



Generador de Lyrics con Cadenas de Markov

Rodrigo Maranzana

Caso de estudio

- Los generadores de texto usan modelos de procesos estocásticos para generar cadenas de texto.
- Los más avanzados, usados en chatbots, trabajan con el contexto de las palabras.
- Un modelo simple es un generador con Cadenas de Markov. No entiende el contexto, solo puede ver el estado actual y el siguiente.
- **Objetivo: crear un generador de letras de canciones con cadenas de Markov.**

Ejemplo

Harder, Better, Faster, Stronger (2001) Álbum Discovery Daft Punk



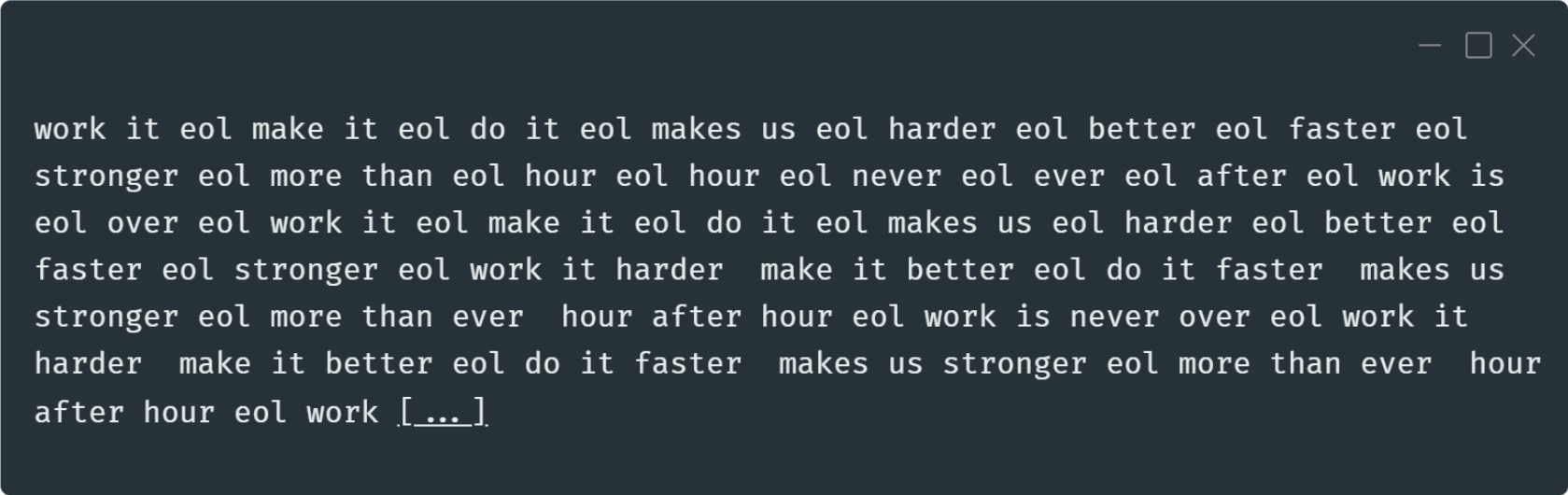
Foto por Daft Punk

Procedimiento

- Limpieza de corpus (string con la letra)
- Conversión a lista de palabras.
- Diccionario de ocurrencias.
- Diccionario de transición.
- Simulador de caminos.
- Visualización de resultado.

Limpieza de corpus

- Reemplazamos caracteres de fin de oración "\n" por "eol" (end of line)
- Eliminamos signos de puntuación.
- Convertimos todo a minúscula.

A dark-themed terminal window with a title bar containing a minus sign, a square icon, and a close 'X' icon. The terminal displays the following text:

```
work it eol make it eol do it eol makes us eol harder eol better eol faster eol  
stronger eol more than eol hour eol hour eol never eol ever eol after eol work is  
eol over eol work it eol make it eol do it eol makes us eol harder eol better eol  
faster eol stronger eol work it harder make it better eol do it faster makes us  
stronger eol more than ever hour after hour eol work is never over eol work it  
harder make it better eol do it faster makes us stronger eol more than ever hour  
after hour eol work [...]
```


Lista de palabras

- Convertimos string en lista de palabras

```
[ 'work' 'it' 'eol' 'make' 'it' 'eol' 'do' 'it' 'eol' 'makes' 'us' 'eol'
  'harder' 'eol' 'better' 'eol' 'faster' 'eol' 'stronger' 'eol' 'more'
  'than' 'eol' 'hour' 'eol' 'hour' 'eol' 'never' 'eol' 'ever' 'eol' 'after'
  'eol' 'work' 'is' 'eol' 'over' 'eol' 'work' 'it' 'eol' 'make' 'it' 'eol'
  'do' 'it' 'eol' 'makes' 'us' 'eol' 'harder' 'eol' 'better' 'eol' 'faster'
  'eol' 'stronger' 'eol' 'work' 'it' 'harder' 'make' 'it' 'better' 'eol'
  'do' 'it' 'faster' 'makes' 'us' 'stronger' 'eol' 'more' 'than' 'ever'
  'hour' 'after' 'hour' 'eol' 'work' 'is' 'never' 'over' 'eol' 'work' 'it'
  'harder' 'make' 'it' 'better' 'eol' 'do' 'it' 'faster' 'makes' 'us'
  'stronger' 'eol' 'more' 'than' 'ever' 'hour' 'after' 'hour' 'eol' 'work'
  'is' 'never' 'over' 'eol' 'work' 'it' 'work' [ ... ] ]
```

Matrices de transición

- El paso natural sería crear matrices de transición para Cadenas de Markov.
- Desventaja: en estos modelos, resultan ser matrices muy dispersas (muchos ceros)
- Es más eficiente crear diccionarios de transición.

Diccionario de ocurrencias

- Guardamos los estados iniciales como claves de un diccionario, y solamente los estados finales que tengan probabilidad mayor a 0.

```
{  
  palabra_1: {sucesor_1: 2, sucesor_2: 5}  
  palabra_2: {sucesor_1: 3}  
  ...  
  palabra_i: { ... }  
}
```

```
>> {  
  'is': {'eol': 1, 'never': 14},  
  'ever': {'eol': 1, 'hour': 14},  
  'makes': {'us': 15},  
  'us': {'eol': 4, 'stronger': 10, 'strong': 1},  
  'faster': {'eol': 3, 'makes': 13},  
  'strong': {'eol': 1},  
  'do': {'it': 16},  
  'it': {  
    'eol': 7, 'harder': 11, 'better': 12,  
    'faster': 14, 'make': 3  
  },  
  'hour': {'eol': 16, 'after': 13},  
  'more': {'than': 15},  
  'make': {'it': 15},  
  'eol': {  
    'make': 2, 'do': 16, 'makes': 2,  
    'harder': 2, 'better': 2, 'faster': 2,  
    'stronger': 2, 'more': 15, 'hour': 2,  
    'never': 1, 'ever': 1, 'after': 1,  
    'work': 30, 'over': 1  
  },  
  'better': {'eol': 14},  
  'over': {'eol': 15},  
  'never': {'eol': 1, 'over': 14},  
  'after': {'eol': 1, 'hour': 13},  
  'work': {'it': 16, 'is': 15},  
  'harder': {'eol': 3, 'make': 10},  
  'than': {'eol': 1, 'ever': 14},  
  'stronger': {'eol': 12}  
}
```


Diccionario de transición

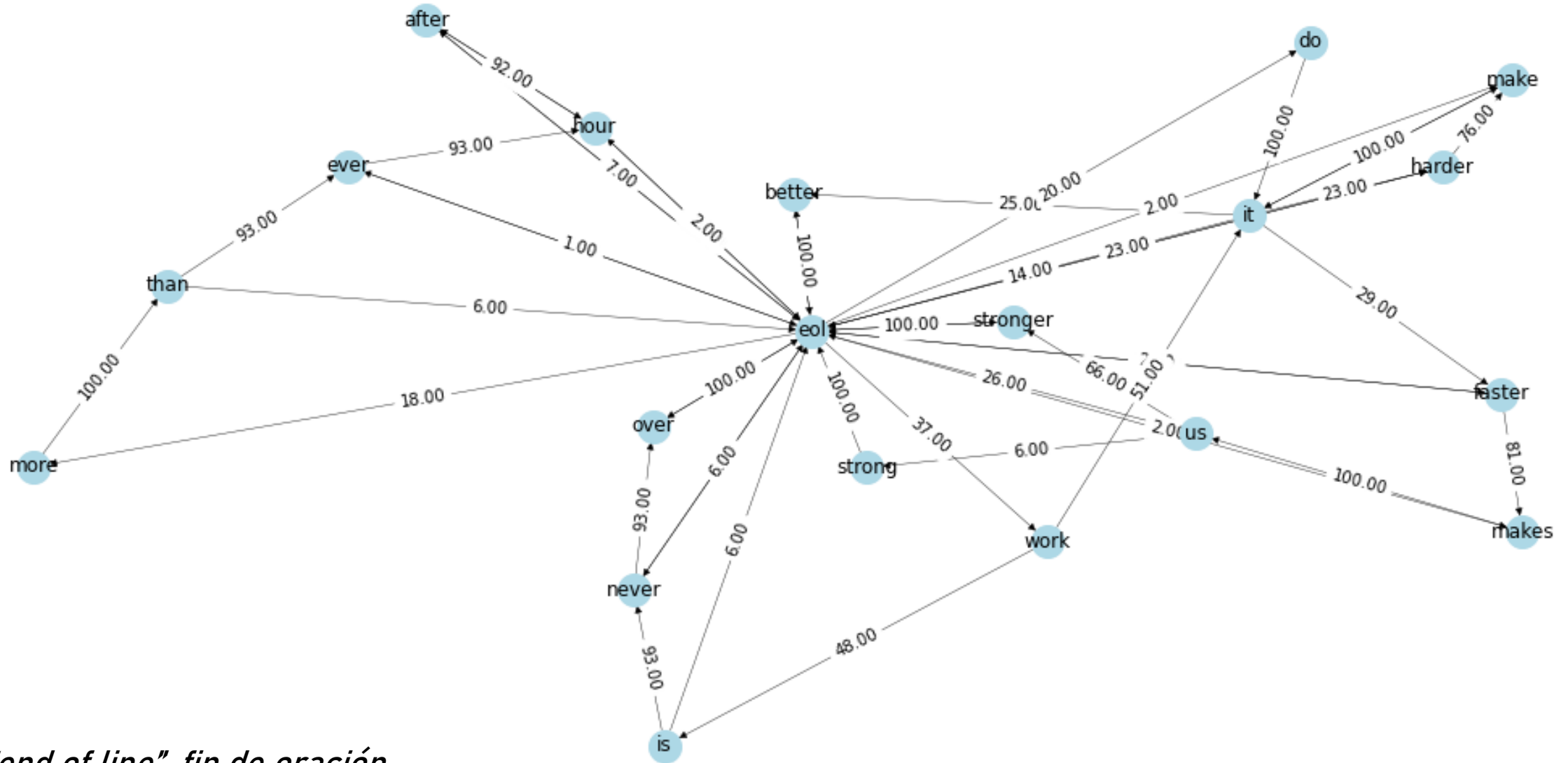
- Convertimos las ocurrencias en probabilidades de transición.
- Recorremos cada clave del diccionario de primer nivel. Luego, dividimos la ocurrencia de cada sucesor por la suma de las ocurrencias.

```
>> {  
  'is': {  
    'eol': 0.06666666666666667,  
    'never': 0.9333333333333333  
  },  
  'ever': {  
    'eol': 0.06666666666666667,  
    'hour': 0.9333333333333333  
  }, 'makes': {'us': 1.0},  
  'us': {  
    'eol': 0.26666666666666666,  
    'stronger': 0.6666666666666666,  
    'strong': 0.06666666666666667  
  },  
  'faster': {  
    'eol': 0.1875,  
    'makes': 0.8125  
  },  
  [ ... ]  
}
```

Grafo de transición

- La relación entre palabras la marca la precedencia.
- Existe una probabilidad de saltar de una palabra a la otra por la cantidad de conexiones entre ellas.
- La relación es “hacia adelante”, es decir, se puede visualizar con un grafo orientado.
 - Los nodos son palabras.
 - Los arcos son probabilidades de transición.

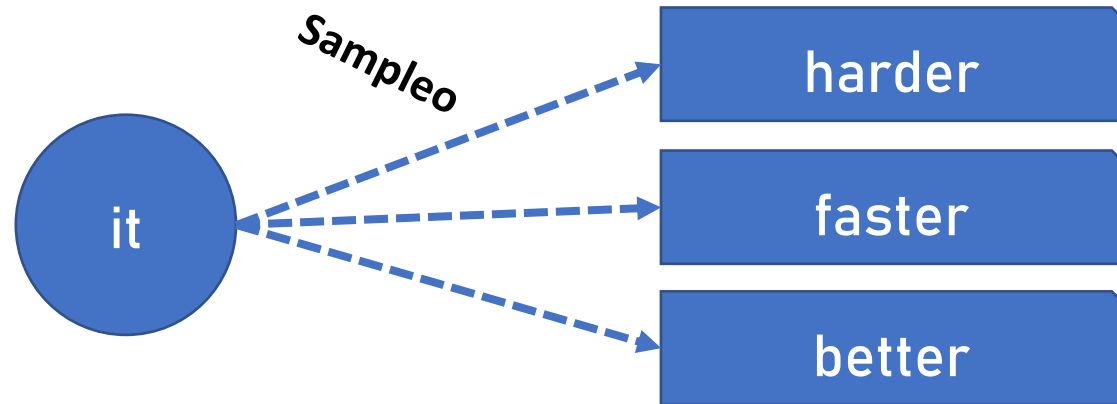
Grafo de transición



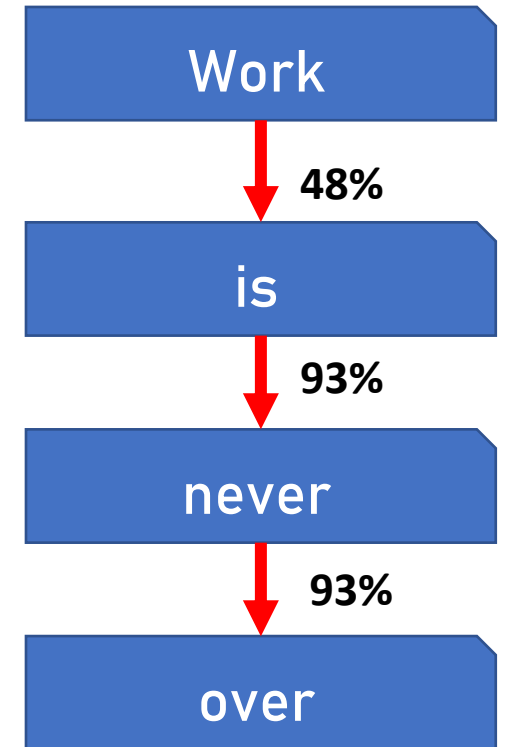
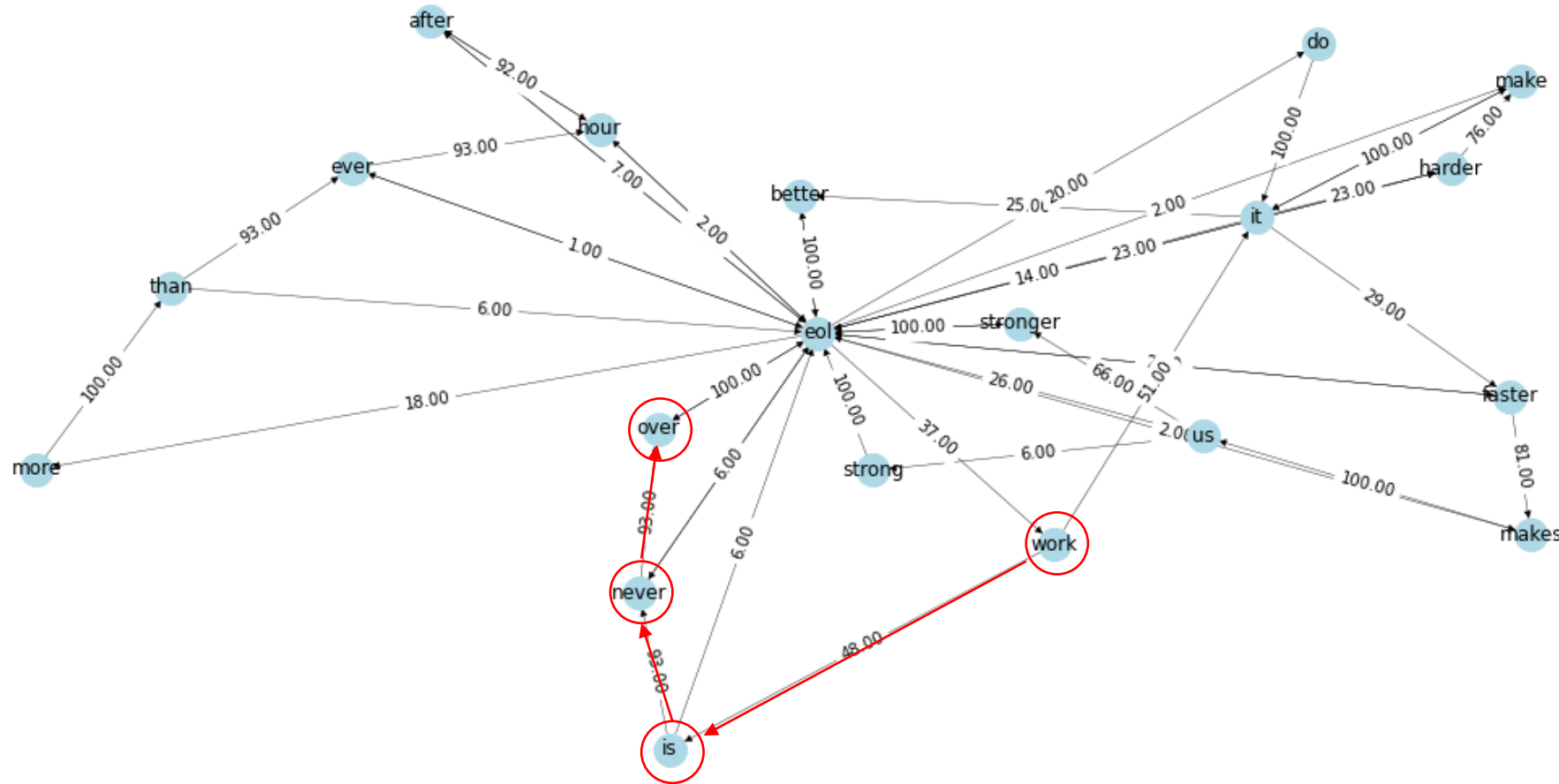
eol: "end of line", fin de oración.

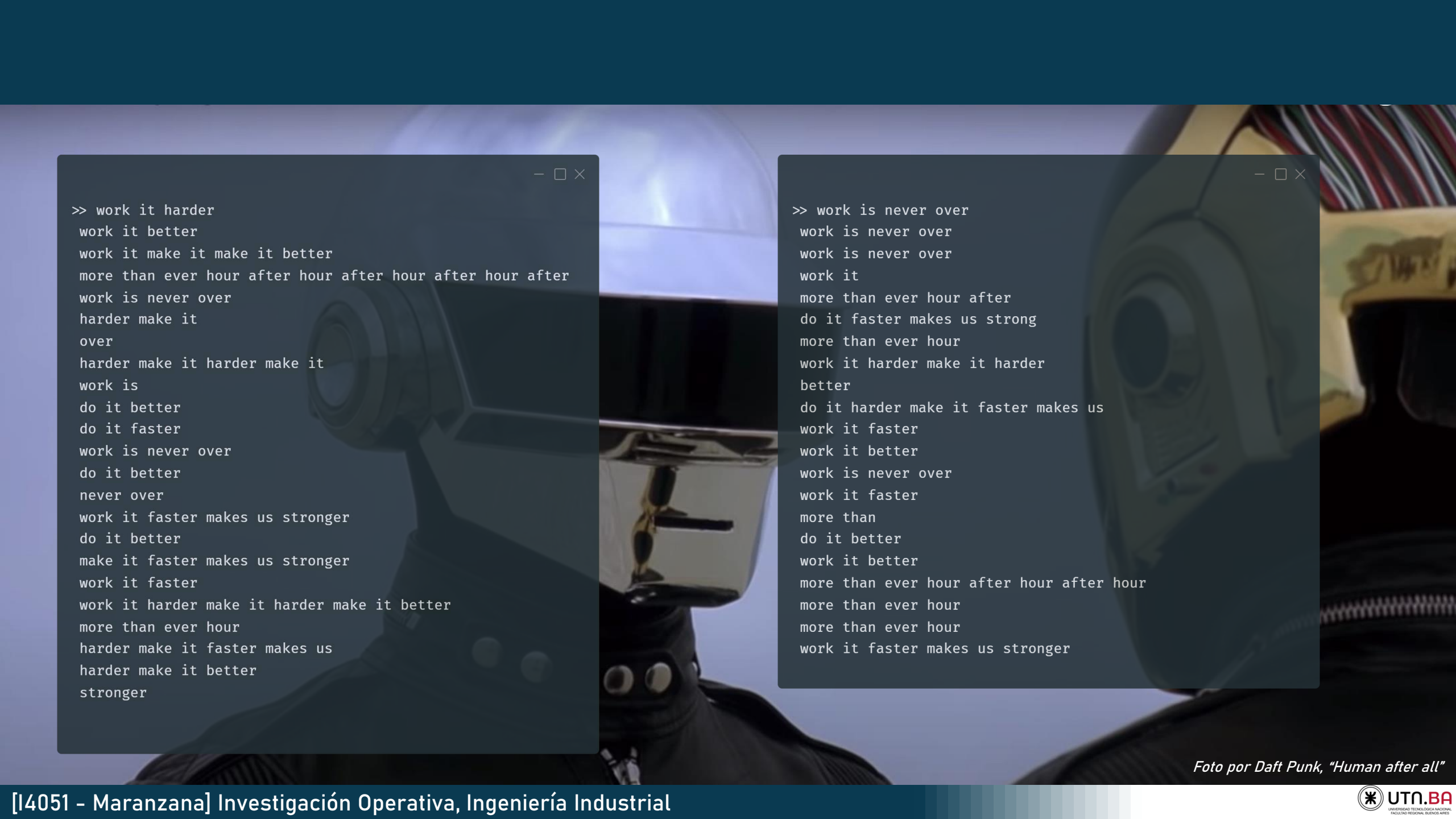
Simulador de caminos

- En un ciclo, recorremos el grafo mediante el diccionario de transición.
- En cada iteración tiramos un dado pesado por la probabilidad de cada estado sucesor.
- Nos dirigimos al siguiente e iteramos.



Ejemplo de secuencia generada





```
>> work it harder
work it better
work it make it make it better
more than ever hour after hour after hour after hour after
work is never over
harder make it
over
harder make it harder make it
work is
do it better
do it faster
work is never over
do it better
never over
work it faster makes us stronger
do it better
make it faster makes us stronger
work it faster
work it harder make it harder make it better
more than ever hour
harder make it faster makes us
harder make it better
stronger
```

```
>> work is never over
work is never over
work is never over
work it
more than ever hour after
do it faster makes us strong
more than ever hour
work it harder make it harder
better
do it harder make it faster makes us
work it faster
work it better
work is never over
work it faster
more than
do it better
work it better
more than ever hour after hour after hour
more than ever hour
more than ever hour
work it faster makes us stronger
```

Foto por Daft Punk, "Human after all"