REVISTA Ingeniería

Research

Acoustic and Language Modeling for Speech Recognition of a Spanish Dialect from the Cucuta Colombian Region

Modelo Acústico y de Lenguaje del Idioma Español para el dialecto Cucuteño, Orientado al Reconocimiento Automático del Habla

Juan Celis Nuñez, Rodrigo Llanos Castro, Byron Medina Delgado, Sergio Sepúlveda Mora, Sergio Castro Casadiego

Departamento de Electricidad y Electrónica. Universidad Francisco de Paula Santander. Colombia.

Email: sergio.castroc@ufps.edu.co

Recibido: 01/03/2017 Modificado: 08/08/2017 *Aceptado*: 22/08/2017

Abstract

Context: Automatic speech recognition requires the development of language and acoustic models for different existing dialects. The purpose of this research is the training of an acoustic model, a statistical language model and a grammar language model for the Spanish language, specifically for the dialect of the city of San Jose de Cucuta, Colombia, that can be used in a command control system. Existing models for the Spanish language have problems in the recognition of the fundamental frequency and the spectral content, the accent, pronunciation, tone or simply the language model for Cucuta's dialect.

Method: in this project, we used Raspberry Pi B+ embedded system with Raspbian operating system which is a Linux distribution and two open source software, namely CMU-Cambridge Statistical Language Modeling Toolkit from the University of Cambridge and CMU Sphinx from Carnegie Mellon University; these software are based on Hidden Markov Models for the calculation of voice parameters. Besides, we used 1913 recorded audios with the voice of people from San Jose de Cucuta and Norte de Santander department. These audios were used for training and testing the automatic speech recognition system.

Results: we obtained a language model that consists of two files, one is the statistical language model (.lm), and the other is the jsgf grammar model (.jsgf). Regarding the acoustic component, two models were trained, one of them with an improved version which had a 100 % accuracy rate in the training results and 83 % accuracy rate in the audio tests for command recognition. Finally, we elaborated a manual for the creation of acoustic and language models with CMU Sphinx software.

Conclusions: The number of participants in the training process of the language and acoustic models has a significant influence on the quality of the voice processing of the recognizer. The use of a large dictionary for the training process and a short dictionary with the command words for the implementation is important to get a better response of the automatic speech recognition system. Considering the accuracy rate above 80 % in the voice recognition tests, the proposed models are suitable for applications oriented to the assistance of visual or motion impairment people.

Keywords: Speech Recognition, acoustic models, language models, CMU Sphinx, Raspberry Pi.

Resumen

Contexto: el reconocimiento automático del habla requiere el desarrollo de modelos de lenguaje y modelos acústicos para los diferentes dialectos que existen. El objeto de esta investigación es el entrenamiento de un modelo acústico, un modelo de lenguaje estadístico y un modelo de lenguaje gramatical para el idioma español, específicamente para el dialecto de la ciudad de San José de Cúcuta, Colombia, que pueda ser utilizado en un sistema de control por comandos. Lo anterior motivado por las deficiencias que presentan los modelos existentes para el idioma español, en el reconocimiento de la frecuencia fundamental y contenido espectral, el acento, la pronunciación, el tono o simplemente al modelo de lenguaje de la variante dialéctica de esta región.

Método: este proyecto utiliza el sistema embebido Raspberry Pi B+ con el sistema operativo Raspbian que es una distribución de Linux y los softwares de código abierto CMU-Cambridge Statistical Language Modeling toolkit de la Universidad de Cambridge y CMU Sphinx de la Universidad Carnegie Mellon; los cuales se basan en los modelos ocultos de Markov para el cálculo de los parámetros de voz. Además, se utilizaron 1913 audios grabados por locutores de la ciudad de San José de Cúcuta y el departamento de Norte de Santander para el entrenamiento y las pruebas del sistema de reconocimiento automático del habla.

Resultados: se obtuvo un modelo de lenguaje que consiste de dos archivos, uno de modelo de lenguaje estadístico (. lm), y uno de modelo gramatical (. jsgf). En relación con la parte acústica se entrenaron dos modelos, uno de ellos con una versión mejorada que obtuvo una tasa de acierto en el reconocimiento de comandos del 100% en los datos de entrenamiento y de 83% en las pruebas de audio. Por último, se elaboró un manual para la creación de los modelos acústicos y de lenguaje con el software CMU Sphinx.

Conclusiones: el número de participantes en el proceso de entrenamiento de los modelos acústicos y de lenguaje influye significativamente en la calidad del procesamiento de voz del reconocedor. A fin de obtener una mejor respuesta del sistema de Reconocimiento Automático del Habla es importante usar un diccionario largo para la etapa de entrenamiento y un diccionario corto con las palabras de comando para la implementación del sistema. Teniendo en cuenta que en las pruebas de reconocimiento se obtuvo una tasa de éxito mayor al 80% es posible usar los modelos creados en el desarrollo de un sistema de Reconocimiento Automático del Habla para una aplicación orientada a la asistencia de personas con discapacidad visual o incapacidad de movimiento

Palabras clave: Reconocimiento del habla, modelos acústicos, modelos de lenguajes, CMU Sphinx, Raspberry Pi.

Cite this work as: J. D. Celis, R. A. Llanos, B. Medina, S. B. Sepúlveda, S. A. Castro, " Acoustic and Language Modeling for Speech Recognition of a Spanish Dialect from the Cucuta Colombian Region", Ingeniería, vol. 22, no. 3, pp. XXX-XXX, 2017.

DOI: http://doi.org/10.14483/udistrital.jour.reving.2017.3.a04

1. Introducción

El reconocimiento automático del habla es un avance tecnológico de creciente interés debido a la facilidad que tiene el ser humano de expresarse mediante el uso de sus pliegues vocales. Aunque esta tecnología lleva tiempo desarrollándose, se presenta en la actualidad como una posible solución en el diseño y desarrollo de una interfaz de acceso por habla que permita el control de un proceso sin necesidad de dedicar tiempo a la supervisión de dicha tarea [1][2]; además que debe ser eficiente y no menos importante, debe ser amigable con el usuario para que sea ejecutado por todo tipo de personas.

La eficiencia de este tipo de sistema se puede evaluar en el porcentaje de éxito del reconocimiento de palabras u oraciones según sea el tipo de reconocedor implementado, pero este índice o criterio depende esencialmente de un diccionario fonético y de dos modelos; por una parte, el modelo acústico que representa la distribución de probabilidad de los fonemas en la señal de audio; por otra parte, el modelo del lenguaje, el cual representa la distribución de probabilidad de una secuencia de palabras. Estos dos modelos en conjunto permiten decodificar la información de las señales de audio del usuario a partir de la relación de las dos probabilidades mencionadas anteriormente.

En la actualidad existen en internet alrededor de quince modelos acústicos y de lenguaje desarrollados para diferentes idiomas o un solo idioma [3] [4], como lo son por ejemplo el inglés (americano y británico), el chino, el ruso y el alemán, así como para sistemas multilenguaje [5] [6]. Aunque para el español también existen modelos libres [7] en organizaciones como VoxForge o SourceForge, la efectividad para aplicaciones basadas en sistemas de control por comandos puede ser baja debido a la variedad de acentos que tiene tal idioma y la velocidad del habla, el modelo de lenguaje utilizado y la característica propia del funcionamiento del reconocedor. Por tal motivo, surge la necesidad de entrenar o desarrollar los modelos acústicos del lenguaje para el español colombiano, lo que indirectamente puede impulsar a la investigación del reconocimiento de dialectos regionales en este país.

En esta investigación se realizó la creación de un modelo de lenguaje y un modelo acústico para el idioma español, específicamente para el dialecto cucuteño, utilizando como herramienta el toolkit CMU Sphinx [8] y el software CMU-Cambridge Statistical Language Modeling toolkit (CMUCLMTK) [9], además de audios grabados por habitantes del territorio, obteniendo modelos nativos que sirven en primer lugar para el desarrollo de sistemas de RAH realizados en la región, que permiten realizar comparaciones con el modelo de español desarrollado por SourceForge en este mismo toolkit. De igual manera se pretende exponer de una manera clara una metodología para la creación de modelos de lenguaje y acústicos para sistemas de RAH.

Teniendo en cuenta las características de un sistema RAH, se clasificó al sistema de reconocimiento desarrollado como un sistema de múltiples usuarios, para palabra aislada (específicamente un comando) y de vocabulario pequeño, ya que solo reconocerá nueve comandos compuestos cada uno por tres palabras. Los modelos entrenados servirán como base para una aplicación de "control por comandos" [10] [11] mediante archivos de audio, por lo cual se emplea una investigación científica con un tipo de estudio experimental para obtener valores cuantitativos que sirvan para comparar la eficiencia y la precisión de los resultados obtenidos.

2. Materiales y métodos

2.1. Reconocimiento Automático del Habla

El Reconocimiento Automático del Habla (RAH, por sus siglas en español; ASR por sus siglas en inglés) es el proceso mediante el cual los sonidos provenientes del aparato productor del habla (fonación y articulación) son transcritos a un conjunto de símbolos ortográficos compatibles con las reglas gramaticales de la lengua objetivo de que se trate, tal y como se representa en la Figura 1.

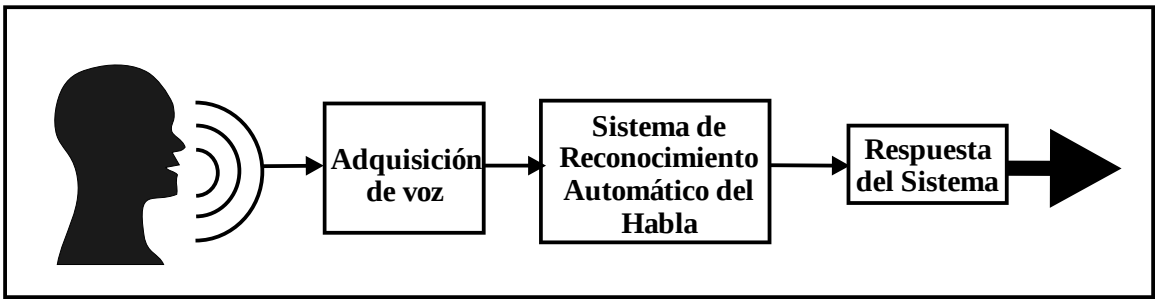

Figura 1. Sistema RAH

En el desarrollo de la inteligencia computacional y artificial se denomina RAH a la capacidad del sistema de funcionar mediante una interfaz de acceso por habla [12]. Dicha capacidad tiene como finalidad transformar la señal acústica del habla producida por el ser humano en información que permita a la interfaz crear un diálogo con el usuario, de tal manera que esta sintetice una repuesta o ejecute un proceso con base en la entrada del sistema [13].

Los procesos de Reconocimiento Automático del Habla están basados en Redes Bayesianas, específicamente en los Modelos Ocultos de Markov (HMM) [14]. Desde el punto de vista matemático el RAH se puede expresar como un problema estadístico en el cual se desea conocer la palabra interpretada de la señal acústica, teniendo como parámetros la palabra predecesora reconocida y los datos obtenidos de la señal acústica; estadística que se puede resolver mediante los HMM.

El software CMU Sphinx es un toolkit de librerías y programas de código abierto desarrollado en Java y realizado bajo la licencia BSD, creado por la universidad Carnegie Mellon para el desarrollo de sistemas de reconocimiento automático del habla. Este cuenta con la familia de programas Sphinx (actualmente versión 4) el cual es un reconocedor del habla de alto nivel, así como PocketSphinx, que es un reconocedor del habla para sistemas embebidos; además del SphinxTrain que permite realizar el entrenamiento de un modelo acústico nuevo [15] o adaptar uno ya existente [16] y el software de código abierto CMUCLMTK de la Universidad de Cambridge el cual permite la creación de los diferentes modelos del lenguaje.

El toolkit para el reconocimiento del habla CMU Sphinx puede ser descargado de su web oficial; es importante verificar que al descargar los paquetes de instalación del software que compone el toolkit, todos sean la misma versión (actualmente 5PreAlpha).

Se usó como base el sistema embebido Raspberry Pi B+ con la distribución Raspbian de Linux, en donde se instala el Toolkit de Reconocimiento del habla CMU Sphinx. Inicialmente, se opera con el CMUCLMTK para generar los modelos de lenguaje, posteriormente con el SphinxTrain se ejecuta la tarea de entrenamiento del modelo acústico y finalmente con el SphinxBase y el PocketSphinx se comprueba el funcionamiento de los modelos obtenidos usando el micrófono de un smartphone y una aplicación móvil como entrada de audio.

2.1.1. Características de los sistemas del RAH

Un sistema de RAH tiene varias características que sirven de igual manera para clasificar la forma en que funcionan y trabajan la información que entra al sistema, como parte de la interfaz de acceso por habla. En la Tabla 1 se describen las características de un sistema RAH.

Tabla 1. Características de los sistemas de RAH

Característica	Descripción
Usuario o hablante	Un sistema de RAH puede ser para un solo usuario o para usuarios múltiples. Un reconocedor de usuarios múltiples es menos preciso, pero permite un mayor alcance en la población que puede hacer uso del reconocedor, ya que no tiene la limitación de detectar el habla de un solo usuario.
Estilo del habla	El RAH puede funcionar de dos formas, uno de palabra aislada y otro de habla continua. Un RAH de habla continua hace que el sistema opere con mayor complejidad, ya que habrá mayor dificultad en determinar los tiempos

	de silencio que marcan los comienzos y finales de los segmentos hablados, existentes en el diálogo.
Tamaño del vocabulario	El vocabulario repercute en la velocidad de operación del reconocedor. Cuanto más grande sea el diccionario, más demorará en encontrar la palabra asignada a la probabilidad generada por el programa. Además, se pueden presentar casos en donde el sistema detecte coincidencias y dé como resultado una palabra equivocada. Las palabras deben estar tanto en el **diccionario fonético,** como en el modelo de lenguaje si se desea reconocer.
Condiciones de operación y ruido	El sistema funcionará dependiendo de las condiciones sonoras en las que opere, especialmente el ruido del entorno sobre el cual funciona. Estos ruidos pueden ocasionar que se decodifique la señal de una manera **incorrecta** obteniendo un resultado diferente al esperado.

2.1.2. Componentes del RAH

Los sistemas del Reconocimiento Automático del Habla se componen esencialmente de tres partes, un diccionario fonético, un modelo del lenguaje y un modelo acústico.

El diccionario fonético es el vocabulario de palabras junto con la división fonética de cada una de ellas, con el cual va a operar el sistema del RAH. Se usa en conjunto con los modelos del lenguaje y acústico para decodificar las señales de audio del usuario, el diccionario fonético utilizado en la creación de los modelos de lenguaje y acústico es el alojado en el repositorio de SourceForge.

El modelo del lenguaje representa la distribución de probabilidad de la palabra y el orden de la misma en el sistema del RAH. Este modelo se encarga de la búsqueda de la palabra y le permite al sistema conocer la palabra siguiente dependiendo de la palabra anterior, es decir da a conocer la posición y probabilidad que tiene una palabra determinada en una oración o comando. Para esta investigación en específico se crearon dos modelos de lenguaje, siendo uno de ellos el modelo de lenguaje gramatical el cual representa la distribución de probabilidad de una secuencia de palabras a partir de una serie de reglas gramaticales, donde se especifican los comandos con el cual funciona el sistema de RAH desarrollado. El modelo acústico representa la distribución de probabilidad de los fonemas en la señal de audio.

2.2. Desarrollo del reconocedor automático del habla

Luego de realizar una breve descripción de estos sistemas de procesamiento de voz, es importante entender la manera como se desarrollan y entrenan los diferentes elementos que componen el reconocedor. En este punto es importante aclarar que el RAH está desarrollado bajo el software de código abierto CMU Sphinx [17] el cual es instalado y ejecutado en el ordenador de placa reducida Raspberry PI B+, de sistema operativo Raspbian [18]. Como se mencionó anteriormente los modelos del lenguaje y acústico se desarrollan para satisfacer las necesidades de un RAH de múltiples hablantes y de un sistema de control por comandos. El proceso de creación de estos sistemas se puede apreciar en la Figura 2.

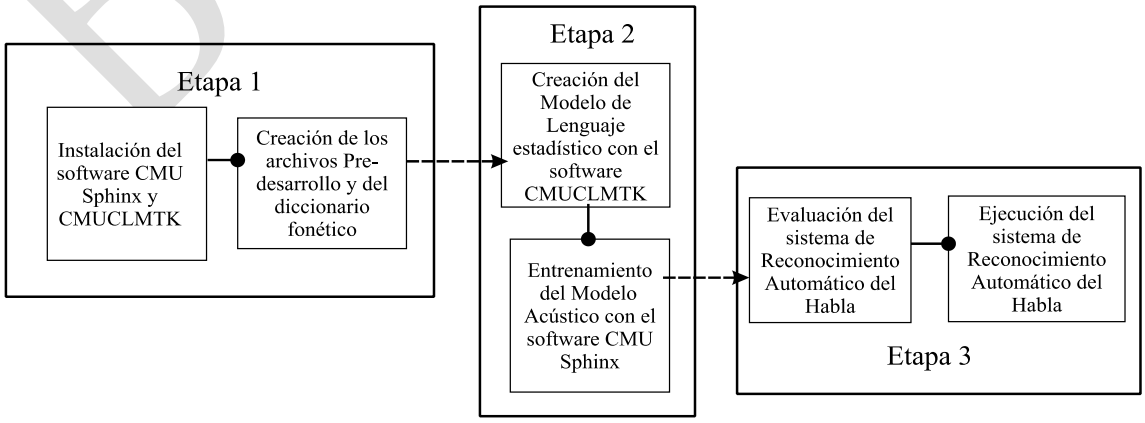

Figura 2. Proceso de desarrollo de un sistema de Reconocimiento Automático del Habla

2.2.1. Preparación pre-desarrollo

Antes de desarrollar el modelo del lenguaje y el entrenamiento del modelo acústico, es necesario preparar los siguientes elementos:

- Un diccionario fonético, este puede ser descargado de SourceForge que de manera gratuita prestan el servicio de recolección y preparación de archivos necesarios para el desarrollo de un sistema de RAH; para esta investigación se utilizaron dos diccionarios fonéticos: el primero descargado de la web SourceForge de 88000 palabras aproximadamente fue utilizado para el entrenamiento del modelo acústico, debido a la gran cantidad de palabras que este posee permite realizar un entrenamiento con un rango amplio de futuros comandos; y el segundo propio del proyecto, el cual contiene solo las trece palabras que forman los comandos, lo que permite limitar la ejecución del reconocedor.

- Un archivo de texto plano con las frases que se emplearán como comandos del sistema y se usará para la creación del modelo del lenguaje.

- Un archivo de texto plano con los fonemas utilizados y que componen cada una de las palabras encontradas en el vocabulario (.Phone), uno con las etiquetas de silencio que el sistema relacionará con los limites de los segmentos hablados de las grabaciones de audio (.Filler).

- Los archivos de texto plano de transcripción y de entrenamiento que contendrán las frases grabadas en los archivos de audio por los usuarios para el entrenamiento y prueba del modelo acústico.

- Los archivos de audio grabados por los usuarios necesarios para el entrenamiento del modelo acústico. Las duraciones de estos archivos deben sumar en total una hora de grabación como mínimo para que el sistema pueda realizar el entrenamiento del modelo acústico.

Es importante resaltar que los archivos de texto plano de transcripción y el archivo de texto plano que contiene las frases necesarias para la creación del modelo del lenguaje, deben estar escritos en formato Unicode UTF-8, además deben comenzar con la etiqueta <s> y finalizar con la etiqueta </s>, las cuales deben estar incluidas como representación del silencio en el archivo de extensión .Filler.

2.2.2. Creación del modelo del lenguaje

El modelo del lenguaje es el elemento del RAH que define las probabilidades relacionadas con la aparición y el orden de las palabras en una oración. Permite junto con las características y datos obtenidos de las señales acústicas en el entrenamiento del modelo acústico, decodificar la información de las señales de habla del usuario.

En el toolkit CMU Sphinx el modelo del lenguaje se puede crear de tres tipos. El primer tipo es lista de palabras claves, el cual se puede realizar a partir de un archivo de audio de gran tamaño y mediante el PocketSphinx se localiza la palabra indicada, dicho proceso genera un archivo cuyo contenido especifica los umbrales de la palabra en el audio. El segundo tipo es gramática, este elemento es de tipo texto y su extensión es .jsgf o .gram; tal archivo puede ser escrito de manera manual y en formato jsgf; son archivos realizados para representar de manera precisa el orden en el cual deben aparecer las palabras y son utilizados principalmente para sistemas de reconocimiento en comando y control.

El tercer tipo es el Modelo del Lenguaje Estadístico (LM), que se crea mediante el software CMUCMLTK. A partir de un archivo de texto plano en el cual se encuentran todas las frases o palabras que puede reconocer el RAH, este software permite crear una serie de archivos, los cuales son:

- Archivo frecuencia de palabra (.wfreq): este archivo, como su nombre lo indica contiene por palabra el número de veces que esta aparece dentro del texto u oraciones encontradas dentro del archivo de texto con el cual es creado.

- Archivo de vocabulario (.vocab): este archivo contiene el vocabulario que conforman el texto u oraciones encontradas dentro del archivo de texto y su creación se realiza a partir del archivo de frecuencia de palabra.

- Archivo gramatical (.indgram): este archivo contiene el mapeo gramatical, es decir la ruta o la secuencia de probabilidades que se han obtenido del texto u oraciones encontradas dentro del archivo, junto con el archivo de vocabulario.

Estos archivos son necesarios para crear el archivo ARPA [19] que es el resultado final de la creación del modelo del lenguaje. El archivo del modelo del lenguaje tiene extensión .lm o .arpa, además puede ser de extensión .lm.bin para ser formato binario, el cual permite ejecutar más rápido el reconocedor y es usado para modelos del lenguaje de gran tamaño.

El proceso de creación de un Modelo del Lenguaje Estadístico se inicia al crear en primer lugar el archivo de frecuencia de palabra. Dicho elemento permitirá crear un archivo de vocabulario que obtendrá todas las palabras que componen el texto que se utiliza para la creación del modelo del lenguaje. Con el archivo de texto en conjunto con el archivo de vocabulario, se obtiene el archivo de mapeo gramatical. Con este archivo y los archivos de vocabulario se puede realizar la creación del modelo del lenguaje necesario para el entrenamiento del modelo acústico y el funcionamiento del RAH, este proceso se muestra en la Figura 3. Dicho proceso es inherente al software CMUCMLTK y todos los archivos mencionados con anterioridad en esta sección son resultado del funcionamiento del mismo, cabe resaltar que el archivo de vocabulario del cual se habla en esta sección, solo cumple una función en la creación del modelo de lenguaje y no se encuentra de ninguna manera relacionado y su contenido es diferente al del archivo de vocabulario expuesto en la sección de preparación pre-desarrollo.

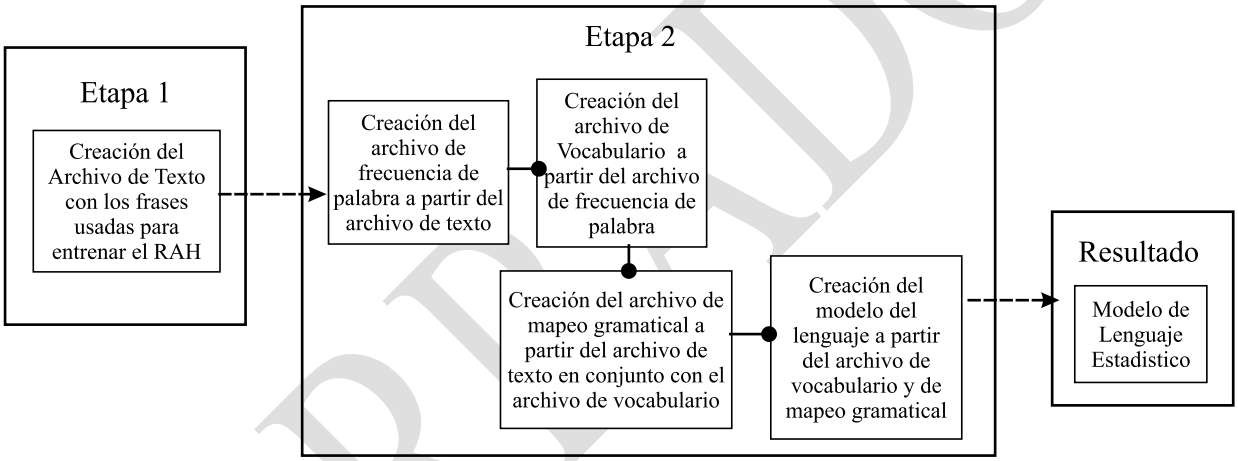

Figura 3. Proceso de creación del modelo del lenguaje estadístico utilizando el software CMUCLMTK

2.2.3. Entrenamiento del modelo acústico

El entrenamiento del modelo acústico se realiza mediante el software SphinxTrain. Para realizar el entrenamiento del modelo acústico se necesita el modelo del lenguaje, el diccionario fonético y los elementos creados en la sección de preparación pre-desarrollo; estos son principalmente los audios grabados por los usuarios para el entrenamiento del sistema además de los siguientes archivos:

- Archivo.phone - archivo que contiene la lista de fonemas presentes en el diccionario fonético.
- Archivo.filler - archivo que contiene la lista de etiquetas "<s>" que representan los silencios y limites de los segmentos hablados de las grabaciones de audio.
- Archivo _train.fileids - lista de archivos de entrenamiento grabados, relacionados con textos de entrenamiento.
- Archivo _train.transcription - archivo que contiene la lista de textos de entrenamiento.
- Archivo _test.fileids - lista de archivos de entrenamiento grabados relacionados con textos de prueba.
- Archivo _test.transcription - archivo que contiene la lista de textos de prueba.

Para iniciar el entrenamiento solo es necesario ejecutar el comando de configuración, el cual creará una carpeta donde se almacenará el modelo acústico entrenado y dentro de esta un archivo de configuración *sphinx_train.cfg*.

Así, se debe configurar según las características y las necesidades del modelo acústico a desarrollar. Se debe incluir dentro de la carpeta del modelo acústico los archivos de pre-desarrollo que se mencionaron anteriormente y una carpeta donde se almacenarán los audios, los cuales deberán estar ordenados según la configuración o la relación expuesta en el Archivo _train.fileids.

En el archivo de configuración se debe modificar las rutas de los archivos de pre-desarrollo, de igual manera y de ser necesario, se debe modificar el tipo de archivo de modelo del lenguaje con el que se desea decodificar y realizar el entrenamiento, escoger la frecuencia a la que se encuentran grabados los audios (preferiblemente 16 kHz) o modificar el filtro aplicado a las señales de audio; la voz humana se encuentra entre los rangos de 400 Hz a 6.000 Hz, por lo cual es recomendable aplicar un filtro en los 200 Hz para eliminar cualquier posible ruido que pueda perturbar el funcionamiento del reconocedor.

El modelo acústico se entrena en tres etapas como se representa en la Figura 4. En la primera parte el sistema verifica los requisitos del entrenamiento con el fin de determinar y comprobar que se encuentren en orden los archivos de pre-desarrollo necesarios en el entrenamiento. La segunda parte es la extracción de características en la cual se divide la señal en segmentos para realizar un procesamiento de dicha señal, extraer sus características y generar un conjunto de vectores como resultado del análisis realizado por el software SphinxTrain.

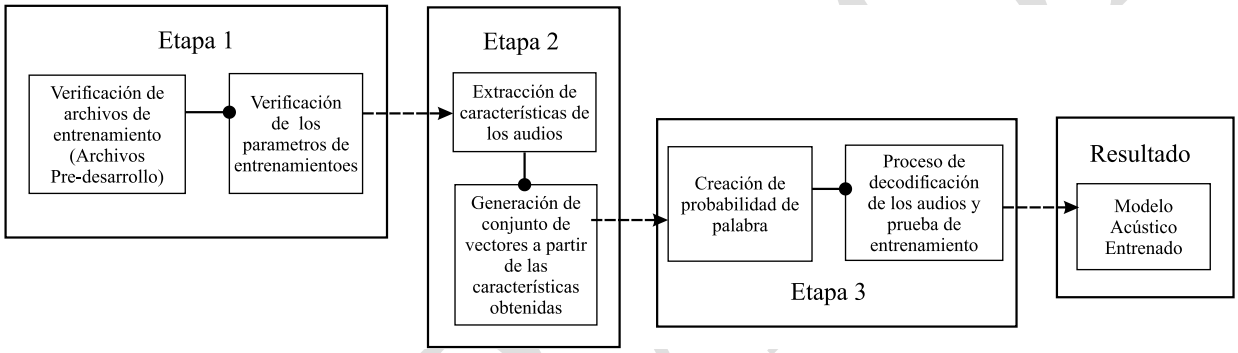

Figura 4. Proceso de entrenamiento del modelo acústico

En relación con el conjunto de vectores se realiza la tercera parte del entrenamiento; en esta se crea un modelo probabilístico teniendo como referencia a los HMM, el cual asocia a las probabilidades de las palabras obtenidas de la señal de audio entrante con las probabilidades existentes en el modelo del lenguaje para estas mismas; el resultado de esta etapa se denomina decodificación y tiene como resultado una carpeta que contiene los elementos de un modelo acústico.

El sistema realiza un *test* de entrenamiento en donde usa los archivos de prueba para verificar el funcionamiento y demostrar los posibles errores del modelo acústico; en la presente investigación estas pruebas son denominadas pruebas de entrenamiento y son el primer filtro de evaluación del sistema. La duración del proceso de entrenamiento puede variar dependiendo de la cantidad de audios a analizar, en promedio una hora de grabaciones tiene una duración de entrenamiento de catorce horas.

2.2.4. Evaluación del sistema de RAH desarrollado

La evaluación de los modelos acústico y de lenguaje se realizó tal y como se muestra en la Figura 5, analizando primero los resultados de entrenamiento, los cuales son producto de la parte final del proceso de entrenamiento del modelo acústico, seguidos de pruebas del sistema de RAH con audios realizados por nativos de la región, de manera extraoficial se realizaron pruebas con acentos diferentes al de la región de Cúcuta para comprobar el posible escalamiento del proyecto a nivel nacional.

Los comandos con los cuales se evaluó el sistema se presentan en la Tabla 2.

Tabla 2. Comandos del sistema de RAH implementado**.**

Comandos versión 1.0	Apagar luz dos	**Comandos versión 2.0**	Apagar luz comedor

Encender luz uno	Apagar luz tres	Encender luz sala	Apagar luz habitación
Encender luz dos	Consultar probabilidad lluvia	Encender luz comedor	Consultar probabilidad lluvia
Encender luz tres	Generar lista mercado	Encender luz habitación	Generar lista mercado
Apagar luz uno	Consultar valor temperatura	Apagar luz sala	Consultar valor temperatura

Por último, se compararon los resultados obtenidos con el sistema de RAH desarrollado con los resultados arrojados por el modelo de lenguaje y modelo acústico de la página SourceForge. Para esta comparación se utilizaron los mismos audios; estos resultados se presentan en la siguiente sección.

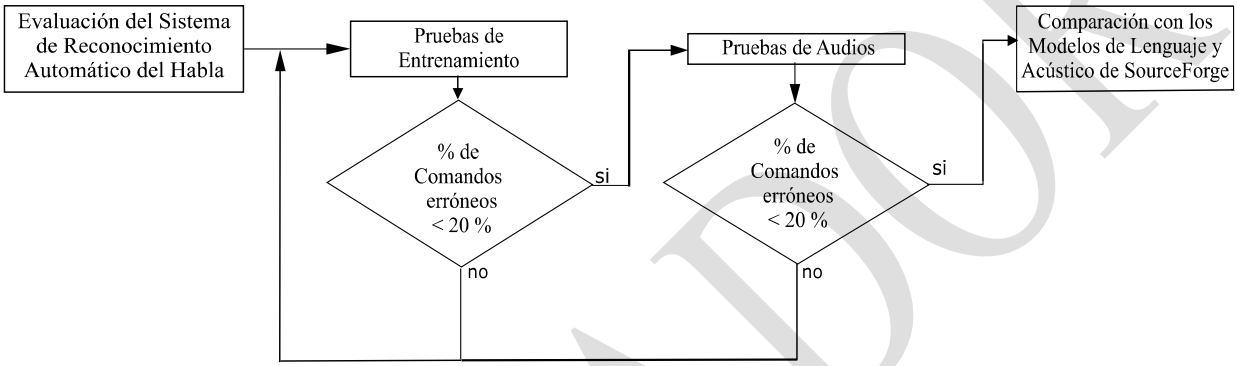

Figura 5. Proceso de evaluación del sistema de RAH

En caso de que los resultados de entrenamiento no superen el 80% de acierto o para este caso específico, no reconozca todos los comandos bajo los cuales operara el sistema, es necesario realizar nuevamente el entrenamiento del sistema, es decir, verificar que los audios estén grabados correctamente y de ser el caso, grabarlos nuevamente; de ser necesario, cambiar el modelo de lenguaje con el que se está decodificando el sistema o los comandos con los cual este opera; y por último modificar el archivo de configuración de entrenamiento Sphinx_train.cfg para que el sistema entrenado cumpla con las características deseadas. Luego se ejecutará nuevamente el entrenamiento mediante el software SphinxTrain.

3. Resultados

En la Tabla 3 se relacionan las personas empleadas para el entrenamiento con su ciudad de origen, el sexo y la cantidad de archivos empleados para ejecutar el entrenamiento de los modelos y las pruebas de los mismos.

Tabla 3. Características de los hablantes

Hablante	Ciudad de origen	Tipo de audio	Género	N° de audios	
				Entrenamiento	Pruebas de entrenamiento
N°1	Lourdes	Entrenamiento	Femenino	200	9
N°2	Cúcuta	Entrenamiento	Femenino	200	9
N°3	Gramalote	Entrenamiento	Femenino	200	9
N°4	Cúcuta	Entrenamiento	Masculino	200	9
N°5	Cúcuta	Entrenamiento	Masculino	200	9
N°6	Cúcuta	Entrenamiento	Masculino	200	9
N°7	Cúcuta	Entrenamiento	Masculino	200	9

Como resultados se obtuvieron dos modelos del lenguaje estadístico y dos modelos del lenguaje de tipo gramática (un modelo de cada tipo por versión), con los cuales operan el sistema y se realizó la decodificación de los modelos acústicos entrenados.

La principal diferencia existente entre las dos versiones de modelos del lenguaje estadísticos y modelos gramaticales que se desarrollaron en este trabajo es el cambio de los comandos con los que opera el sistema. La justificación de dicho cambio se orienta a mejorar el reconocimiento al momento de interpretar las órdenes dadas por el usuario.

Con base en los dos modelos acústicos (1.0 y 2.1) entrenados, se elaboró un manual para entrenar este tipo de modelos, de igual forma en el desarrollo de los modelos de lenguaje para ambos procesos se emplea la herramienta CMU Sphinx. Las versiones de los modelos acústicos 1 y 2 se diferencian al igual que los modelos del lenguaje en los comandos que debe reconocer.

Los modelos desarrollados están orientados para una aplicación de reconocimiento del habla "*control por comandos*" entrenados para la identificación de nueve comandos (Tabla 2). Como se aprecia en la Tabla 4, la versión 1 en los resultados de entrenamiento se presentó un porcentaje de comandos exitosos del 100 %. No obstante, este valor obtenido solo es válido para los audios con los cuales se desarrollaron los modelos lo cual se evidenció cuando se ejecutaron pruebas con audios externos. Al realizar las pruebas de reconocimiento, estas en algunas ocasiones presentaban errores en los primeros seis comandos debido a que eran fonéticamente y gramaticalmente muy parecidos. En la versión 2 fue necesario realizar un nuevo entrenamiento, porque solo se alcanzó una tasa de acierto del 66 % en el modelo 2.0 producto de un error en los archivos de pre-desarrollo. Esto fue solucionado regrabando los archivos, logrando alcanzar en la versión 2.1 tasa de acierto en el entrenamiento del 100 %. A la versión 2.0 no se le realizó prueba de audios debido a que los resultados de entrenamiento del modelo acústico presentaron una exactitud inferior al 80 %.

Tabla 4. Resultados del entrenamiento de los modelos acústicos

Versión del modelo acústico	Resultado entrenamiento		Resultado pruebas de audio	
	N° de audios	Tasa de acierto	N° de audios	Tasa de acierto
Modelo acústico 1.0	1463	100 %	450	30 %
Modelo acústico 2.1	1463	100 %	450	83 %

Los valores de la tasa de acierto se tabularon tal y como se definió en la sección de evaluación del sistema de RAH desarrollado. Dicha tasa se calculó según la cantidad de comandos reconocidos correctamente, es decir el sistema debía reconocer todas las palabras de manera correcta y en el orden correcto.

En la versión 2.1 se logró disminuir el valor de porcentaje de los comandos erróneos a casi una cuarta parte de la versión 1.0 y por debajo del 20 % gracias a que se mejoró la calidad de los audios de entrenamiento, se cambió la frecuencia de trabajo a 16 kHz y se utilizó un diccionario grande de entrenamiento, pero para la evaluación este mismo se redujo a los comandos de ejecución. De igual manera tal y como se muestra en la Figura 6, se comprobó que la metodología implementada y los cambios realizados en los archivos de pre-desarrollo fueron efectivos y se aplicaron correctamente, puesto que el error de las pruebas de audio de la versión final (versión 2.1) es inferior al error de pruebas de audio del sistema RAH conformado por los modelos acústicos y de lenguaje alojados en los repositorios de SourceForge, los cuales se evaluaron con los mismos audios con los que se trabajó en esta investigación, aunado a lo anterior, presentaron una tasa de acierto del 43%.

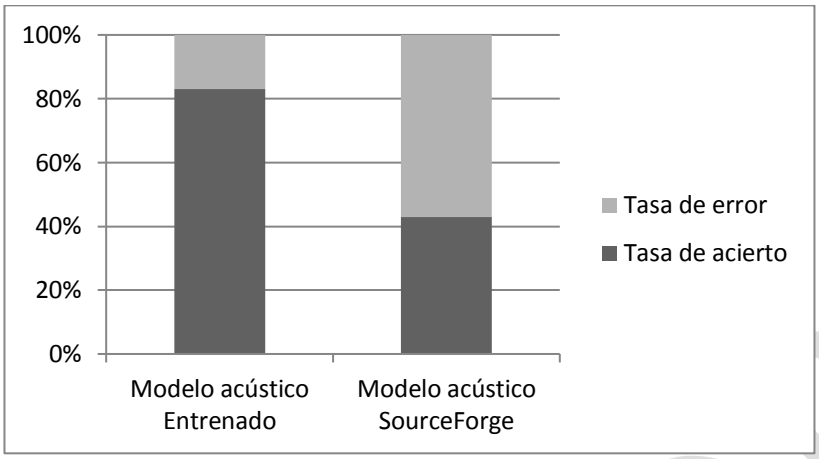

Figura 6. Resultado Modelo Acústico Entrenado *Vs* SourceForge

La tasa de acierto obtenida con los modelos desarrollados es del 83%, la cual se obtuvo al promediar la tasa de reconocimiento individual de 30 personas con las que se llevaron a cabo la evaluación de los modelos. Cada persona probó el sistema con quince comandos para un total de 450 archivos de audio, esta información se puede encontrar en la Figura 7. Todos los hablantes son originarios de la ciudad Cúcuta, excepto el No. 1 y No 2 que son de Lourdes y Gramalote respectivamente.

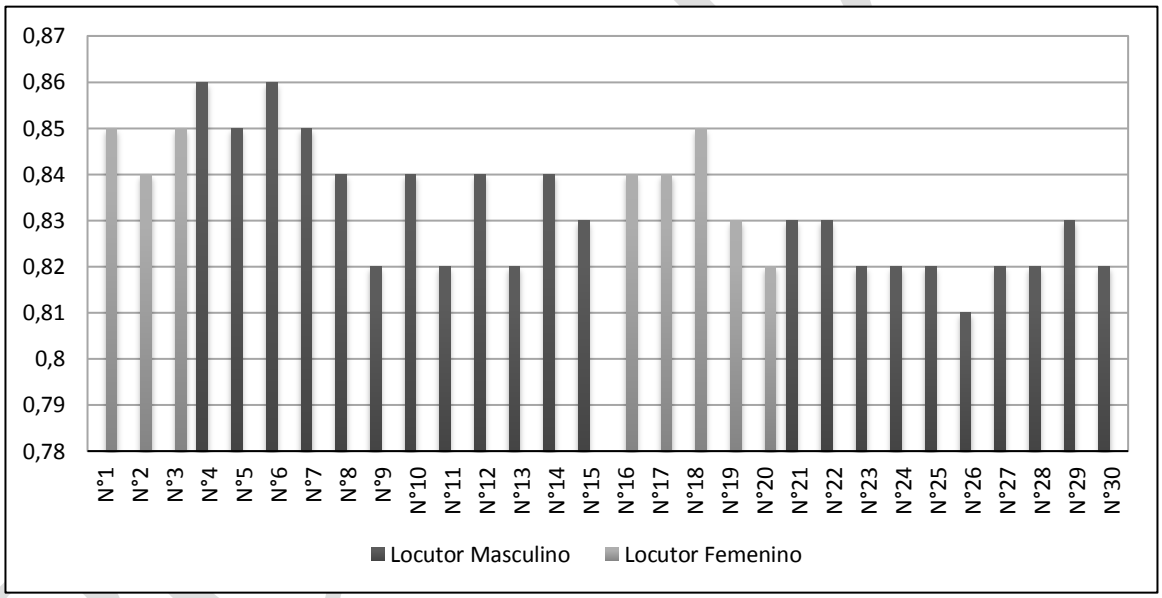

Figura 7. Tasa de acierto de reconocimiento por locutor en la prueba de audios.

4. Conclusiones

El número de participantes en el proceso de entrenamiento de los modelos acústicos y de lenguaje influye significativamente en la calidad del procesamiento de voz del reconocedor debido a que entre mayor sea la población que participe ente proceso mejor va a ser la respuesta del sistema, pues consigue un rango más grande de características fonéticas.

La ventaja de emplear un diccionario largo como el de SourceForge para el entrenamiento es que el sistema entrena mayor cantidad de palabras las cuales pueden ser usadas posteriormente como comandos del sistema, la desventaja es el tiempo de entrenamiento ya que es directamente proporcional a la cantidad de palabras entrenadas. Se utilizó un diccionario corto para la ejecución del reconocedor a fin de limitar las palabras que él puede entender a los comandos que se van emplear en la futura aplicación del mismo mejorando así la tasa de acierto del sistema.

Teniendo en cuenta que en las pruebas de reconocimiento se obtuvo una tasa de éxito superior al 80% es posible usar estos modelos para el desarrollo y la implementar un sistema de RAH para una aplicación orientada a la asistencia de personas con discapacidad visual o incapacidad de movimiento basada en la metodología de control "Command and control". Para esto se puede emplear el sistema embebido raspberry pi el cual se empleó en este trabajo y cuenta con diferentes protocolos de comunicación, una buena capacidad de procesamiento y control, además puede prestar diferentes servicios como por ejemplo servidor web, servidor ftp y servidor SQL que un proyecto como este demandaría.

5. Agradecimientos

Un agradecimiento especial a las personas que sirvieron de locutor y prestaron su voz para el desarrollo de esta investigación; de igual manera a los evaluadores, editor y a la revista en general, por la atención prestada y las contribuciones dadas en pro del fortalecimiento y mejoramiento del artículo, las cuales representaron un aporte significativo para la obtención de la versión final del mismo.

References

[1] F. Moumtadi, F. Granados-Lovera, J. Delgado-Hernández, "Activación de funciones en edificios inteligentes utilizando comandos de voz desde dispositivos móviles". *Ingeniería. Investigación y Tecnología,* abril-junio 2014, pp.175-186. [En línea]. Disponible en: http://www.revele.com.veywww.redalyc.org/articulo.oa?id=40430749002.

[2] J.M. Alcubierre, J. Minguez, L. Montesano, L. Montano, O. Saz, E. Lleida," Silla de Ruedas Inteligente Controlada por Voz". Primer Congreso Internacional de Domótica, Robótica y Teleasistencia para todos, 2005. [En línea]. Disponible en: https://www.researchgate.net/profile/Javier_Minguez/publication/237524693_Silla_de_Ruedas_Inteligente_Controlada_por_Voz/links/00b4952bfed6f95e49000000.pdf.

[3] M. Y. El Amrani, M.M. H. Rahman, M. R. Wahiddin y A. Shah, "Building CMU Sphinx Language Model for The Holy Quran using Simplified Arabic Phonemes". *Egyptian Informatics Journal*, vol. 17, no. 3, November 2016, pp. 305–314.

[4] M. Saqer, "Voice speech recognition using hidden Markov model Sphinx-4 for Arabic". M.S. thesis, University of Houston-Clear Lake, ProQuest Dissertations Publishing, 2012. [En línea]. Disponible en: https://search.proquest.com/docview/1029871476?accountid=43636.

[5] U. Uebler, "Multilingual speech recognition in seven languages". *Speech Communication*, vol. 35, no. 1–2, August 2001, pp. 53–69.

[6] J. Köhler, "Multilingual phone models for vocabulary-independent speech recognition tasks". *Speech Communication*, vol. 35, no. 1–2, August 2001, pp. 21–30.

[7] V. Z. Këpuska, P. Rojanasthien, "Speech Corpus Generation from DVDs of Movies and TV Series". *Journal of International Technology and Information Management*, vol. 20, no. 1-2, 2011, pp. 49-82. [En línea]. Disponible en: https://search.proquest.com/docview/1357567679?accountid=43636.

[8] CMU Sphinx Project by Carnegie Mellon University*, Open Source Speech Recognition Toolkit*. [En línea]. Disponible en: http://cmusphinx.sourceforge.net/.

[9] Y. Wang, X. Zhang, "Realization of Mandarin continuous digits speech recognition system using Sphinx", *2010 International Symposium on Computer Communication Control and Automation (3CA)*, 2010. [En línea]. Disponible en: http://ieeexplore.ieee.org/document/5533801/.

[10] A. Ceballos, A. F. Serna-Morales, F. Prieto, J. B. Gómez, T. Redarce, "Sistema audiovisual para reconocimiento de comandos". *Ingeniare: Revista Chilena de Ingeniería*, vol. 19, no. 2, 2011, pp. 278-291. [En línea]. Disponible en: https://search.proquest.com/docview/906290348?accountid=43636.

[11] A**.** Ceballos, Tesis para optar al grado de Magíster. Universidad Nacional de Colombia, Sede Manizales. Colombia. 2009.

[12] R. Calvo Arias, "Reconocimiento de voz". Proyecto de Graduación licenciatura en Ingeniería Electrónica, Instituto Tecnológico de Costa Rica. Escuela de Ingeniería Electrónica, 2002. [En línea]. Disponible en: http://repositoriotec.tec.ac.cr/handle/2238/5652.

[13] E. Gamma, D. Amaya Hurtado, O. Sandoval, "Revisión de las tecnologías y aplicaciones del habla sub-vocal". *Ingeniería,* vol. 20, no. 2, pp. 277–288. [En línea]. Disponible en: http://dx.doi.org/10.14483/udistrital.jour.reving.2015.2.a07.

[14] S. Oberle, "Detection and estimation of acoustical signals using hidden Markov model". Ph.D. dissertation, Eidgenoessische Technische Hochschule Zuerich, Switzerland, ProQuest Dissertations Publishing, 1999. [En línea]. Disponible en: https://search.proquest.com/docview/304550977?accountid=43636.

[15] A. Varela, H. Cuayáhuitl y J. A. Nolazco-Flores, "Creating a Mexican Spanish version of the CMU Sphinx-III speech recognition system", *Progress in Pattern Recognition, Speech and Image Analysis,* Springer*,* 2003, pp. 251–258.

[16] R. Mingov, E. Zdravevski y P. Lameski, "Application of Russian Language Phonemics to Generate Macedonian Speech Recognition Model Using Sphinx", *ICT Innovations 2016*, September 2016. [En línea]. Disponible en: https://www.researchgate.net/publication/308626983_Application_of_Russian_Language_Phonemics_to_Generate_Macedonian_Speech_Recognition_Model_Using_Sphinx.

[17] P. Lamere, P. Kwok, E. B. Gouv, R. Singh, W. Walker, y P. Wolf, *The CMU sphinx-4 speech recognition system*, Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, Hong Kong, 2003. [En línea]. Disponible en: http://mlsp.cs.cmu.edu/people/rsingh/papers_old/icassp03-sphinx4_2.pdf.

[18] M. Raab, R. Gruhn y E. Noeth, "A scalable architecture for multilingual speech recognition on embedded devices". *Speech Communication*, vol. 53, no. 1, January 2011, pp. 62-74.

[19] L. Villaseñor, M. Montes, M. Pérez, D. Vaufreydaz, *Comparación léxica de corpus para generación de modelos de lenguaje*, IBERAMIA workshop on Multilingual Information Access and Natural Language, 2002. [En línea]. Disponible en: http://hal.inria.fr/docs/00/32/64/02/PDF/Villasenor02a.pdf.