

Reconocimiento automático de voz para la escritura de expresiones algebraicas

Automatic speech recognition for the writing of algebraic expressions

¹Maximiliano Asís Lopez^a, ¹Henry Garrido Angulo^a, ¹Edgar Hernández Medina^a, ¹Oscar Minaya Salinas^a

RESUMEN

El reconocimiento automático de voz puede llegar a ser muy útiles, ya que no solo han impactado en industria de las telecomunicaciones y de los sistemas multimedia, sino que también otras áreas como telemática, juegos, automóviles, sistemas para personas con discapacidad, medicina y educación. El objetivo de este trabajo fue diseñar e implementar un sistema de reconocimiento de voz que permite la escritura de expresiones algebraicas a través de órdenes de voz. El sistema de reconocimiento de voz fue desarrollado con la plataforma Java utilizando el software de código abierto CMU Sphinx-4 de la Universidad Carnegie Mellon; el cual se basa en los modelos ocultos de Markov para el cálculo de los parámetros de voz. Para el entrenamiento y las pruebas del sistema de reconocimiento automático de voz se utilizó el modelo acústico generado a partir del corpus acústico de VoxForge. Gracias a la incorporación del software de Sphinx-4, el sistema es capaz de transcribir automáticamente las expresiones algebraicas con una tasa de acierto de 83,23%.

Palabras clave: Reconocimiento de voz; escritura de expresiones algebraicas; modelos acústicos; modelos de lenguajes; CMU Sphinx.

ABSTRACT

Automatic Speech Recognition can be very useful, since not only have they impacted the telecommunications industry and multimedia systems, but also other areas such as telematics, games, automobiles, systems for people with disabilities, medicine and education. The objective of this work was to design and implement a voice recognition system that allows the writing of algebraic expressions through voice commands. The speech recognition system was developed with the Java platform using the CMU Sphinx-4 open source software from Carnegie Mellon University; which is based on the hidden Markov models for the calculation of voice parameters. For the training and testing of the automatic speech recognition system, the acoustic model generated from the VoxForge acoustic corpus was used. Thanks to the incorporation of Sphinx-4 software, the system is able to automatically transcribe algebraic expressions with a success rate of 83,23%.

keyword: Speech recognition; writing of algebraic expressions; acoustic models; language models; CMU Sphinx.

¹Universidad Nacional Santiago Antunez de Mayolo. ^aLic. en matemáticas

INTRODUCCIÓN

El habla, y más concretamente el lenguaje como medio de comunicación, es una de las características fundamentales que diferencian al ser humano de los demás animales y representa un factor clave de su evolución (Schwartz, 1993); la misma es considerada el principal modo de comunicación y la forma más eficiente y natural de intercambio de información entre seres humanos (Gaikwad et. al., 2010). Los resultados de los estudios realizados por la consultora tecnológica Gartner ubican al reconocimiento de voz entre las tecnologías que se estabilizarán, y cuyos beneficios están ampliamente demostrados (Gartner, 2013).

Los sistemas más exitosos en el área de reconocimiento automático de voz (ASR: Automatic Speech Recognition), se basan en la utilización de la técnica de análisis estocástico: Modelos Ocultos de Markov (HMM: Hidden Markov Models) (Karnjanadecha y Zahorian, 2001; (Rabiner y Juang, 1993).

La aplicación de las tecnologías de reconocimiento de voz hacia la escritura de expresiones matemáticas se puede mencionar los siguientes: El primer de estos proyectos, Math Speak & Write, descrito en el documento por Guy et. al. (2004) permite la edición de expresiones básicas de la matemática en un entorno controlado por el reconocimiento de voz, aunque ofrece alternativas como el reconocimiento de escritura manual o el uso de los métodos de entrada tradicionales como el mouse o el teclado.

Utiliza el motor de reconocimiento de voz de Windows, y está disponible para su descarga.

La segunda de estas investigaciones, CamMath, apropia de nuevo la tecnología del reconocimiento de voz incluida con el sistema operativo Microsoft Windows para el control por medio de la voz del programa de escritura de expresiones matemáticas Scientific Notebook. Como demostración del funcionamiento del sistema, se ofrecen una serie de videos donde se muestran el proyecto en funcionamiento. Desafortunadamente, no hay una versión disponible para descargar (Cameron y Bilmes, 2007).

El último de estos proyectos, creado por Angela Wigmore y su equipo de trabajo en Kingston University bajo el nombre de TalkMaths, utiliza la confiable tecnología de reconocimiento de voz del motor Dragon NaturallySpeaking para el control y la inserción de expresiones matemáticas en el programa Scientific Notebook. Es importante recalcar que el proyecto estará consignado bajo el código abierto, aunque en la actualidad no haya todavía una aplicación final para descargar (Wigmore et. al., 2009).

El desarrollo de nuevas tecnologías que permitan el control total de la computación personal por aquellas personas que por una u otra circunstancia no pueden aprovecharlas es, sin duda alguna, una de las metas más importantes de las ciencias informáticas, biológicas y médicas actuales. Los avances en el desarrollo de software y la mejora siempre

creciente del hardware están levantando las barreras que antes se consideraban imposibles de superar, la capacidad de procesamiento, los nuevos algoritmos y la disponibilidad global de información permiten que día tras día se avance a pasos agigantados para lograr este importante objetivo: dar acceso universal a los recursos de la computación.

Por otro lado los programas ofimáticos y de redacción de documentos actuales cuentan con algunas herramientas para insertar expresiones algebraicas en dichos documentos tales como: Editor de ecuaciones de Microsoft, MathType, LibreOffice Math, Scientific WorkPlace, Tex. Sin embargo, estas presentan ciertos niveles de dificultad para su uso, con barreras de usabilidad tales como hacer clic en diferentes opciones para realizar una sola acción o aprender un lenguaje de programación o de formato específico para realizar este tipo de redacciones; actividades que exigen un tiempo de aprendizaje mayor para sus usuarios. El problema se acentúa más cuando un usuario con ciertas discapacidades físicas, visuales o motrices trata de realizar estas tareas, debido que en la mayoría de los casos requiere de asistencia o periodos más amplios de aprendizaje para interactuar con las herramientas. Además, las aplicaciones de reconocimiento de voz existentes tienen un alto costo de licenciamiento asociado.

En consecuencia, esta investigación surge como una alternativa a los complejos menús de opciones que tienen las

aplicaciones existentes y de la poco intuitiva estructura en los lenguajes técnicos utilizados para la escritura de expresiones algebraicas, y el resultado permitirá al usuario dictarle al computador las expresiones algebraicas de manera similar a como le dicta un profesor a un estudiante, y recuperarla en forma gráfica. Dicha representación gráfica puede ser entonces utilizada en el programa de edición de documentos electrónicos que el usuario desee utilizar. En base a lo expresado, el objetivo de la presente investigación fue diseñar e implementar un software de reconocimiento de voz para escritura de expresiones algebraicas a través de órdenes de voz.

MATERIALES Y MÉTODOS

Los procesos de Reconocimiento Automático del Habla están basados en Redes Bayesianas, específicamente en los Modelos Ocultos de Markov (HMM) (Oberle, 1999; Lee, 2008). 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 version 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 (Varela et. al., 2003) o adaptar uno ya existente (Mingov et al., 2016).

El software de reconocimiento de voz para escritura de expresiones algebraicas fue desarrollado bajo el software de código abierto CMU Sphinx (Lamere et. al., 2003) el cual es instalado y ejecutado en el ordenador. Los modelos del lenguaje y acústico se desarrollaron para satisfacer las necesidades de un software de reconocimiento de voz de múltiples hablantes y de un sistema de control por comandos.

Para el desarrollar el software de reconocimiento de voz se realizó una metodología general que va desde la captura de las muestras de voz hasta el reconocimiento de las mismas, desarrollando una serie de pasos: Obtener muestras de voz, realizar el Preprocesamiento a las muestras del voz, parametrización de las señales de voz, creación del Modelo fonético, Definición de los modelos trifenemas, entrenamiento de los modelos trifenemas, definición de la Gramática y Diccionario, diseñar y programar los diferentes algoritmos y métodos de reconocimiento de voz y la verificar la eficiencia(%) del modelo de

reconocimiento automático de voz.

La técnica empleada para la recolección de datos fue la observación con el instrumento Ficha de Observación.

La eficiencia del sistema, se evaluó experimentalmente, usando como medida la tasa de palabras erradas, más conocido por sus iniciales en inglés como WER (word error rate) que es una técnica utilizada de manera común en el reconocimiento de voz.

RESULTADOS

Se desarrolló un prototipo de una aplicación en Java que, utilizando las tecnologías de reconocimiento de voz independientes del hablante expuestas por la librería CMU Sphinx-4, permita el dictado en español, en un lenguaje cercano al natural, de expresiones de la matemática, específicamente de expresiones usadas en la aritmética y el álgebra, y su posterior conversión a un archivo de imagen conteniendo la forma visual de dicha expresión matemática, utilizando para ello un formato generado por Latex. El prototipo software desarrollado se compone de cuatro módulos independientes integrados en una aplicación Java gracias al uso de clases (figura 1). Estos módulos se integran mediante el uso de un módulo central, encargado de facilitar la interacción entre los diferentes módulos que componen el prototipo y presentarlos en forma de una aplicación completa, incluyendo la representación gráfica de la expresión algebraica.

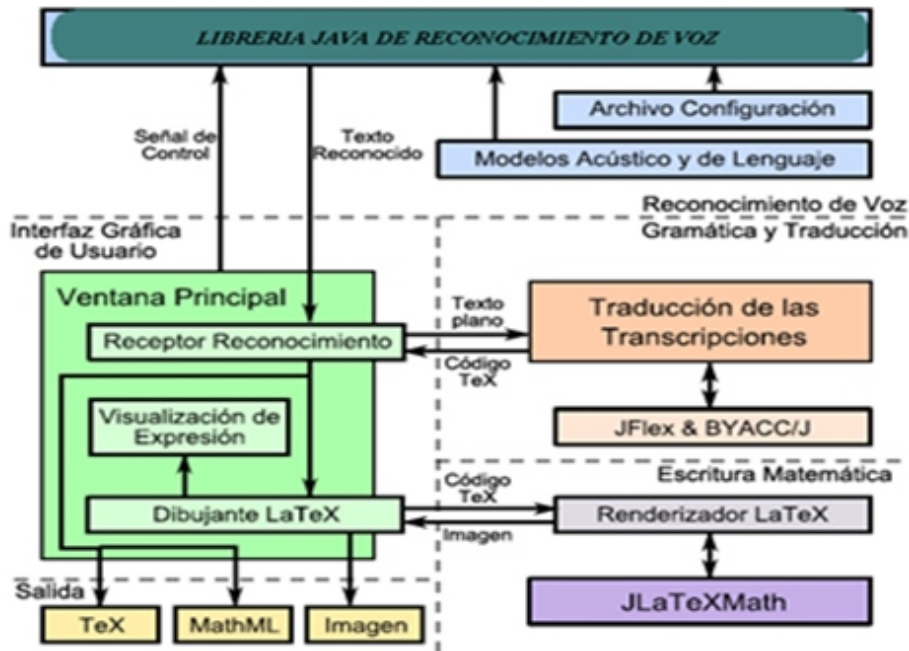


Figura 1. Diagrama de bloques del prototipo del sistema de reconocimiento de voz

Módulo de Reconocimiento de Voz

Módulo de reconocimiento de voz, encargado de la captura de voz, identificación de las palabras en el discurso hablado y transcripción de las palabras identificadas a su representación en forma de cadena de texto para su posterior procesamiento.

Durante esta investigación, se determinó que la mejor forma de crear un modelo acústico para la aplicación era el entrenamiento mediante el uso de SphinxTrain, alimentándole muestras de voz pregrabadas junto a su transcripción textual. Las muestras de voz se obtuvieron del corpus de código abierto VoxForge.

Para automatizar el proceso de entrenamiento, se programó un script que permitiese la creación de los archivos de trabajo como el programa SphinxTrain los requiere. Este script que hemos denominado *arcvoz.sh* descomprime los paquetes de audio provenientes de

VoxForge, renombra los archivos de audio de forma homogénea; crea un listado de estos, toma sus transcripciones y las incorpora a un archivo de transcripciones único que contiene un formato especial requerido por SphinxTrain. Luego, genera un diccionario a partir de las palabras existentes en las transcripciones y por último extrae las palabras que no tienen su equivalente en el vocabulario de pronunciación para ser editadas manualmente.

Una vez que todas las palabras de los archivos de transcripción tienen su equivalencia en el vocabulario, se puede comenzar el proceso de entrenamiento con SphinxTrain. Este proceso requiere el Java Runtime Environment y el código fuente de SphinxTrain. Luego, genera un diccionario a partir de las palabras existentes en las transcripciones y por último extrae las palabras que no tienen su

equivalente en el vocabulario de pronunciación para ser editadas manualmente.

Una vez que todas las palabras de los archivos de transcripción tienen su equivalencia en el vocabulario, se puede comenzar el proceso de entrenamiento con SphinxTrain. Este proceso requiere el Java Runtime Environment y el código fuente de SphinxTrain. Adicionalmente, requiere la creación de un archivo de configuración que le indica las rutas donde se encuentran los archivos de audio, de transcripción y otros parámetros sobre el nombre del modelo y el formato interno de los archivos de onda (resolución de bits y tasa de muestreo). Después de esto se ejecuta el script inicial de SphinxTrain, el cual genera como resultado los archivos que constituyen el modelo acústico. Dependiendo del equipo utilizado y la cantidad de archivos, este proceso puede llevar desde algunos minutos a varias horas.

Módulo de Traducción de las Transcripciones.

Este módulo se encarga de procesar el texto obtenido por medio del reconocimiento de voz y convertirlo en una serie de comandos reconocibles para el módulo de salida que se esté usando. Para ello, el módulo realiza una tarea de traducción, por lo que se determinó utilizar las mismas técnicas de diseño de compiladores para su elaboración. Por lo tanto, este módulo debe realizar al menos un análisis léxico y un análisis sintáctico sobre el texto obtenido y, a partir de las

expresiones algebraicas reconocidas, generar los comandos para el módulo de salida.

Para poder realizar el análisis léxico de la cadena de entrada se optó por la construcción de un intérprete utilizando las herramientas Lex y YACC, en sus versiones para el lenguaje Java JLex y JFlex. Esto permite crear toda una aplicación que trabaje con las clases del reconocedor de voz e interpretación y traducción de la cadena de entrada, en lugar de hacer módulos independientes que se distribuyen por separado.

Módulo de interfaz gráfica de usuario (GUI)

Encargado de presentar una ventana de resultados al usuario y permite controlar las diferentes funciones del sistema de reconocimiento de voz.

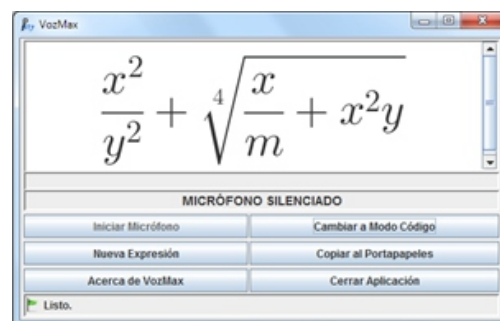


Figura 2. Interfaz gráfica.

Pausar/Iniciar. Micrófono A través de esta operación, el usuario puede desactivar el dispositivo de entrada de sonido temporalmente, para tomar un descanso o cuando se desea que el sistema no reconozca algunas frases que se van a pronunciar, como conversaciones o llamadas telefónicas. Presionando el mismo botón, se reactiva la captura de sonidos a través del micrófono.

llamadas telefónicas. Presionando el mismo botón, se reactiva la captura de sonidos a través del micrófono.

Cambiar a modo Código/Cambiar a modo Gráfica. Esta opción le permite al usuario cambiar entre el modo visual de la expresión matemática hacia el modo de edición de código en Tex de la misma y viceversa. Si el usuario tiene experiencia en escritura de ecuaciones usando el lenguaje Tex, puede editar el código manualmente. Igualmente, si el usuario requiere dicho código en Tex, puede copiarlo directamente al programa donde lo desea. Cuando el usuario regresa del modo de código al modo de Gráfica, la versión gráfica de la expresión se actualiza automáticamente.

Nueva expresión. Este botón le permite al usuario reinicializar el sistema para el dictado de una nueva expresión matemática, sin importar el estado en que se encuentra la expresión anterior.

Copiar al Portapapeles. A través de este botón, el usuario puede copiar la representación gráfica o el código en lenguaje Tex, dependiendo del modo de usuario en que se encuentre actualmente, al Portapapeles de su sistema operativo. El usuario podrá entonces incrustar o pegar dicho contenido copiado a otra aplicación de su computador.

Acerca de VozMax. Esta opción permite al usuario visualizar la información de versión del software, su creador y más información acerca de los proyectos que son instrumentales en la creación de este software.

Cerrar Aplicación. Este botón permite cerrar la aplicación.

Módulo de salida. Módulos encargados de la transformación de las expresiones matemáticas procesadas por la aplicación en una representación visual como una imagen, para su verificación por parte del usuario o en un objeto utilizable por otros programas o por el sistema operativo, para su inclusión dentro de documentos electrónicos e impresiones. Para este efecto se crearon dos módulos de salida básica: el módulo de exportación a lenguaje Tex y el módulo de exportación a archivo de imagen.

Módulo de salida a lenguaje Tex

Este módulo trabaja en conjunto con el subsistema de traducción de las transcripciones, pues este genera directamente código de Tex. Las únicas funciones que este módulo exhibe para el usuario de la aplicación son la visualización del código generado, su copia al Portapapeles del sistema y su inclusión en una aplicación o documento donde este lo requiera.

Módulo de exportación de imagen

Para permitir la inclusión fácil de las expresiones matemáticas procesadas por el sistema en documentos de texto o en aplicaciones gráficas se hace propicia la generación de una forma visual de dicha expresión y no así una versión más abstracta, como la generada en lenguaje Tex. Para ello, este módulo de exportación permite la creación de un archivo de imagen en formato PNG.

DISCUSIÓN

El desarrollo del sistema se dividió en cuatro módulos, los cuales son: módulo de reconocimiento de voz, módulo de traducción de cadenas a lenguaje Latex, módulo de interfaz gráfica de usuario y módulo de salida. Para la creación del modelo acústico se encontraron que existen varios corpus comerciales con altos costos de licenciamiento, los cuales no es posible adquirir; no obstante, entre todas las opciones que se encontraron, se descubrió un proyecto colaborativo de creación de corpus de distintos lenguajes, el proyecto VoxForge. Este proyecto de código abierto busca involucrar usuarios de todo el mundo que puedan aportar grabaciones cortas de algunas frases en sus lenguajes nativos para la construcción de un gran base de datos de grabaciones y transcripciones que pueda ser usado correctamente para la creación de un modelo acústico independiente del hablante para aplicaciones de código libre, aplicaciones que a menudo, como este proyecto, no cuentan con la capacidad financiera de adquirir la licencia de un corpus comercial.

Fue necesaria la descarga de una gran cantidad de archivos de audio con sus transcripciones desde los servidores de VoxForge. En total se adquirieron alrededor de 645 megabytes distribuidos en 3371 archivos de audio en formato de onda con una velocidad de muestreo de 48 kHz y una resolución de 16 bits, junto con sus archivos de texto de transcripción.

Los modelos acústicos y de lenguaje generados en este proyecto, se basó en los contenidos de VoxForge del día 5 de enero de 2016, tomó alrededor de 2 horas 1 de procesamiento en un computador HP Pavilion dv5-2046la 14.5", Corei5 2.26GHz, 4GB,500GB, Win7HP de 64 bits para su elaboración.

Al hablar del lenguaje matemático se tiende a pensar en las expresiones más complejas que se pueden trabajar como las integrales, transformadas, ecuaciones diferenciales, entre otras. Sin embargo, por tratarse este proyecto del lenguaje matemático en lenguaje hablado, no es adecuado pensar en términos de símbolos, números y letras, sino en términos de palabras. El primer problema al tratar de procesar el lenguaje hablado de las matemáticas consiste entonces en identificar cada una de las palabras que van a ser reconocidas.

Por tratarse de un desarrollo de un prototipo, la recomendación fue delimitar el conjunto de expresiones matemáticas que el sistema debía soportar inicialmente. Así, el conjunto de palabras aceptables es un subconjunto de las palabras utilizadas en el lenguaje matemático hablado. Específicamente, se tomó la decisión de soportar al menos las cuatro operaciones básicas de suma, resta, multiplicación y división, así como las operaciones de potencia y radicación.

CONCLUSIONES

Para construir este tipo de sistemas requiere de una enorme cantidad de

conocimientos matemáticos y mucha programación.

Podemos mencionar, entre otros, conocimientos de estadística y probabilidad, reconocimiento de patrones, lenguajes formales y autómatas, y eso, sin mencionar las habilidades de programación.

A continuación se enumeran las conclusiones que han resultado del desarrollo de la investigación.

1. Se programó la creación, integración, y ejecución de los componentes del Sistema de reconocimiento de voz (Modelos Acústicos, Modelo de Lenguaje, Diccionario, Algoritmo de Búsqueda). De igual manera, se programaron las siguientes funciones: captura y parametrización de voz en tiempo de ejecución para su reconocimiento.
2. La creación del modelo acústico y de lenguaje es uno de los pasos más importantes en el desarrollo de un sistema de reconocimiento de voz, pues requiere grandes recursos humanos y computacionales para su generación. En esta investigación se generó usando el corpus acústico provisto por el proyecto Voxforge.
3. El lenguaje hablado utilizado en el idioma español para comunicar expresiones algebraicas presenta ambigüedad en aquellas fraccionarias, radicales y de potencia, debido a que no es posible determinar el fin de un operando en estos tipos de expresiones sin el uso de elementos

paralingüísticos. En algunos casos se presentan confusiones respecto al sentido de dichas expresiones. Por lo tanto, no es posible desarrollar una gramática formal independiente del contexto que genere este mismo lenguaje.

4. La adición de una interfaz gráfica de usuario al prototipo desarrollado permite su uso sencillo, al utilizar para este una interfaz simple. Esto lo hace propicio para todo tipo de usuarios, incluyendo aquellos que requerirían un alto nivel de capacitación, investigación y ayuda para el uso de herramientas más avanzadas, obteniendo iguales o mejores resultados.
5. Se creó el sistema de reconocimiento de expresiones algebraicas que alcanzó un 83.23% de reconocimiento en promedio, según el estándar WER

AGRADECIMIENTO

Expresar agradecimientos por su ayuda en la realización del proyecto a: A la Universidad Nacional Santiago Antúnez de Mayolo; Por el financiamiento del proyecto de investigación.

A la comunidad GNU en general, y al grupo CMU Sphinx en especial, por el desarrollo del software libre que ha hecho posible este proyecto.

REFERENCIAS BIBLIOGRÁFICAS

- Cameron, Elliott y Bilmes, Jeff. 2007. Computer Based Mathematics Using Continuous Speech Recognition. Workshop on non-verbal acoustic interaction. <<https://melodi.ee.washington.edu/vj/files/CHIWS-07-Elliott.pdf>> [Consulta:12-04-2017].
- Gaikwad, S. K. ; Gawali, B. W. y Yannawar, P. 2010. A Review on Speech Recognition Technique.
- Guy, Cassandra, et. al. 2004. Math Speak & Write, a Computer Program to Read and Hear Mathematical Input. <<http://www.cs.berkeley.edu/~fateman/msw/AcademicPaper.pdf>> [Consulta:12-03-2017].
- Gaikwad, S. K. ; Gawali, B. W. y Yannawar, P. 2010. A Review on Speech Recognition Technique.
- Guy, Cassandra, et. al. 2004. Math Speak & Write, a Computer Program to Read and Hear Mathematical Input. <<http://www.cs.berkeley.edu/~fateman/msw/AcademicPaper.pdf>> [Consulta:12-03-2017].
- Gartner. 2013. Hype Cycle for Emerging Technologies Maps Out Evolving Relationship Between Humans and Machines <http://www.gartner.com/newsroom/id/2575515>,2013.
- Jancovic, P. y Kokuer, M. 2007. Incorporating the voicing information into HMMbased automatic speech recognition. IEEE Workshop in Automatic Speech Recognition and Understanding. pag.42-46.
- Karnjanadecha, M.; Zahorian, S. 2001. Signal modeling for High-Performance Robust Isolated Word Recognitions. IEEE Transactions On speech and Audio Processing. Vol9 N° 6.
- Lamere,P.; et. al. 2003. The CMU sphinx-4 speech recognition system, Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, Hong Kong. <http://mlsp.cs.cmu.edu/people/rsingh/papers_old/icassp03-sphmx4.pdf>[Consulta:12-05-2017].
- Mingov, R.; Zdravevski E. y Lameski, P. 2016. “Application of Russian Language Phonemics to Generate Macedonian Speech Recognition Model Using Sphinx”, ICT Innovations 2016, September 2016. <https://www.researchgate.net/publication/308626983_Application_of_Russian_Language_Phonemics_to_Generate_Macedonian_Speech_Recognition_Model_Using_Sphinx>[Consulta:12-06-2017]
- Oberle, S. 1999. “Detection and estimation of acoustical signals using hidden Markov model”. Ph.D. dissertation, Hochschule

- Zuerich, Switzerland, ProQuest Dissertations Publishing. <<https://search.proquest.com/docview/304550977?accountid=43636>> [Consulta:12-05-2017].
- Rabiner, L.; Juang, H. 1993. "Fundamentals of Speech Recognition". Prentice Hall. NY, USA.
- Schwartz, L. A. 1993. Language and Modern Human Origins, Yearbook of Physical Anthropology, 36,91-126.
- Varela, A.; Cuayáhuitl, H. y Nolasco-Flores, J. A. 2003. "Creating a Mexican Spanish version of the CMU SphinxIII speech recognition system", Progress in Pattern Recognition, Speech and Image Analysis, Springer,, pp. 251-258.
- Wigmore, Angela M., et. al. 2009. TalkMaths : A Speech User Interface for Dictating Mathematical Expressions into Electronic Documents. <<http://www.eee.bham.ac.uk/SLaTE2009/papers%5CSLaTE2009-07-v2.pdf>> [Consulta:12-05-2017]. Image Analysis, Springer,, pp. 251-258.
- Wigmore, Angela M., et. al. 2009. TalkMaths : A Speech User Interface for Dictating Mathematical Expressions into Electronic Documents. <<http://www.eee.bham.ac.uk/SLaTE2009/papers%5CSLaTE2009-07-v2.pdf>> [Consulta:12-05-2017].

CORRESPONDENCIA

Dr. Maximiliano Asís López
maxasis@hotmail.com