadenas de Markov

Desarrollo de aplicaciones avanzadas de ciencias computacionales (Gpo 301)

Santiago de Querétaro, Querétaro, 24 de mayo del 2024

Aldo Tena García - A01275222

A01275222@tec.mx

Reflexiones

¿Qué preprocesamiento tendrías que hacer con los códigos antes de compararlos? (Por ejemplo, a = b + c y r = m + n se tomarían como completamente distintos). ¿Cómo podrías evitar eso?

Antes de comparar los códigos se podrían tener varias etapas de preprocesamiento para evitar comparaciones no adecuadas o que tomen elementos innecesarios:

1. Eliminar comentarios y espacios en blanco: ya que estos no afectan de ninguna forma la lógica del código, excepto en casos específicos como en lenguajes como Python y su indentación.
2. Normalizar los identificadores: los nombres de variables, funciones y clases pueden ser diferentes, pero tener un funcionamiento idéntico entre ambos códigos.
3. Estructuras de código: se pueden considerar las estructuras de control (como bucles y condicionales) y las llamadas a funciones de manera independiente de su nombre, como se mencionó en el punto 2.
4. Formato de código: verificar que las diferencias en el formato de código no afecten su posterior procesamiento.
5. Utilizar funciones de un parse de código: en vez de solo usar tokens, se podría generar un AST (Abstract Syntax Tree) usando herramientas como javalang.

¿Crees que esta técnica es adecuada para encontrar la similitud entre códigos?

Esta técnica tiene ventajas y desventajas en cuanto a cómo está procesando la información recibida, debido a que captura los tokenes generados por javalang, es capaz de reflejar en cierta parte la estructura y flujo del código, sin embargo, al depender únicamente de los tokens es altamente susceptible a cambios de nombres de identificadores. Se podría cambiar la estructura o el orden del código y no se consideraría como plagio.

¿Es más eficiente que la técnica de similitud por distribuciones de probabilidad?

Cada herramienta tiene un uso diferente dentro del análisis de códigos, si se requiere medir la similitud léxica es mejor la distribución de probabilidad, en cambio, si se quiere medir la similitud en el orden del código es más adecuada usar matrices de transición.