

# PRÁCTICA 4: RECONOCIMIENTO DE VOZ USANDO POCKETSPHINX.

## DESARROLLO

Comenzamos la práctica por ejecutar *roslaunch bring\_up pocketsphinx\_test.launch*

```
/home/lui/MobileRobots-2020-2-for-Covid19/catkin_ws/src/bring_up/launch/pocketsphinx_t...
File Edit View Search Terminal Help
INFO: fsg_search.c(859): 90 frames, 3148 HMMs (34/fr), 7706 senones (85/fr), 126
0 history entries (14/fr)

INFO: fsg_search.c(869): fsg 0.10 CPU 0.113 xRT
INFO: fsg_search.c(871): fsg 11.66 wall 12.811 xRT
[INFO] [1587509974.360552]: OUTPUT: "MOVE LEFT"
INFO: cmn_live.c(120): Update from < 8.90 -17.21 -0.46 -3.31 -13.53 3.96 1.85
-3.05 -2.85 2.00 -0.46 -0.56 -0.59 >
INFO: cmn_live.c(138): Update to < 8.78 -17.24 -0.49 -3.56 -13.81 3.34 2.34
-3.18 -2.07 1.46 -0.27 -0.46 -0.45 >
INFO: fsg_search.c(859): 146 frames, 5260 HMMs (36/fr), 12786 senones (87/fr), 1
631 history entries (11/fr)

INFO: fsg_search.c(869): fsg 0.11 CPU 0.074 xRT
INFO: fsg_search.c(871): fsg 7.97 wall 5.422 xRT
[INFO] [1587509982.331406]: OUTPUT: "MOVE BACK"
INFO: cmn_live.c(120): Update from < 8.78 -17.24 -0.49 -3.56 -13.81 3.34 2.34
-3.18 -2.07 1.46 -0.27 -0.46 -0.45 >
INFO: cmn_live.c(138): Update to < 15.53 -9.91 0.24 -0.72 -10.67 0.54 -1.02
-2.79 -2.45 1.93 -1.86 -1.47 2.17 >
INFO: fsg_search.c(859): 204 frames, 5578 HMMs (27/fr), 14945 senones (73/fr), 1
769 history entries (8/fr)

INFO: fsg_search.c(869): fsg 0.15 CPU 0.074 xRT
```

En la terminal se mostrará la información anterior, este es pocketsphinx ejecutándose, al principio no era muy comprensible que el programa ya se estaba ejecutando, pero después de unas pruebas se puede observar que el programa está recibiendo por micrófono los comandos que la gramática especifica.

```
grammar voice_cmd;

public <move> = MOVE FORWARD;

public <move2> = MOVE <direction> [<speed>];

<direction> = FORWARD | BACK | LEFT | RIGHT;

<speed> = FULL SPEED | HALF SPEED;
```

Se elaboraron unas cuantas pruebas iniciales utilizando esta gramática y sus distintas combinaciones.

```
/home/lu/MobileRobots-2020-2-for-Covid19/catkin_ws/src/bring_up/launch/pocketsphinx_t...
File Edit View Search Terminal Help
INFO: fsg_search.c(859): 131 frames, 3631 HMMs (27/fr), 9704 senones (74/fr), 94
0 history entries (7/fr)

INFO: fsg_search.c(869): fsg 0.06 CPU 0.047 xRT
INFO: fsg_search.c(871): fsg 1.45 wall 1.100 xRT
[INFO] [1587510042.682181]: OUTPUT: "MOVE LEFT"
INFO: cmn_live.c(88): Update from < 28.01 -1.95 -1.46 1.99 -6.66 0.46 -4.53 -2
.39 0.27 0.01 -3.21 -1.19 0.51 >
INFO: cmn_live.c(105): Update to < 31.96 -0.82 0.07 1.88 -7.34 -1.36 -3.91 -
2.72 0.32 -0.13 -2.66 -1.15 1.18 >
INFO: cmn_live.c(120): Update from < 31.96 -0.82 0.07 1.88 -7.34 -1.36 -3.91 -
2.72 0.32 -0.13 -2.66 -1.15 1.18 >
INFO: cmn_live.c(138): Update to < 30.90 -1.74 -0.25 2.72 -6.97 -1.08 -3.97 -
3.25 0.07 -0.08 -2.49 -1.77 0.79 >
INFO: fsg_search.c(859): 236 frames, 2440 HMMs (10/fr), 7970 senones (33/fr), 70
2 history entries (2/fr)

INFO: fsg_search.c(869): fsg 0.10 CPU 0.041 xRT
INFO: fsg_search.c(871): fsg 4.80 wall 2.026 xRT
[INFO] [1587510047.485628]: OUTPUT: "MOVE FORWARD FULL SPEED"
INFO: cmn_live.c(88): Update from < 30.90 -1.74 -0.25 2.72 -6.97 -1.08 -3.97 -3
.25 0.07 -0.08 -2.49 -1.77 0.79 >
INFO: cmn_live.c(105): Update to < 33.75 -0.53 2.07 2.75 -8.54 -2.56 -3.03 -
3.56 -0.01 0.23 -2.53 -2.01 0.46 >
```

Algunas veces se pueden encontrar mensajes como el siguiente donde pocketsphinx no reconoce la oración y arroja un error.

```
INFO: fsg_search.c(869): fsg 0.06 CPU 0.046 xRT
INFO: fsg_search.c(871): fsg 2.41 wall 2.023 xRT
ERROR: "fsg_search.c", line 940: Final result does not match the grammar in fram
e 118
INFO: cmn_live.c(88): Update from < 21.50 -7.50 0.58 0.63 -10.87 2.00 -1.65 -
3.38 -3.43 2.18 0.96 -2.03 1.28 >
INFO: cmn_live.c(105): Update to < 23.65 -6.66 0.34 1.45 -10.36 2.29 -1.92
-3.29 -3.37 1.82 2.21 -2.12 0.55 >
INFO: cmn_live.c(120): Update from < 23.65 -6.66 0.34 1.45 -10.36 2.29 -1.92
-3.29 -3.37 1.82 2.21 -2.12 0.55 >
INFO: cmn_live.c(138): Update to < 23.00 -6.86 0.67 0.70 -10.15 2.29 -2.22
-3.27 -3.39 1.62 2.13 -1.95 0.32 >
INFO: fsg_search.c(859): 141 frames, 2381 HMMs (16/fr), 7088 senones (50/fr), 79
0 history entries (5/fr)
```

El error simplemente hace referencia que el resultado (la entrada por micrófono) no se encontró en la gramática propuesta dentro del archivo voice\_cmd.gram. Como nota para que el programa detecte mejor las oraciones un acento más claro como el estadounidense ayuda, cuando se utilizó una entonación británica parecía no identificar algunas palabras correctamente. Esto está influenciado por el archivo .dic que contiene la fonética de las oraciones empleadas.

Ahora procedemos a editar el archivo *pocketsphinx\_test.launch* para ahora utilizar una gramática relacionada a un restaurante.

```
#JSGF V1.0;
grammar restaurant;
public <command> = (<justina> | <wantCombo> | <wantBeverage>);
<justina> = JUSTINA (YES | NO | WAIT | START | TAKE A ORDER | THIS IS THE ORDER);
<wantCombo> = I WANT [A] (PRINGLES | CHIPS | PASTA | COOKIES | PASTA | NOODLES | TUNA $
<wantBeverage> = I WANT A (TEA | BEER | COKE | WATER);
```

Analizando la gramática que se tiene para el restaurante podemos observar que un comando puede tener tres formas y cada una de estas tiene una función diferente.

1. <Justina>: Este segmento indicará el comando que Justina debe realizar.
2. <wantCombo>: Indica el combo que se va a llevar, está formado de I want, que es necesario, [A] que puede ser o no dicho y de dos partículas necesarias con múltiples opciones separadas por un And obligatorio.
3. <wantBeverage>: Formado de I want a, que es necesario y de 4 posibles opciones de bebidas.

Algunos ejemplos de comandos que se pueden formar tomando en cuenta la gramática provista son:

- Justina this is the order
- Justina yes
- I want pringles and cookies
- I want chips and pasta
- I want a beer
- I want tea

A continuación se añadirán capturas de algunas frases reconocidas por pocketsphinx.

```
INFO: fsg_search.c(869): fsg 0.22 CPU 0.048 xRT
INFO: fsg_search.c(871): fsg 6.69 wall 1.494 xRT
[INFO] [1587510987.713041]: OUTPUT: "I WANT PRINGLES AND COOKIES"
INFO: cmn_live.c(120): Update from < 28.34 -10.12 1.63 6.25 -4.97 -0.93 -6.16
-1.76 2.74 1.87 1.54 -2.39 2.36 >
INFO: cmn_live.c(138): Update to < 26.83 -10.23 1.06 6.65 -5.18 -1.05 -7.30
-1.82 2.57 1.57 1.51 -2.16 2.57 >
INFO: fsg_search.c(859): 70 frames, 1226 HMMs (17/fr), 3214 senones (45/fr), 305
history entries (4/fr)
```



```

[INFO] [1587510991.369889]: OUTPUT: "I WANT A BEER"
INFO: cmn_live.c(88): Update from < 27.93 -8.66 2.64 5.74 -7.50 -0.58 -5.68 -2
.75 1.31 2.11 3.30 -0.99 2.37 >
INFO: cmn_live.c(105): Update to < 29.53 -9.14 2.71 7.92 -7.02 -0.73 -6.89 -
1.86 1.50 1.68 2.53 -1.14 2.44 >
INFO: cmn_live.c(120): Update from < 29.53 -9.14 2.71 7.92 -7.02 -0.73 -6.89 -
1.86 1.50 1.68 2.53 -1.14 2.44 >
INFO: cmn_live.c(138): Update to < 29.70 -7.28 1.34 6.88 -5.81 -1.24 -6.81 -
4.10 0.37 1.85 1.97 -0.64 3.08 >
INFO: fsg_search.c(859): 189 frames, 1714 HMMs (9/fr), 6665 senones (35/fr), 434
history entries (2/fr)

```

```

[INFO] [1587511001.285916]: OUTPUT: "JUSTINA THIS IS THE ORDER"
INFO: cmn_live.c(88): Update from < 29.82 -8.36 1.21 6.73 -5.30 -1.25 -7.52 -4
.42 1.42 2.65 1.20 -0.97 3.44 >
INFO: cmn_live.c(105): Update to < 30.16 -9.29 1.38 7.64 -5.38 -1.19 -7.31 -
4.44 1.88 2.49 1.32 -0.83 3.65 >
INFO: cmn_live.c(120): Update from < 30.16 -9.29 1.38 7.64 -5.38 -1.19 -7.31 -
4.44 1.88 2.49 1.32 -0.83 3.65 >
INFO: cmn_live.c(138): Update to < 29.65 -8.93 0.01 5.48 -4.91 -1.28 -6.82 -
4.64 1.16 3.31 1.26 -0.87 3.64 >
INFO: fsg_search.c(859): 187 frames, 1919 HMMs (10/fr), 7400 senones (39/fr), 44
1 history entries (2/fr)

```

```

[INFO] [1587511001.285916]: OUTPUT: "JUSTINA THIS IS THE ORDER"
INFO: cmn_live.c(88): Update from < 29.82 -8.36 1.21 6.73 -5.30 -1.25 -7.52 -4
.42 1.42 2.65 1.20 -0.97 3.44 >
INFO: cmn_live.c(105): Update to < 30.16 -9.29 1.38 7.64 -5.38 -1.19 -7.31 -
4.44 1.88 2.49 1.32 -0.83 3.65 >
INFO: cmn_live.c(120): Update from < 30.16 -9.29 1.38 7.64 -5.38 -1.19 -7.31 -
4.44 1.88 2.49 1.32 -0.83 3.65 >
INFO: cmn_live.c(138): Update to < 29.65 -8.93 0.01 5.48 -4.91 -1.28 -6.82 -
4.64 1.16 3.31 1.26 -0.87 3.64 >
INFO: fsg_search.c(859): 187 frames, 1919 HMMs (10/fr), 7400 senones (39/fr), 44
1 history entries (2/fr)

```

```

INFO: fsg_search.c(869): fsg 0.11 CPU 0.069 xRT
INFO: fsg_search.c(871): fsg 10.96 wall 6.763 xRT
[INFO] [1587511024.964376]: OUTPUT: "JUSTINA WAIT"
INFO: cmn_live.c(120): Update from < 30.02 -10.89 2.16 8.96 -7.09 -2.10 -6.46
-4.24 1.92 1.66 0.75 -1.40 3.22 >
INFO: cmn_live.c(138): Update to < 29.12 -10.59 1.95 9.59 -6.74 -2.38 -6.45
-4.25 1.97 1.89 0.06 -1.46 3.57 >
INFO: fsg_search.c(859): 85 frames, 3845 HMMs (45/fr), 8886 senones (104/fr), 98
5 history entries (11/fr)

```

## **Cambios a parámetros de pocketsphinx\_text.launch.**

Los cambios realizados al archivo launch fueron cuatro:

- Cambiar el valor del parámetro gram a restaurant.
  - El archivo .gram contiene la gramática del comando, es decir, las partes que lo conforman y las distintas posibilidades que cada parte puede ser. Si se tuviera que hacer una analogía este archivo es muy similar al parser o análisis sintáctico de un compilador, donde se analiza una cadena de símbolos con base a una serie de reglas gramaticales.
- Cambiar el valor del parámetro dict a restaurant.dict
  - El archivo .dict contiene la fonética de cada una de las palabras que serán reconocidas y están especificadas dentro del archivo de gramática. Esto le da la posibilidad al programa de ser adaptado a otros lenguajes utilizando la fonética del lenguaje objetivo, también de adaptar a pronunciaciones particulares.
- Cambiar el valor del parámetro grammar a restaurant.
  - Cambiamos la especificación de gramática para que corresponda a la de restaurant.
- Cambiar el valor del parámetro rule a command.
  - En el archivo anterior la gramática conformaba a una oración de nombre “move2” en el caso de la gramática de restaurante la oración tiene el nombre “command”. Esta parte se explicó anteriormente dos páginas atrás cuando se habló de la gramática de restaurant.

## **OBSERVACIONES**

Me resultó bastante interesante como el conocimiento adquirido en mi materia de compiladores me ayudó a comprender como funciona el procesamiento de voz en el programa. Esto tiene sentido ya que un compilador prácticamente toma un lenguaje y lo vuelve algo comprensible para la computadora, que es lo mismo que estamos haciendo con pocketsphinx, sólo que en vez de tomar un lenguaje de computadora estamos tomando lenguaje natural o humano y volviéndolo algo comprensible para esta última.