PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

FACULTAD DE CIENCIAS E INGENIERÍA

DISEÑO DE UN CORRECTOR ORTOGRÁFICO PARA UN SISTEMA DE RECONOCIMIENTO ÓPTICO DE CARACTERES

Tesis para optar el Título de Ingeniero Electrónico

Presentado por:

ROBERTO CARLOS SALAS DAMIÁN

Lima - Perú
2007

RESUMEN

Los sistemas de corrección usan como principio la lingüística computacional. En este contexto, un computador realiza un análisis ortográfico de los caracteres reconocidos por un OCR (Optical Character Recognition). Un OCR es un software que extrae de una imagen los caracteres que componen un texto para almacenarlos en un formato con el cual puedan interactuar programas de edición de texto.

El rendimiento de los sistemas de reconocimiento de caracteres es baja cuando se trata de digitalizar documentos deteriorados debido a las manchas y otros factores que evitan que se reconozcan las palabras del texto original. Ante este problema, lo que se propone en esta tesis es la implementación de un sistema de corrección ortográfica a la salida del OCR, que permitirá mejorar su eficiencia al momento del reconocimiento de los caracteres. De esta manera, la digitalización de los documentos históricos podrá garantizar una calidad óptima.

El sistema de corrección de ortográfica se basa en la búsqueda de patrones dentro de un texto. Esta búsqueda trata de encontrar todas las coincidencias de un patrón dentro de un texto, teniendo en consideración que la coincidencia del patrón con el texto puede tener un número limitado de diferencias. Este problema tiene aplicaciones en recuperación de información, biología computacional y procesamiento de señales, entre otras.

Como conclusión principal se obtiene que con el modelo de corrección basado en la búsqueda de patrones se alcanza un rendimiento de 80%, además el tiempo de procesamiento requerido para analizar una palabra es de tan solo 0.1seg lo cual refleja un alto rendimiento. Con esto, podemos concluir también que la metodología desarrollada para realizar la corrección de las palabras es una buena opción para este objetivo.

INDICE

INTRODUCCIÓN

Actualmente nos encontramos en un proceso de conversión digital de los fondos documentales en todos los ámbitos en los que actúan las diversas disciplinas de la documentación. La digitalización es, en nuestros días, un proceso en el cual se ven enfocados buena parte de las instituciones y organizaciones que cuentan con fondos documentales valiosos como por ejemplo patrimonios históricos y culturales.

Las grandes bibliotecas del mundo también se encuentran inmersas en este gran cambio de la digitalización de documentos, puesto que ellas son las más grandes fuentes de culturización e información en todos los aspectos.

La digitalización permite simplemente la eliminación del papel físico con lo cual obtenemos ventajas tales como:

- Importante reducción del espacio físico.
- Agilidad en la búsqueda de documentos.
- Acceso inmediato y concurrente de la información. Puesto que varios usuarios pueden tener acceso al mismo documento simultáneamente.
- Seguridad de los documentos; los originales se mantienen en archivos, no se pierden ni se deterioran.

En particular los documentos históricos con el transcurrir del tiempo se van deteriorando, haciendo difícil su lectura y produciendo la pérdida de valiosa información. La política que adoptan los encargados de la conservación de los documentos es digitalizar sólo aquellos documentos que ya se encuentran a punto de perderse, en donde su proceso de digitalización básicamente consiste en dos métodos: escanear los documentos y guardarlos en un dispositivo de almacenamiento o almacenar la información en microfilm. Este método de digitalización tiene muchas desventajas ya que no permite hacer arreglos a los

documentos porque no se encuentra en un formato editable. En el caso del método de escaneo se requiere de un dispositivo de memoria para poder almacenar los datos. Por otro lado, los documentos que no son digitalizados simplemente se almacenan en pequeñas cajas en un almacén lo cual significa que se necesitará mucho espacio físico para poder guardarlos.

Una solución para evitar la pérdida de los documentos deteriorados es el uso de sistemas de reconocimiento óptico de caracteres (OCR), los cuales funcionan bien cuando se quiere digitalizar documentos en buenas condiciones, pero cuando se trabaja con documentos deteriorados hay ciertas deficiencias por lo que el objetivo del presente estudio es diseñar e implementar un sistema de análisis ortográfico que use el principio de búsqueda de patrones para incrementar el rendimiento de un OCR cuando se use con documentos deteriorados. Como resultado se obtendrá un documento que sea lo más fiel a los originales, además de ser almacenados en un formato editable, lo que nos permitiría: ahorrar espacio en el almacenamiento, tener la documentación siempre al alcance y evitar el deterioro de los archivo.

CAPITULO 1:

EL ACERVO DOCUMENTAL HISTÓRICO Y LOS SISTEMAS DE RESPALDO TECNOLÓGICO

1.1 LA DIGITALIZACIÓN DE DOCUMENTOS

1.1.1 Tendencias mundiales en el desarrollo de alta performance

Hoy en día, los progresos en las denominadas tecnologías de la información, que abarcan los equipos y aplicaciones informáticas así como las telecomunicaciones, están teniendo un gran efecto. De hecho, se dice que estamos en un nuevo tipo de sociedad llamada *Sociedad de la información* o *Sociedad de Conocimiento*. Sin lugar a dudas, las nuevas tecnologías han llevado consigo un cambio espectacular y drástico en todas las empresas; en esta línea cabe destacar la Internet como el elemento revolucionario. Todo este avance tecnológico permite contar con computadoras más veloces, memorias de mayor capacidad, escáneres con una mejor resolución, todo lo cual se ve potenciado por el desarrollo de la Internet. Se prevé que las futuras bibliotecas sean bibliotecas virtuales y que todos los documentos de las empresas e instituciones sean capturados por la digitalización ya que su costo tiende a disminuir.

La Biblioteca Nacional de Austria tiene planes concretos para elaborar su política de digitalización en un plazo de tres años. La Biblioteca Nacional y Universitaria de la República de Macedonia no tiene un programa de digitalización, pero sí un proyecto de investigación cuyo objetivo es desarrollar bases de datos multimedia digitalizados en la colección de la biblioteca [9].

1.2 FACTORES RELACIONADOS CON LA DIGITALIZACION EN EL PAÍS

1.2.1 Demanda

La importancia de la Digitalización es una cuestión fuera de dudas, es de vital importancia para la realización de varias acciones con la documentación de hoy en día y este interés por la digitalización se debe a las siguientes razones:

- Seguridad. En caso de pérdida, los documentos digitales se pueden restaurar fácilmente, además el acceso a la documentación puede asegurarse estableciendo roles de administración.
- Consultas ágiles y eficientes. La consulta de los documentos es instantánea; en cambio, consultar los originales requiere personal, tiempo, espacio físico adecuado y medidas de preservación adicionales.
- Consultas simultáneas. La documentación digitalizada puede consultarse simultáneamente desde distintas ubicaciones geográficas.
- Consultas a través de Internet. Ofrece soluciones multinivel y multiusuario que permiten el acceso a la documentación a través de un navegador, ofreciendo versatilidad, escalabilidad y comodidad en el acceso.
- Amortización y Economía. Se pueden prestar tantas copias digitales como se desee sin tener que ofrecer el acceso físico a los originales. Así mismo permite el ahorro del papel y del espacio físico.

1.2.2 Consumidores

El usuario principal del sistema de digitalización que se esta proponiendo será el Instituto Riva Agüero y otros posibles usuarios podrían ser:

- Instituciones educativas tales como escuelas, colegios, institutos superiores y universidades.
- Instituciones del estado tales como la Sunat, la Reniec.
- Archiveros históricos, documentales y notarías.
- La Biblioteca Nacional del Perú.

1.2.3 Recursos Tecnológicos

Con el avance continuo de la tecnología electrónica, los recursos tecnológicos que se requieren para el proceso de digitalización son cada vez más accesibles, siendo sus costos más baratos y con mejores características.

Por otro lado, en actualidad existen recursos de software que son totalmente gratis. Mediante el uso de estos software libres se puede realizar la digitalización de documentos sin la necesidad de invertir dinero en la licencia del software.

1.2.4 Competidores

Las empresas que actualmente vienen brindando el servicio de digitalización de documentos en la ciudad de Lima son:

- Digiperu
- Datacom
- Microx Office
- Imaging Perú, etc.

1.2.5 Proveedores

Los proveedores serán aquellas empresas que se dedican a vender equipos de escaneo tales como escáneres, cámaras y tecnologías de reconocimiento de caracteres (ICR, OCR, OMR), las empresas que pueden servir como proveedores son:

- A&D Asociados
- Adisa Service

- Dimensión Corporativa
- Opensoft, etc

1.2.6 Reguladores: INDECOPI, Derechos de Autor

La legislación sobre derechos de autor (*copyright* o propiedad intelectual) ha sido elaborada con el fin de proteger y respetar la creación u obra original de una persona en cualquier formato en la que ésta se exprese. La normativa sobre derechos de autor que rige en el Perú está compuesta por el Decreto Legislativo 822 (Ley sobre el Derecho de Autor) promulgada en 1996, el Convenio de Berna de 1886 del cual el Perú es país signatario y la Decisión 351 de la Junta del Acuerdo de Cartagena.

La selección del material para ser digitalizado es una decisión importante en cualquier proyecto. Los criterios diferirán dependiendo de los objetivos establecidos y estarán limitados por factores adicionales, como pueden ser restricciones legales, políticas institucionales, dificultad técnica de la digitalización. En este sentido los siguientes puntos deben tenerse en cuenta:

1) Los criterios de selección deberían ser explícitos y discutidos por todas las partes involucradas en el proceso
2) Los criterios deberán estar completamente documentados, de forma que las razones para digitalizar o no estén claras a lo largo de todo el proceso.

1.3 ARCHIVO HISTÓRICO DEL INSTITUTO RIVA AGÜERO

1.3.1 Proceso de tratamiento para documentos con respaldo tecnológico

La metodología usada actualmente en el proceso de digitalización por parte del Instituto Riva-Agüero se basa en el escaneo de documentos, donde los documentos se guardan en formatos de imágenes. En la figura 1.1 se puede observarse el procedimiento que usan.

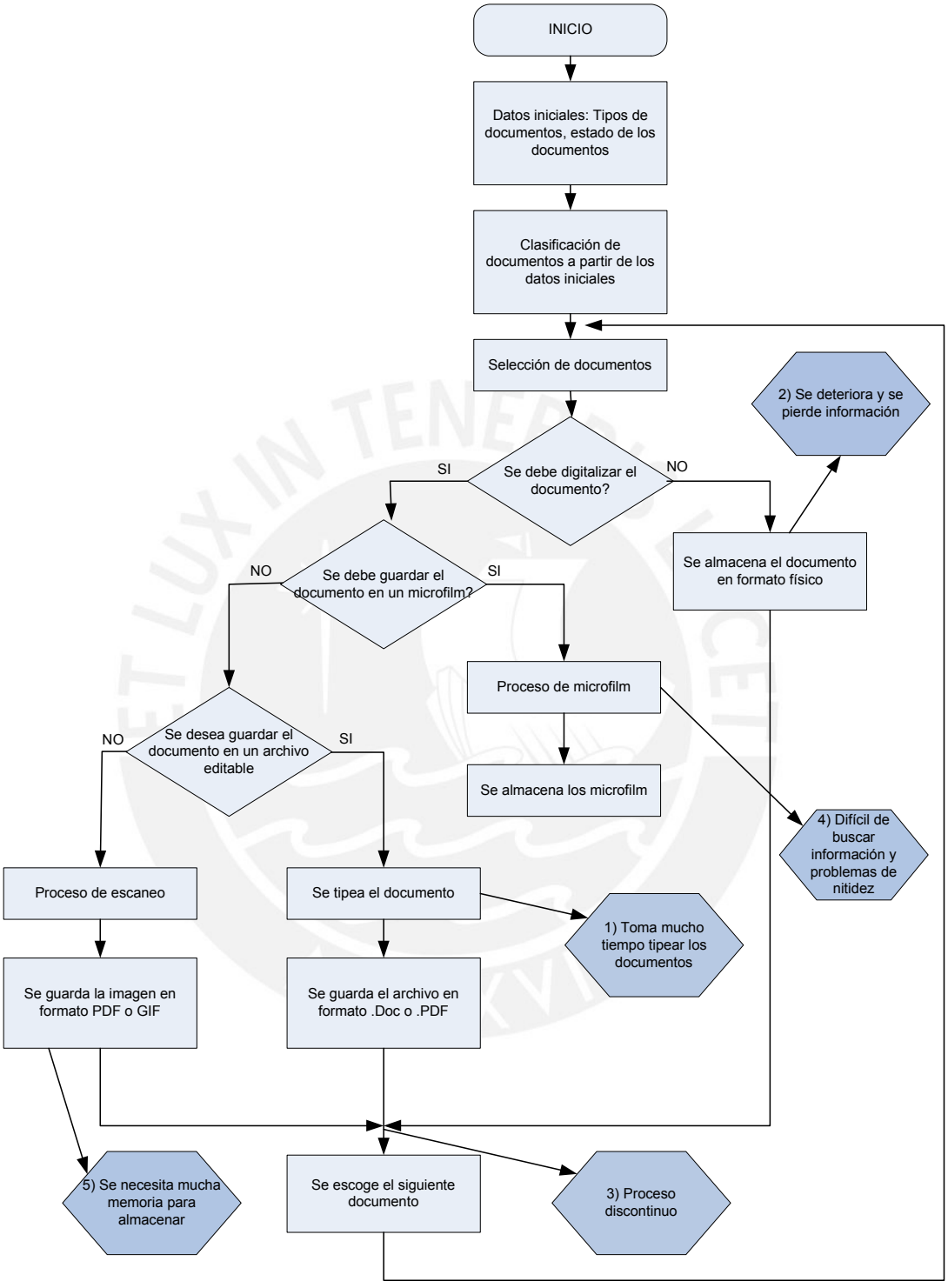


Figura 1.1 Metodología de digitalización aplicado en el Instituto Riva Agüero

El cuadro siguiente resume el análisis de los procedimientos seguidos para el almacenamiento de los documentos en el Instituto Riva Agüero:

Análisis de los procedimientos de almacenamiento del IRA

Hechos	Problemas y Causas
1) Toma mucho tiempo tipear los documentos.	Los documentos que nos son digitalizados mediante el uso del escáner o el microfilm se proceden a tipear y esto implica que se requiere una persona para que realice esta tarea, además este proceso es muy lento ya que una persona es incapaz de tipear los documentos a gran velocidad por lo que se tendría una gran desventaja si se quisiera guardar gran cantidad de información escrita.
2) Se deteriora y pierde información.	Aquellos documentos que se almacenan en su formato físico original corren el riesgo de perder la información que llevan debido a que están expuestos a factores externos como la humedad, el polvo, a roedores, a polillas y sobre todo es que se tendrá contacto directo con ellos lo que poco a poco irá deteriorándolos y por ende a perder su valiosa información.
3) Proceso discontinuo	El proceso de digitalización que se desarrolla en el Instituto de la Riva-Agüero es de forma discontinua puesto que los tiempos en los cuales se desarrolla la digitalización se realiza sólo cuando hay personas disponibles para esa tarea de lo contrario esta se suspende y se realiza en otra oportunidad.
4) Difícil de buscar información y problemas de nitidez.	El proceso de digitalización mediante el microfilm presenta cierta dificulta al momento de buscar una información específica debido a que se tiene que pasar secuencialmente las imágenes que se encuentran almacenados en el rollo fotográfico hasta llegar al punto deseado lo cual puede implicar mucho tiempo y otro factor es que siempre que se quiera ver la información contenida en este medio se requiere de un hardware especial.
5) Se necesita mucha memoria para almacenar	En el proceso de digitalización por el escáner se requiere de mucha cantidad de memoria para poder almacenar las imágenes ya que estos se guardan en formato gif. Y esto imposibilidad la distribución de la información a través de la Internet ya que satura el sistema.

1.3.2 Capacidad operativa

Un problema grave para la conservación de documentos históricos es que las personas encargadas de los documentos carecen del conocimiento para poder llevar a cabo la digitalización y esto incluye la poca capacitación que han recibido para manejar adecuadamente las computadoras, escáneres, cámaras, y es uso de software. A esta carencia del uso de equipos para digitalización se suma el desconocimiento de las diferentes técnicas que existen para la digitalización de documentos como: redes neuronales para el reconocimiento de caracteres, métodos de Markov para manuscritos, etc.

1.3.3 Infraestructura: Escasez de equipos

Actualmente los centros educativos y varias instituciones de nuestro país presentan una gran deficiencia en cuanto a la digitalización, debido a que no cuentan con el equipo tecnológico adecuado. Esto se debe a que su presupuesto es limitado; por ejemplo, en el Instituto Riva-Agüero cuentan con pocos escáneres y computadoras. Como ya se mencionó anteriormente, los documentos se guardan en formato de imágenes y estos ocupan mucha memoria en la computadora. Por último también están los requerimientos de software ya que es necesario utilizar programas especializados para obtener mejores resultados y estos requieren del pago de una licencia para su uso.

Por otro lado, en el Instituto Riva-agüero también existe información almacenada en microfilm y para poder ver la información contenida en él se requiere de un equipo especial. Actualmente ese equipo ya se esta dejando de usar debido a la aparición de nuevas tecnologías más eficientes.

1.4 Declaración del problema

En el Instituto Riva Agüero el proceso de digitalización o almacenamiento de archivos se desarrolla en forma lenta; es decir, la digitalización en esta institución es mínima, donde cerca del 70% de sus documentos se encuentran almacenados en forma física.

La política que adoptan los encargados de la conservación de los documentos es digitalizar sólo aquellos documentos que ya se encuentran a punto de perderse, en donde su proceso de digitalización básicamente consiste en dos métodos: escanear los documentos y guardarlos en un dispositivo de almacenamiento o almacenar la información en microfilm. Por otro lado, los documentos que no son digitalizados simplemente se almacenan en pequeñas cajas en un almacén, lo cual significa que se necesitará espacio físico para poder guardarlos y además los documentos se irán deteriorando con el pasar del tiempo y cuando ya estén por perderse recién se procederá a su digitalización.

Con estos criterios que se toman para la digitalización de documentos se tiene un alto riesgo de la pérdida de valiosa información de los documentos históricos lo que conlleva a la pérdida de nuestra propia historia y cultura.

1.5 Conclusiones

Dado el incremento por la demanda de la digitalización, es necesario brindar un sistema de digitalización adecuado que satisfaga las necesidades de los usuarios.

El funcionamiento del sistema de digitalización deber ser eficiente de manera que el usuario no pierda tiempo tipeando los documentos ni ocupe demasiado

espacio en la memoria del computador por causa del escaneo de los documentos.

La falta tecnológica vista en el Instituto Riva Agüero, refleja la necesidad de fomentar más información acerca de las modernas técnicas de digitalización que se basan en OCR.

CAPITULO 2:

LA DIGITALIZACIÓN DE DOCUMENTOS Y LOS SISTEMAS DE CORRECCIÓN ORTOGRÁFICA

2.1 Estado del Arte

2.1.1 Presentación del asunto de estudio

El reconocimiento de caracteres tiene como propósito asociar a una imagen la identidad (clase) correspondiente entre los símbolos de un determinado alfabeto. Si la información del carácter se obtiene a través de medios ópticos, suele hablarse de Reconocimiento Óptico de Caracteres. En este caso, el problema del reconocimiento de los caracteres es sólo una parte de un problema mayor que se conoce como Análisis o Comprensión de Documentos. Se trata, evidentemente, de obtener una representación simbólica lo más fiel y completa posible a partir de la imagen digitalizada de un documento escrito.

Las etapas que consta un sistema de análisis de documentos son: adquisición de la imagen y preproceso; análisis de la página, segmentación de ésta en bloques de gráficos, bloques de texto, líneas de texto y caracteres así como ordenación de estos elementos; reconocimiento de los caracteres impresos independientemente del tipo y tamaño de letra y, finalmente, recuperación de

errores en el texto por corrección ortográfica y aplicación de un modelo de lenguaje.

El presente estudio muestra diferentes tecnologías que se emplean para la digitalización de documentos así como también el uso de conocimiento lingüístico para la corrección de las palabras.

2.1.2 Estado actual de la investigación

El Instituto Riva Agüero actualmente almacena gran cantidad de documentos históricos. Estos documentos con el pasar de los años se han deteriorado y la información que contienen está en riesgo de perderse. Para evitar la pérdida de dicha información se utilizará una técnica de digitalización de documentos que usará un OCR (reconocimiento óptico de caracteres). Este OCR presenta problemas al momento de reconocer los caracteres debido a que los documentos históricos se encuentran deteriorados, para solucionar este problema se emplea un corrector ortográfico que permita analizar los caracteres que son reconocidos por el OCR. Este corrector ortográfico formará palabras en base a los datos proporcionados por el OCR, para poder verificar si la palabra dada corresponde a una palabra de la Lengua Española.

Estos problemas asociados a la masiva utilización del papel, incluyen: ocupación de grandes volúmenes de espacio (almacenamiento) para contener los fondos documentales, incremento de las labores de explotación del mismo (indexación, búsqueda y flujos del papel), necesidad de reducción del tiempo de proceso tendente a proporcionar un mejor servicio al cliente, etc.

2.1.2.1 Tecnologías de la digitalización

El rápido avance de tecnologías asociadas a los SGED, proporciona hoy la posibilidad de implantar soluciones documentales razonables en costes y eficientes en su funcionamiento. Cabe destacar:

1. La Microfilmación

La microfilmación es una técnica de archivo de documentos basado fundamentalmente en el cambio de soporte de los documentos electrónicos o de papel en otro de un material sintético muy resistente y durable. Normalmente es una cinta de 30,5 metros de un material plástico flexible, sobre la cual se ha depositado una capa de material tipo fotográfico de alta calidad.

Para la recuperación de las imágenes del microfilm sólo basta de un aparato que básicamente consiste en un lente de microscopio que proyecta la imagen de la película en una pantalla. La búsqueda de los documentos se realiza mediante un computador. Si se desea rescatar la información se puede hacer a través de la pantalla, pero si se quiere restituir el documento en papel, se puede hacer mediante el uso de una impresora.

2. Tecnologías de almacenamiento masivo

Una imagen digitalizada requiere de miles de Bytes de memoria; todo un fondo documental en formato electrónico ocupa frecuentemente cientos de GigaBytes o varios TeraBytes. El progresivo abaratamiento de las memorias no volátiles (discos magnéticos en RAID, discos ópticos WORM, WMRA, CD-R, DVD, BD-R, etc.) y un espectacular aumento de su densidad de grabación y disminución del tiempo de acceso, permiten en la actualidad el almacenamiento masivo digital de cualquier volumen.

3. Tecnologías de digitalización electrónica

De las técnicas micrográficas (microfilmación en microfilm o microfichas iniciadas en los años 30 por Kodak), se ha pasado a partir de los años 80 a la utilización de digitalización de imagenes mediante electrónica de estado sólido. Esta tecnología ha permitido el progreso y la utilización de escáneres y cámaras

electrónicas cada vez más económicas, más rápidas y de mayor calidad. Un Sistema de Gestión Electrónica de Documentos incluye habitualmente los siguientes componentes: Escáneres, dispositivos de almacenamiento, unidad de proceso, comunicación entre equipos e impresoras.

2.1.2.2 La lingüística computacional

La lingüística computacional es la ciencia que trata de la aplicación de los métodos computacionales en el estudio del lenguaje natural (Gelbukh and Bolshakov, 1999). Esta ciencia es una combinación de dos ciencias más grandes; la lingüística, que estudia las leyes del lenguaje humano, y la inteligencia artificial, que investiga los métodos computacionales para el manejo de sistemas complejos (ver figura 2.1). El problema u objetivo más importante de la lingüística computacional es la *comprensión del lenguaje*, es decir, la transformación del lenguaje hablado o escrito a una representación formal del conocimiento, como por ejemplo una red semántica.

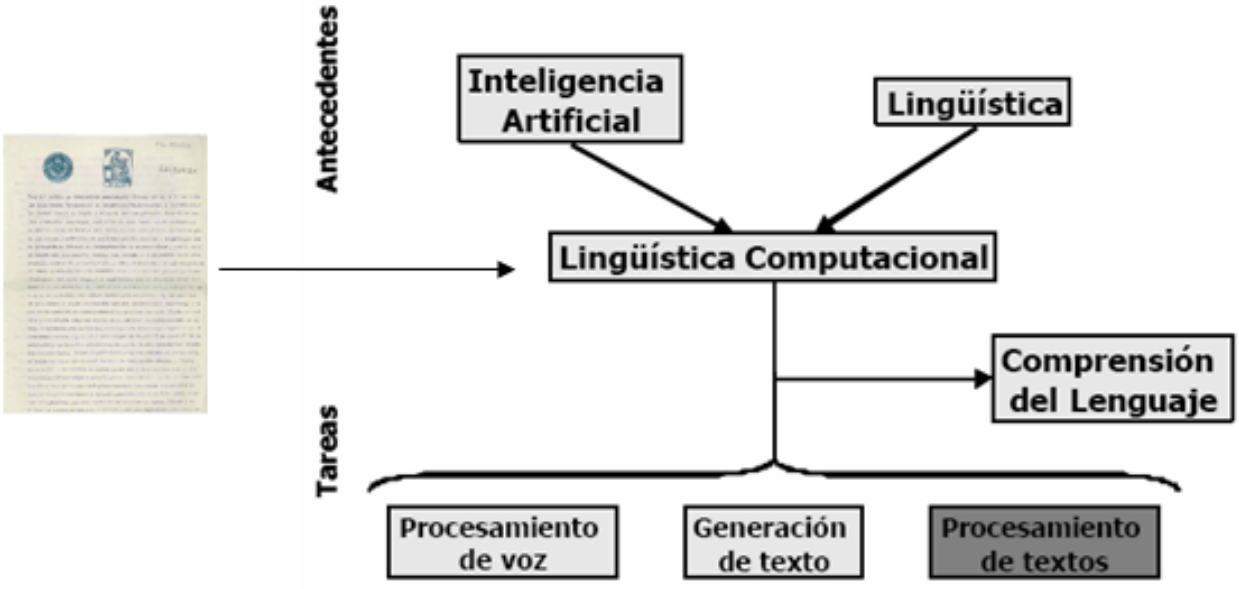


Figura 2.1. Antecedentes y tareas de la lingüística computacional

La solución tradicional de este problema consiste en construir un procesador lingüístico constituido por diferentes módulos independientes (ver figura 2.2):

- El *módulo morfológico* se encarga de reconocer las palabras. Básicamente, convierte las cadenas de letras a una entrada de un diccionario, y pone las marcas de tiempo, género y número.
- El *módulo sintáctico* reconoce oraciones. Este módulo convierte las cadenas de palabras marcadas a una estructura grafica, en donde se hacen explicitas algunas relaciones entre las palabras de la oración.
- El *módulo semántico* reconoce la estructura completa del texto y lo convierte a una "red semántica".

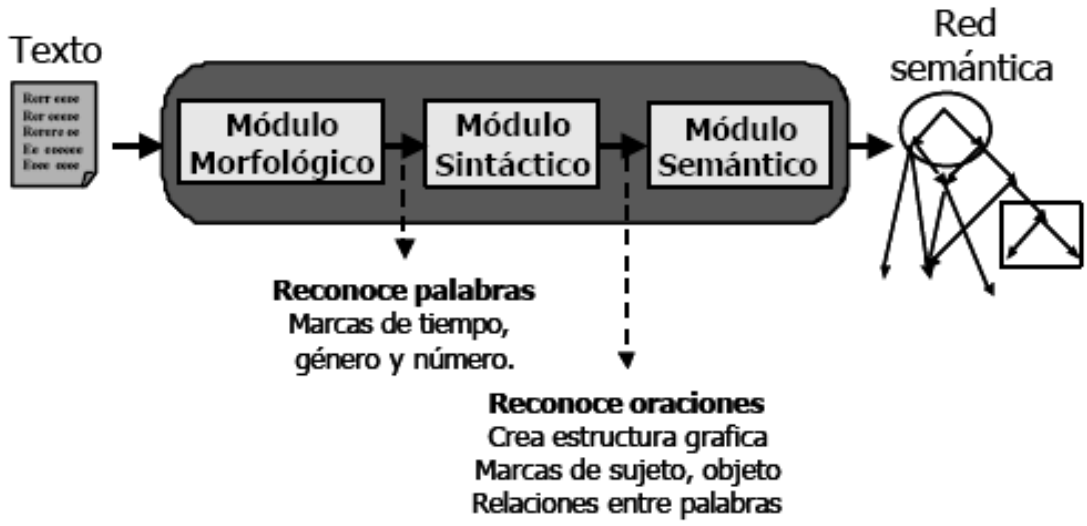


Figura 2.2 Procesador lingüístico

Otras áreas de investigación de la lingüística computacional se muestran en la figura 2.2. La más grande de estas áreas, y tal vez la más importante, es el procesamiento automático de textos. El procesamiento automático de textos considera una gran diversidad de tareas (Figura 2.3), desde muy simples, como la separación de palabras, hasta muy complejas como algunas tareas de *minería de texto*. La minería de textos (text mining) dentro del acceso, recuperación y organización de información es un conjunto de técnicas que permiten extraer información relevante y desconocida de forma automática

dentro de grandes volúmenes (habitualmente) de información textual, normalmente en lenguaje natural y no necesariamente estructurada.

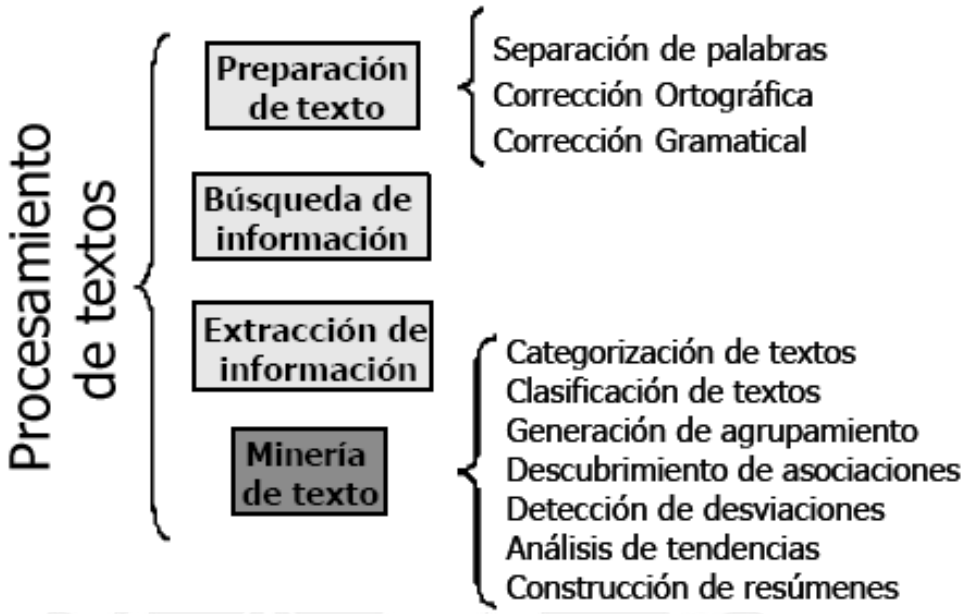


Figura 2.3 Tareas del procesamiento de textos

2.1.3 Síntesis sobre el asunto de estudio

La microfilmación y la digitalización son actualmente los dos medios disponibles más adecuados para la preservación del documento original así como para su acceso y difusión. La microfilmación había sido hasta hace unos pocos años el único existente; sin embargo, las nuevas tecnologías han permitido el desarrollo de la digitalización que está comenzando a desplazar al microfilm gracias a que supera sus limitaciones y ofrece nuevas ventajas en cuanto a control de la calidad de la imagen obtenida, a las posibilidades de navegación y al acceso al documento digitalizado.

Entre las principales ventajas de la digitalización podemos mencionar:

- Elevada capacidad de almacenamiento.
- Permite la copia de imágenes digitalizadas a alta velocidad y sin pérdida de calidad.

- Control de calidad durante la adquisición de la imagen digitalizada, por lo que hace posible la mejora de las imágenes con legibilidad reducida.
- Permite la automatización del servicio a usuarios y del proceso de copia.
- Posibilita el acceso en línea, a varios usuarios al mismo tiempo, sobre redes de comunicación, por ejemplo a través de Internet.
- Permite la migración o refrescado de los datos almacenados sin pérdida.

No obstante, a pesar de estas ventajas, la digitalización presenta algunos inconvenientes que, durante sus primeros años, hicieron plantearse a muchos responsables del diseño de proyectos de reproducción de documentos la conveniencia del empleo del microfilm o de la digitalización. Entre estas desventajas destacan:

- La perdurabilidad del soporte empleado para el almacenamiento de la imagen digital es escasa, puesto que la tecnología, de lectura y almacenamiento, se vuelve obsoleta rápidamente. Esto obliga a la migración o refrescado de los datos almacenados a un nuevo soporte y a la renovación de los equipos.

- El coste de la microfilmación era, por aquel entonces, menor y más estable que el de la digitalización. Hoy en día es una tecnología que se encuentra en continuo cambio, mejorando en capacidad tecnológica, calidad, rapidez y precios.

2.2 Conceptualizaciones generales

2.2.1 Lingüística Computacional

1 Definición

La Lingüística Computacional (Computational Linguistics) puede considerarse una disciplina de la linguística aplicada y la Intelingencia Artificial. Tiene como

objetivo la realización de aplicaciones informáticas que imiten la capacidad humana de hablar y entender. A la Lingüística Computacional se le llama a veces Procesamiento del Lenguaje Natural (PLN), o Natural Language Processing (NLP). Ejemplos de aplicaciones de PLN son, por ejemplo, los programas que reconocen el habla así como los traductores automáticos.

2 Terminología básica de la lingüística computacional

2.1 Corpus lingüístico

La lingüística de corpus es una línea de trabajo muy importante en la lingüística funcionalista actual, que se distingue nítidamente por su metodología: tiene un carácter empírico, puesto que realiza sus investigaciones sobre la base de colecciones extensas de textos naturales, las que se denominan corpus. Esas muestras de textos son analizadas mediante el empleo intensivo de programas computacionales, es decir, es un tipo de estudio lingüístico que se destaca por el empleo de las modernas tecnologías de la información.

Los corpus deben ser “preparados” para su tratamiento informático, esto es, anotados y preanalizados mediante procesos de le matización (organización en clases de las formas idénticas o relacionadas de una palabra bajo una entrada) y etiquetados (marcado de la categoría de palabra y rasgos sintácticos significativos). Existen procedimientos y criterios rigurosos para el diseño, recolección, tamaño y organización de los corpus de manera que sean confiables y apropiados para el tipo de investigación que se pretende emprender.

2.2 Analizador sintáctico

Un analizador sintáctico es un programa que reconoce si una o varias cadenas de caracteres forman parte de un determinado lenguaje. Los lenguajes habitualmente reconocidos por los analizadores sintácticos son los lenguajes libres de contexto. Los analizadores sintácticos fueron extensivamente

estudiados durante la década de 1970, detectándose numerosos patrones de funcionamiento en ellos que permitieron la creación de programas generadores de analizadores sintácticos a partir de una especificación de la sintaxis del lenguaje, tales y como yacc, GNU bison y javacc.

2.3 Procesamiento de lenguajes naturales

El Procesamiento de Lenguajes Naturales, abreviado PLN o NLP del idioma inglés *Natural Language Processing*, es una subdisciplina de la Inteligencia Artificial y la rama ingenieril de la lingüística computacional. El PLN se ocupa de la formulación e investigación de mecanismos eficaces computacionalmente para la comunicación entre personas o entre personas y máquinas por medio de lenguajes naturales. El PLN no trata de la comunicación por medio de lenguajes naturales de una forma abstracta, sino de diseñar mecanismos para comunicarse que sean eficaces computacionalmente —que se puedan realizar por medio de programas que ejecuten o simulen la comunicación—. Los modelos aplicados se enfocan no sólo a la comprensión del lenguaje de por sí, sino a aspectos generales cognitivos humanos y a la organización de la memoria. El lenguaje natural sirve sólo de medio para estudiar estos fenómenos.

2.4 Traducción automática

La traducción automática consiste en convertir un texto de un idioma a otro automáticamente, por medio del ordenador. Se trata de una disciplina que ha contribuido de manera determinante al desarrollo de la lingüística computacional. Es seguramente también una de las aplicaciones informáticas que mayores recursos humanos y económicos ha recibido. El mercado ofrece en la actualidad un amplio abanico de productos y es difícil para el profano elegir el más adecuado para sus necesidades. Con todo, es importante saber que un texto producido por un sistema de traducción automática debe ser revisado con cuidado antes de darlo por válido y publicarlo.

2.5 Lexicografía

La lexicografía se ocupa de la representación del vocabulario de una lengua natural o de un sector de ella (un dialecto, sociolecto, etc.); se trata por tanto de una rama de la lexicología (ciencia que estudia el vocabulario de una lengua, su estructura, composición y variación), que privilegia los aspectos aplicados –la composición de diccionarios-, sin que esto signifique que no le competa la reflexión acerca de los problemas teóricos que conlleva esa labor. Por el contrario, se suele definir la lexicografía como "la teoría de *la descripción de diccionarios" y la "codificación de la estructura paradigmática y sintagmática del léxico de una lengua, la transmisión ordenada de información léxica (y gramatical) en forma de diccionario" (Lewandowski, 1992).*

2.2.2 Búsqueda de patrones

1 Definición

La búsqueda de patrones (lexicografía) permite reconocer entre una serie de caracteres el tipo y funcionalidad de ciertos patrones. Por ejemplo reconocer si una palabra empieza por un dígito indica que posiblemente es un número constante (o un error), si no es así posiblemente la palabra es un identificador que define algún valor u operador que altera un valor.

2 Terminología básica de búsqueda de patrones

2.1 Búsqueda monopatrón

El problema de búsqueda de patrones trata de encontrar todas las coincidencias de un patrón *P = p1…pn* en el texto *T = t1…tn*, donde tanto P y T son secuencias de caracteres sobre un alfabeto finito Σ.

Los algoritmos de búsqueda monopatrón se pueden clasificar de acuerdo a la forma en que buscan el patrón en el texto. Todos ellos utilizan una ventana de búsqueda del tamaño del patrón, la cual se desliza de izquierda a derecha a lo largo del texto, y dentro de la cual se busca el patrón, el esquema general se muestra en la figura 2.4.

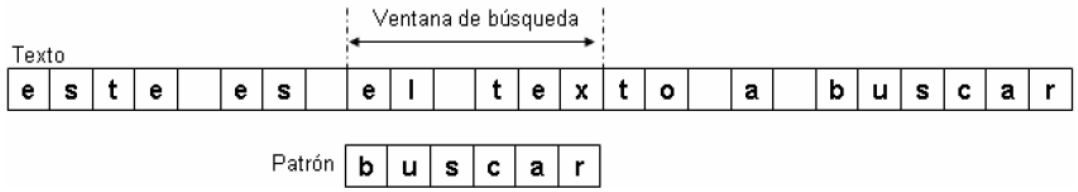


Figura 2.4 Búsqueda monopatrón: La búsqueda se realiza en una ventana que se desplaza a lo largo del texto. La ventana de búsqueda es del tamaño del patrón.

La diferencia entre los algoritmos radica en la forma en que la ventana se desplaza. Dada una cadena *x, y*, y *z*, se dice que *x* es prefijo de *xy*, un sufijo de *yx*, y un factor de *yxz*. En base a esto último la clasificación de los algoritmos de búsqueda de monopatrón es la siguiente:

Búsqueda por prefijo (Figura 2.5). La búsqueda se hace hacia delante en la ventana de búsqueda, leyendo todos los caracteres del texto uno después de otro. En cada posición de la ventana, se busca el prefijo más largo de la ventana que también es un prefijo del patrón. Entre los algoritmos más conocidos que utilizan este método están el algoritmo de Knuth-Morris-Pratt (KMP) , Shift-And y Shift-Or [11].

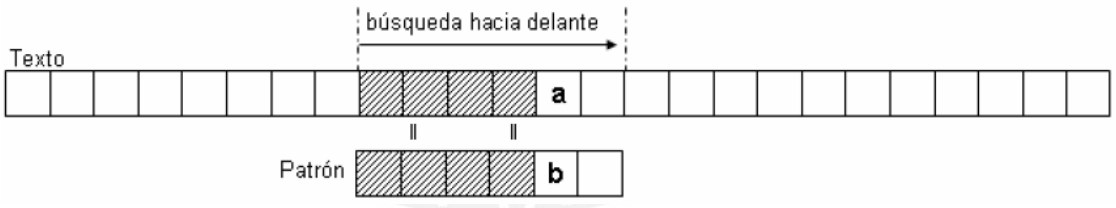


Figura 2.5. Búsqueda monopatrón por prefijo: Se busca el prefijo más largo del patrón en la ventana de búsqueda.

Búsqueda por sufijo *(*Figura 2.6*).* La búsqueda se hace hacia atrás a lo largo de la ventana de búsqueda, se lee el sufijo más largo de la ventana que también es un sufijo del patrón. Este método permite evitar leer algunos caracteres del texto.

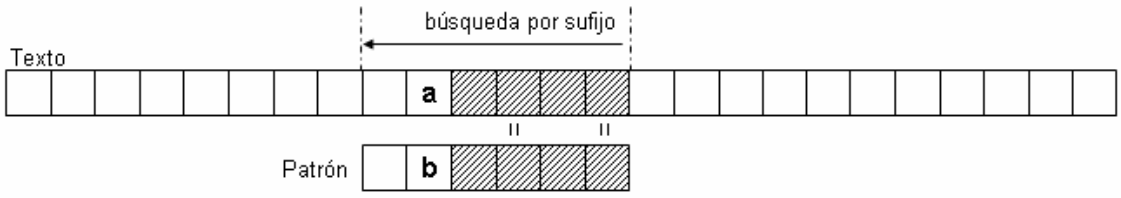


Figura 2.6. Búsqueda monopatrón por sufijo: Se busca un sufijo del patrón en la ventana de búsqueda.

Búsqueda por Factor (figura 2.7*)*. La búsqueda se hace hacia atrás en la ventana de búsqueda, buscando el sufijo más largo de la ventana que también es un factor del patrón. La principal desventaja de este método es que requiere una forma para reconocer el conjunto de factores del patrón, lo cual es bastante complejo.

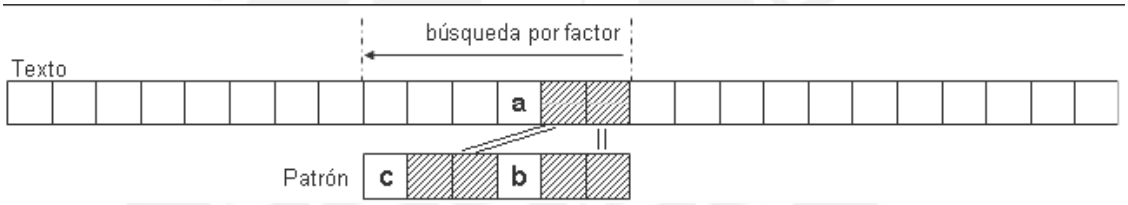


Figura 2.7 Búsqueda monopatrón por factor: Se busca un factor del patrón en la ventana de búsqueda.

Estas tres formas para buscar un patrón permiten a los algoritmos ser eficientes en distintos casos, dependiendo del tamaño del patrón y el tamaño del alfabeto.

2.2 Búsqueda multipatrón

El problema de búsqueda de patrones se puede extender para buscar un conjunto de patrones simultáneamente $P = \{p^1, p^2, ..., p^r\}$, donde cada p^i es una cadena $p^i = p^i_1 p^i_2 ... p^i_{mi}$ sobre un conjunto de caracteres finitos Σ. La búsqueda se hace sobre un texto $T = t_1 ... t_u$. El objetivo es encontrar todos los pares i,j tal que $t_{j-|pi|+1} ... t$ sea igual a p^i [11].

La solución más simple a este problema es realizar r búsquedas con alguno de los algoritmos clásicos de búsqueda monopatrón. Esto lleva a una

complejidad en el peor de los casos de O(*ru*). La complejidad de la fase de búsqueda se puede reducir a O(*u*+*nocc*)(donde *nocc* es el número total de coincidencias) usando algún tipo de extensión de los algoritmos de búsqueda monopatrón.

Los tres esquemas para búsqueda monopatrón se aplican para búsqueda multipatrón. Para cada esquema existen diversas extensiones posibles de acuerdo a la forma en que se manipula el conjunto de patrones y a cómo se obtienen los desplazamientos.

2.3 Búsqueda aproximada

Una generalización del problema de búsqueda de patrones es la búsqueda aproximada de patrones o también conocida como búsqueda permitiendo errores, en la cual se tiene un patrón P y un margen de error k. Este problema tiene sentido sólo si $0<k<m$, puesto que en otro caso cualquier subcadena de longitud m puede ser convertida a P sustituyendo los m caracteres [2]. Cuando $k = 0$ corresponde a una búsqueda exacta monopatrón. El nivel de error $\alpha = k/m$ nos da una medida de la fracción del patrón que puede ser alterado.

2.4 Algoritmos de paralelismo de bit

El paralelismo de bits se ha usado extensamente para búsqueda aproximada, teniendo sus mejores resultados para patrones cortos, que en muchos casos son los patrones de interés. La técnica de paralelismo de bits aprovecha la ventaja del paralelismo de bits intrínseco en las operaciones de bits de una palabra de computadora [13]. Es decir, se pueden empaquetar muchos valores en una palabra de computadora y actualizar todos éstos con una sola operación. Tomando ventaja del paralelismo de bits, el número de operaciones que un algoritmo desempeña se puede reducir por un factor de *w*, donde *w* es el número de bits de una palabra de computadora. En las arquitecturas actuales *w* es igual a 32 ó 64, el incremento de velocidad es muy significante en la práctica.

Algunas notaciones que se utilizan para describir los algoritmos de paralelismo de bits son las siguientes: se usa exponentes para expresar repeticiones de bits, por ejemplo, $0^3 1 = 0001$. Una secuencia de bits $b_1 \ldots b_l$, se conoce como máscara de bits de longitud l, la cual se almacena dentro de la palabra de computadora de longitud w. Generalmente se utiliza la sintaxis del lenguaje C para las operaciones sobre los bits, es decir, "|" es una operación OR, "&" es una operación AND, " ^ " es una operación XOR, " ~ " complementa todos los bits, y "<<" (">>") mueve los bits a la izquierda (derecha) e ingresa ceros a partir de la derecha (izquierda), por ejemplo, $b_l b_{l-1} \ldots b_2 b_1 << 3 = b_{l-3} \ldots . b_2 b_1 000$ [6].

Los algoritmos de paralelismo de bits simulan los algoritmos clásicos. Algunos paralelizan el cálculo de la matriz de programación dinámica y algunos paralelizan el cálculo del autómata finito no determinístico (NFA). La técnica más simple [13], empaqueta cada fila i del NFA en diferentes palabras de computadora R_i, cada estado está representado por un bit. Cada que se lee un carácter del texto, todas las transiciones del autómata se simulan usando operaciones de bits entre las $k+1$ máscaras de bits, las cuales tienen la misma estructura, es decir, el mismo bit está alineado a la misma posición del texto. Para actualizar los valores de R'_i en la posición del texto j teniendo los valores actuales R_i se aplica la siguiente fórmula:

$$R'_0 ((R_0 << 1) \mid 0^{m-1} 1) \& B[t_j]$$

$$R'_i ((R_i << 1) \& B[t_j] \mid R_{i-1} \mid (R_{i-1} << 1) \mid (R'_{i-1} << 1) \quad (1)$$

Donde B es una tabla que almacena una máscara de bits $b_m \ldots b_1$ para cada carácter del patrón. La máscara en $B[c]$ tiene el j^{th} bit activo si $P_j = c$. La búsqueda se inicia con $R_i = 0^{m-i} 1^i$.

2.3 Modelo Teórico

La tecnología necesaria para navegar desde un extremo de la cadena de digitalización al otro consta principalmente de: hardware, software y redes. Éstos son el centro de esta sección. Una perspectiva integral de la infraestructura técnica también incluye protocolos y normas, políticas y

procedimientos (para el flujo de trabajo, mantenimiento, seguridad, actualizaciones, etc.) y los niveles de habilidad y responsabilidades del trabajo del personal de una organización.

De esto se concluye que la cadena de digitalización y la infraestructura técnica que la sostiene se dividen en tres componentes fundamentales: creación, gestión y entrega.

- La Creación de imágenes se ocupa de la captura o conversión inicial de un documento u objeto a la forma digital, por lo general con un escáner o cámara digital. A la imagen inicial se le pueden aplicar uno o más pasos de procesamiento de archivo o de imagen, que pueden alterar, agregar o extraer datos. Las clases generales de procesamiento incluyen la edición de la imagen (escalarla, comprimirla, otorgarle nitidez, etc.).
- La Gestión de archivos se refiere a la organización, almacenamiento y mantenimiento de imágenes.
- La Entrega de la imagen comprende el proceso de hacer llegar las imágenes al usuario y abarca redes, dispositivos de visualización e impresoras.

Las computadoras y sus interconexiones de red son componentes integrales de la cadena de digitalización. Cada eslabón de la cadena comprende una o más computadoras y sus diversos componentes (RAM, CPU, bus interno, tarjetas de expansión, soporte de periféricos, dispositivos de almacenamiento y soporte de red). Los requisitos de configuración cambiarán, dependiendo de las necesidades informáticas específicas de cada componente.
A continuación se muestra el modelo teórico:

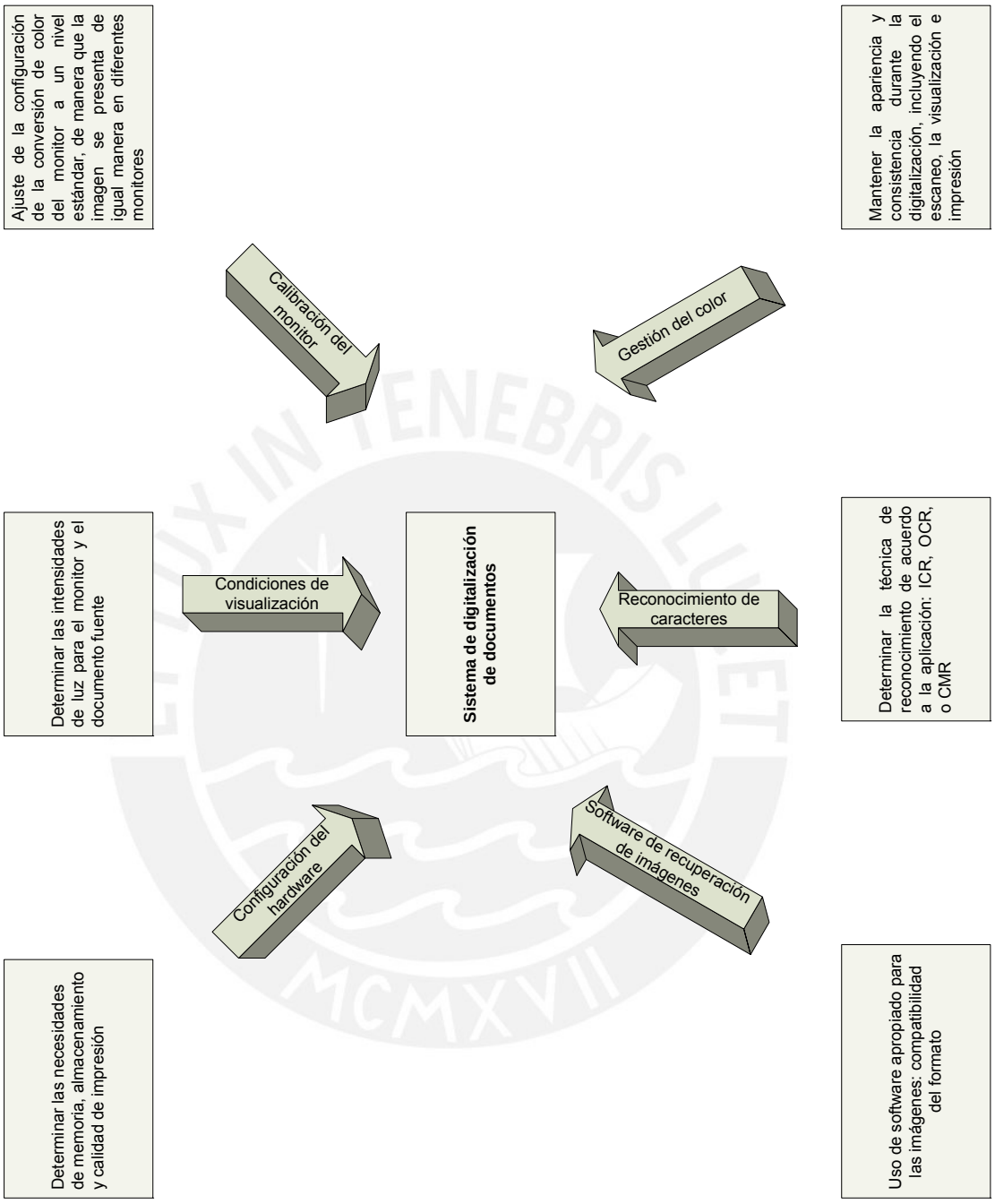


Figura 2.8 Diagrama del Modelo Teórico

2.4 Conclusiones

Los sistemas de digitalización modernos, tales como los basados en OCR, se presentan como buenas alternativas actuales para automatizar el proceso de digitalización de los documentos almacenados en las instituciones locales.

Para lograr un eficiente sistema de digitalización se debe usar un adecuado OCR que este acompañado de un sistema basado en lingüística computacional que verifique las salidas dadas por el OCR; es decir, por ejemplo, que verifique que la palabra reconocida exista en el diccionario de la Lengua Española.

CAPITULO 3:

DISEÑO E IMPLEMENTACIÓN DEL SISTEMA DE CORRECCIÓN ORTOGRÁFICA

3.1 Análisis e Instrumentos de cálculo

3.1.1 Hipótesis principal

Dado que el rendimiento de los sistemas de reconocimiento de caracteres es bajo cuando se trata de digitalizar documentos deteriorados debido a las manchas y otros factores que evitan que se reconozcan las palabras del texto original; entonces, la implantación de un sistema de corrección ortográfica a la salida del OCR permitirá mejorar la eficiencia del OCR al momento del reconocimiento de los caracteres. De esta manera, la digitalización de los documentos históricos podrá garantizar una calidad óptima.

3.1.2 Objetivos de la tesis

3.1.2.1 Objetivo general

Diseñar e implementar un sistema de corrección ortográfica que use el principio de búsqueda de patrones para incrementar el rendimiento de un OCR, cuando sea usado para digitalizar libros actuales no muy deteriorados.

Como resultado de este sistema se obtendrá una imagen lo más parecida al documento original. Además, el documento digitalizado se almacenará en un formato editable para que el usuario final pueda realizar ciertas correcciones, añadir algún dato, o simplemente guardarlo en un dispositivo de almacenamiento.

3.1.2.2 Objetivos específicos

Los objetivos específicos de la presente tesis son los siguientes:

1) Diseñar una técnica adecuada para el análisis de los datos proporcionados por el OCR (Red Neuronal).

2) Lograr un sistema en el cual el tiempo requerido para el análisis de cada palabra sea el mínimo.

3) Entregar los datos procesados en un archivo editable tal como los documentos Word.

4) Minimizar los requerimientos de software para implementar el sistema propuesto.

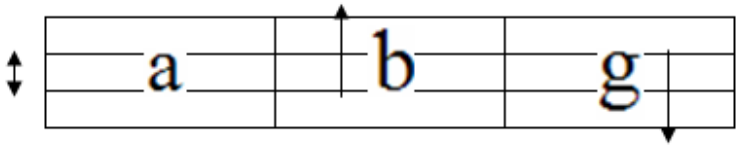
5) Diseñar un eficiente algoritmo para la búsqueda de palabras en la base de datos de Lengua Española.

3.2 Consideraciones preliminares para el diseño del sistema

Para el desarrollo del sistema de corrección se han considerado los siguientes datos preliminares:

- El tipo del carácter

El tipo del carácter se refiere a la codificación que se ha hecho a las letras del alfabeto de acuerdo a la forma la misma, esto es:



Los caracteres tipo 1 serán aquellos que se encuentren en el rango del tamaño de la letra "*a*"; los caracteres tipo 2 serán aquellos que tengan una parte sobresaliente para arriba como se muestra para la letra "*b*"; los caracteres tipo 3 serán aquellos que tengan una parte sobresaliente hacia abajo como se muestra para la letra "*g*"; y finalmente, los caracteres tipo 4 serán aquellos símbolos que representan los puntos, tildes, diéresis y las vocales con tilde (á, é, í, ó, ú). De acuerdo a esta clasificación, se tienen los siguientes resultados:

Caracteres tipo 1:

a c e i m n o r s u v w x z

Caracteres tipo 2:

b d f h k l t - A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Caracteres tipo 3:

g j p q y

Caracteres tipo 4:

. : ´ á é í ó ú

Esta clasificación de los caracteres será una información muy útil al momento de hacer la corrección del OCR. Los tipos 1, 2, 3 corresponden a los caracteres que se encuentran en sus respectivos recuadros, mientras que el tipo 4 corresponde a los caracteres especiales de la codificación

ASCII y en esta categoría tenemos: á, é, í, ó, ú, ñ, Ñ y otros caracteres que contienen los símbolos mostrados en el recuadro.

- Número de caracteres que conforman la palabra

  Un dato muy importante dentro del sistema diseñado es el número de caracteres que conforman la palabra que se está reconociendo. Esta información es proporcionada por la etapa de pre-procesamiento del texto que se esta digitalizando.

- Archivo de texto (*.txt) con la información de los caracteres

  El archivo con los resultados de la red neuronal contiene los caracteres de las palabras que se están reconociendo mediante un OCR, este documento tendrá la siguiente estructura: Un arreglo, donde cada fila es información que corresponde a un caracter: la primera celda contiene el ASCII del caracter reconocido; la siguiente, la "probabilidad" dada por el OCR; la tercera celda el segundo ASCII más cercano; la cuarta celda su "probabilidad" respectiva y así sucesivamente hasta completar 10 celdas. Esta información corresponde a 5 caracteres en orden de importancia.

- Texto original que se está digitalizando

  Finalmente, se contará con el texto original se que esta digitalizando para verificar el correcto funcionamiento del sistema corrector ortográfico.

3.3 Estudio de las etapas que se requieren para implementar el sistema de corrección

3.3.1 OCR

Se encarga del reconocimiento de los caracteres del texto a digitalizar, para ello usa una red neuronal que entrega la probabilidad de reconocimiento de los cinco caracteres que más se asemejan al caracter que se esta analizando.

Por ejemplo, si el OCR quiere reconocer la letra “o” se genera una tabla que contiene todas la letras del alfabeto tanto mayúsculas como minúsculas y a cada una de ellas le otorgará una probabilidad de reconocimiento que dependerá de cuánto se parece la letra del alfabeto a la letra que se está reconociendo. Esta probabilidad varía desde 0.0 a 1.0, lo cual se puede interpretar en términos de porcentajes como una variación de 0% que corresponde a 0.0 y 100% que corresponde a 1.0; para el caso de la letra “o” el OCR puede dar como resultado lo siguiente:

111	0.98352137	68	0.06634963	99	0.00067212	101	0.00016528	106	1.4329E-05

Esta tabla se interpreta de la siguiente manera:

- Los números 111, 68, 99, 101, 106 representan los códigos ASCII de las letras del alfabeto. El sistema al leer esta información buscará en una tabla a qué letra del alfabeto corresponde cada código ASCII. Para el presente ejemplo, el primero código ASCII es el 111, entonces usando la tabla de códigos ASCII de la figura 3.1 se puede apreciar que este código corresponde a la letra “o“. De la misma forma se procede con el análisis de los códigos 68, 99, 101 y 106 los siguientes corresponden a los siguientes caracteres respectivamente: D c e j.

- Los números que se encuentran junto a los códigos ASCII representan las probabilidades de reconocimientos otorgadas por el OCR. Por ejemplo, el código ASCII 111, que corresponde a la letra “o”, tiene asignado el número 0.98352137. El sistema leerá esta información y lo expresará en porcentaje, en donde 1.0 representa el 100%. Para el ejemplo (código ASCII 111), el sistema sabrá que éste código tiene una probabilidad de reconocimiento de 98.352137%, el cual se puede aproximar a un 98.35%.

Carácteres no imprimibles				Carácteres imprimibles								
Nombre	Dec	Hex	Car.	Dec	Hex	Car.	Dec	Hex	Car.	Dec	Hex	Car.
Nulo	0	00	NUL	32	20	Espacio	64	40	@	96	60	`
Inicio de cabecera	1	01	SOH	33	21	!	65	41	A	97	61	a
Inicio de texto	2	02	STX	34	22	"	66	42	B	98	62	b
Fin de texto	3	03	ETX	35	23	#	67	43	C	99	63	c
Fin de transmisión	4	04	EOT	36	24	$	68	44	D	100	64	d
enquiry	5	05	ENQ	37	25	%	69	45	E	101	65	e
acknowledge	6	06	ACK	38	26	&	70	46	F	102	66	f
Campanilla (beep)	7	07	BEL	39	27	'	71	47	G	103	67	g
backspace	8	08	BS	40	28	(	72	48	H	104	68	h
Tabulador horizontal	9	09	HT	41	29	)	73	49	I	105	69	i
Salto de línea	10	0A	LF	42	2A	*	74	4A	J	106	6A	j
Tabulador vertical	11	0B	VT	43	2B	+	75	4B	K	107	6B	k
Salto de página	12	0C	FF	44	2C	,	76	4C	L	108	6C	l
Retorno de carro	13	0D	CR	45	2D	-	77	4D	M	109	6D	m
Shift fuera	14	0E	SO	46	2E	.	78	4E	N	110	6E	n
Shift dentro	15	0F	SI	47	2F	/	79	4F	O	111	6F	o
Escape línea de datos	16	10	DLE	48	30	0	80	50	P	112	70	p
Control dispositivo 1	17	11	DC1	49	31	1	81	51	Q	113	71	q
Control dispositivo 2	18	12	DC2	50	32	2	82	52	R	114	72	r
Control dispositivo 3	19	13	DC3	51	33	3	83	53	S	115	73	s
Control dispositivo 4	20	14	DC4	52	34	4	84	54	T	116	74	t
neg acknowledge	21	15	NAK	53	35	5	85	55	U	117	75	u
Sincronismo	22	16	SYN	54	36	6	86	56	V	118	76	v
Fin bloque transmitido	23	17	ETB	55	37	7	87	57	W	119	77	w
Cancelar	24	18	CAN	56	38	8	88	58	X	120	78	x
Fin medio	25	19	EM	57	39	9	89	59	Y	121	79	y
Sustituto	26	1A	SUB	58	3A	:	90	5A	Z	122	7A	z
Escape	27	1B	ESC	59	3B	;	91	5B	[	123	7B	{
Separador archivos	28	1C	FS	60	3C	<	92	5C	\	124	7C	|
Separador grupos	29	1D	GS	61	3D	=	93	5D	]	125	7D	}
Separador registros	30	1E	RS	62	3E	>	94	5E	^	126	7E	~
Separador unidades	31	1F	US	63	3F	?	95	5F	_	127	7F	DEL

Figura 3.1 Tabla de Códigos ASCII

3.3.2 Entrega de datos en un archivo *.txt

Este archivo contiene toda la información que el corrector ortográfico necesita para hacer el análisis correspondiente a los caracteres, esta información contiene el código ASCII y el respectivo porcentaje de reconocimiento de los cinco primeros caracteres. A continuación se muestra un ejemplo de este archivo:

sample - Notepad

File Edit Format View Help

68	0.896224535	111	0.008098567	106	0.000350097	110	3.64074E-07	115	3.54889E-07
101	0.998294623	99	0.00070102	117	1.96851E-05	68	1.62244E-06	116	3.56301E-07
99	0.956481034	101	0.038795035	111	0.003427917	117	0.002966899	122	0.002067434
105	0.999494917	108	0.002871474	116	2.02777E-10	114	0	101	0
100	0.999332463	110	7.16337E-05	68	3.83532E-05	117	2.38453E-05	111	1.86883E-07
105	0.999947986	108	0.01985402	116	3.755E-10	114	2.4E-14	109	0
114	0.999881467	116	2.85996E-06	122	1.82603E-06	117	6.74396E-09	118	3.78871E-09
101	0.997940149	99	0.001607468	117	2.11294E-05	68	2.18344E-06	111	1.63444E-07
115	0.999944557	117	1.54002E-06	110	8.47365E-07	106	2.95118E-07	116	5.66959E-08
108	0.983863608	105	2.59698E-05	116	1.05579E-05	118	1.95968E-08	115	1E-15
100	0.999635775	68	0.000117003	117	4.49408E-05	99	7.41232E-07	111	5.11184E-07
101	0.998164528	99	0.002153928	117	0.000102422	68	2.66966E-06	111	1.32392E-06
106	0.999946344	110	1.36271E-05	111	8.3484E-06	100	3.82594E-06	115	2.82801E-06
101	0.998265711	99	0.001317514	117	1.13755E-05	68	3.12941E-06	116	2.28539E-07
114	0.999971802	122	4.97116E-06	116	9.74715E-07	117	3.44014E-09	115	1.19739E-09
99	0.975214911	101	0.025884992	111	0.004973823	68	0.001949977	117	0.001447452
101	0.998068274	99	0.002387841	117	3.1553E-05	68	1.79437E-06	100	1.24268E-06
114	0.999943079	122	1.04212E-05	116	5.51721E-06	110	3.31551E-08	117	1.52795E-08
110	0.99143906	117	0.002092758	68	3.92648E-05	100	2.50365E-05	106	8.18789E-06
117	0.995486843	110	0.000892035	111	0.000415173	114	9.26678E-05	104	9.39588E-06
101	0.999342126	99	8.7938E-05	117	1.40255E-05	116	2.65877E-07	111	8.91218E-10
115	0.999988897	117	7.10464E-06	106	6.0742E-06	110	1.5646E-06	116	3.10173E-08
116	0.982971982	108	0.045311712	118	0.007712423	105	0.000605313	114	3.89527E-05
114	0.999982558	122	2.13244E-06	116	8.40364E-07	117	3.5889E-09	115	1.11239E-09
111	0.983521367	68	0.06634963	99	0.000672125	101	0.000165283	106	1.43295E-05

Figura 3.2 Archivo con la información dada por la red neuronal

3.3.3 Análisis de caracteres

Este análisis se realiza en base a las probabilidades otorgadas por el OCR, el cual consiste en determinar que caracteres del alfabeto corresponden a los 5 primeros caracteres que tienen la posibilidad de ser la letra del texto original.

Por ejemplo, La palabra que se quiere reconocer es:

Comprometer

Por otro lado, se sabe que por cada caracter que reconoce el OCR entrega una probabilidad a cada letra del alfabeto de acuerdo al parecido que ésta tiene con la letra original. Una vez asignada la probabilidad de cada letra, lo que se hace es escoger a los cinco mayores y se analiza si la letra de mayor probabilidad pasa el umbral de reconocimiento, el cual es la probabilidad mínima para afirmar con total seguridad que se trata de la letra del texto original, si ese es el caso entonces solo se considera esa letra para el análisis ortográfico, pero si

ninguna letra pasa el umbral de reconocimiento lo que se hace es considerar a las cinco letras con mayor probabilidad.

Para la palabra del ejemplo la salida que se tiene luego de hacer el análisis es el siguiente:

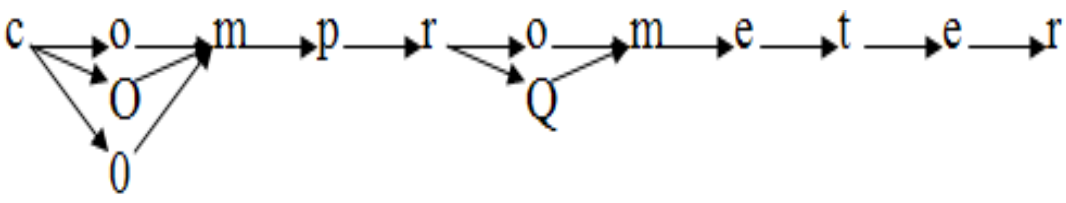
c o m p r o m e t e r

c	90%	o	87%	m	92%	p	95%	r	90%	o	88%	m	92%	e	90%	t	93%	e	92%	r	95%
o	15%	O	80%	n	10%	P	14%			Q		n	20%			1	18%	o	25%		
		0	78%																		

En este ejemplo se puede observar que el OCR ha reconocido con total seguridad nueve letras de la palabra en cuestión. En el caso de la primera palabra “o” no se puede reconocer completamente, pero se ha escogido a las tres primeras letras que más se parecen a la letra original, lo mismo sucede para la segunda letra “o”. Con este resultado se hará el análisis ortográfico.

3.3.4 Combinación de caracteres

Esta etapa se encarga de obtener todas las posibles palabras que se pueden obtener de la combinación de los caracteres que han sido seleccionados en la etapa anterior. Por ejemplo, para la palabra que se esta analizando “*comprometer* ” se harían las siguientes combinaciones:



Como resultado de estas combinaciones se obtiene las siguientes palabras:

comprometer

comprQmeter

cOmprometer

cOmprQmeter

c0mprometer

c0mprQmeter

Otra tarea de esta es obtener un porcentaje global de la palabra que se forma de la combinación de las letras y esto se obtiene calculando el promedio ponderado de los porcentajes de cada una de las letras, esto es:

$$\frac{\text{Letra1} + \text{letra2} + \text{letra3} \ldots\ldots + \text{letran}}{((P1) + (P2) + (P3) + \ldots + (Pn))/n} = PG$$

Donde:

P1, P2, P3, Pn: son los porcentajes de sus letras correspondientes.

PG : porcentaje global de la palabra.

Finalmente, el resultado que se obtiene para la palabra en estudio como resultado de esta etapa es la siguiente:

Palabra	Porcentaje Global
comprometer	90.5
comprQmeter	89.5
cOmprometer	88.5
cOmprQmeter	86.75
c0mprometer	76.75
c0mprQmeter	75.5

3.3.5 Análisis Ortográfico: Bases de datos

En esta etapa se analizará cada una de las palabras generadas en la etapa anterior y dependiendo de si existen o no en el diccionario se toman las siguientes decisiones:

- Si la palabra no existe en el diccionario se buscarán las más parecidas y serán etiquetadas con una prioridad de 1 y luego de esto se calculará su error de semejanza.

- Si la palabra existe en el diccionario será etiquetada con una prioridad de 2 y también se calculará su error de semejanza

A continuación de muestra el cálculo del error de semejanza:

Error de Semejanza	=	100	-	Porcentaje global de la palabra

Para buscar la palabra dentro del diccionario se utilizará el procesamiento de paralelismo de bits. A continuación se muestra el diagrama de funcionamiento de esta etapa.

3.3.6 Selección de la palabra correcta

En esta etapa del sistema se seleccionará la palabra que más se asemeja a la palabra original para ello se seguirá el siguiente procedimiento:

1. Se leerá el archivo que contiene las palabras seleccionadas en la etapa anterior

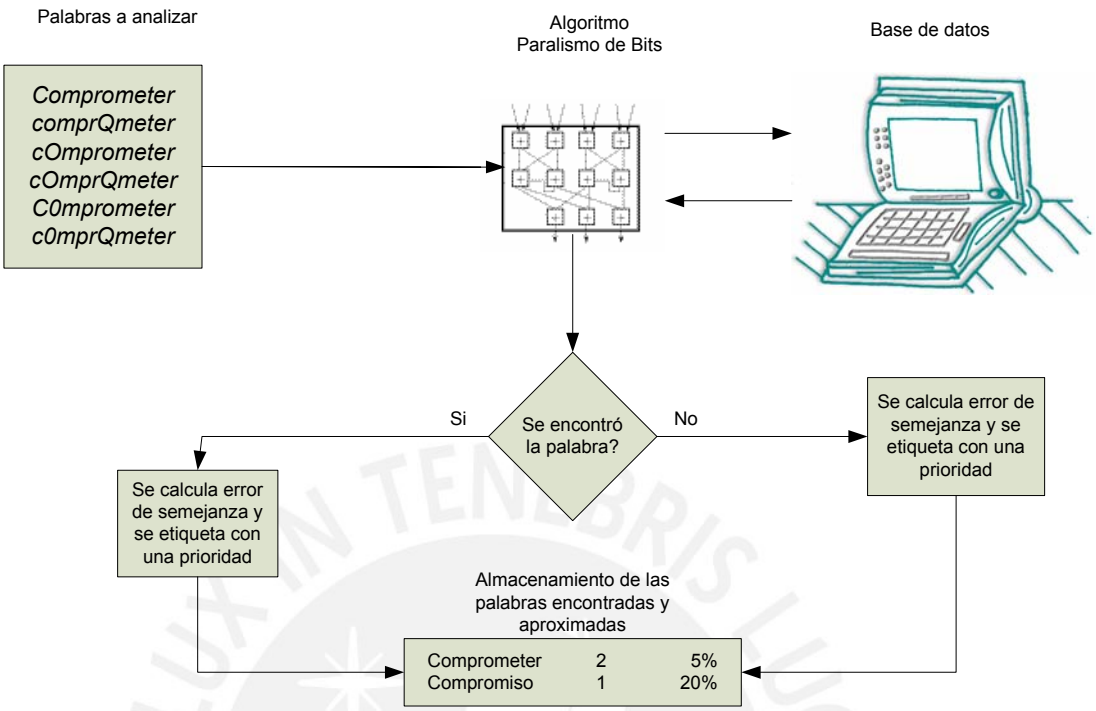


Figura 3.3 Diagrama del Sistema de Corrección

2. Se ordenarán las palabras primero de acuerdo a su prioridad y luego de acuerdo su error de semejanza.
3. Finalmente, se escogerá aquella palabra que posea la más alta prioridad y menor error de semejanza.

El siguiente diagrama muestra como se realiza este proceso:

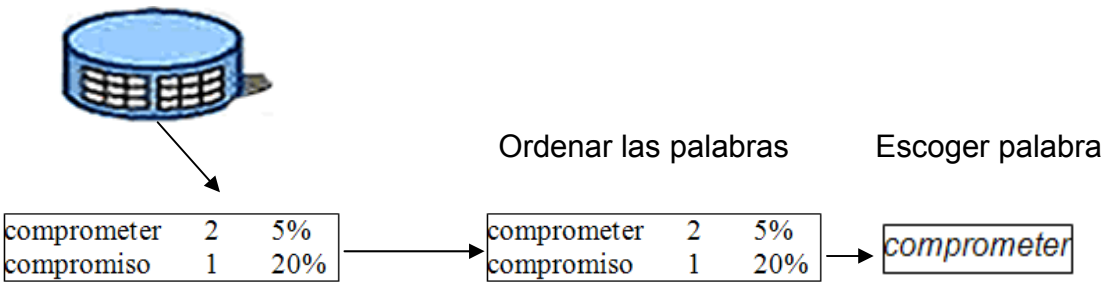


Figura 3.4 Diagrama para seleccionar una palabra

3.3.7 Entrega en un archivo editable

Esta etapa se encarga de entregar las palabras seleccionas en la etapa anterior en un formato editable, para el presenta sistema la información se entregará en un archivo .txt. El siguiente diagrama:

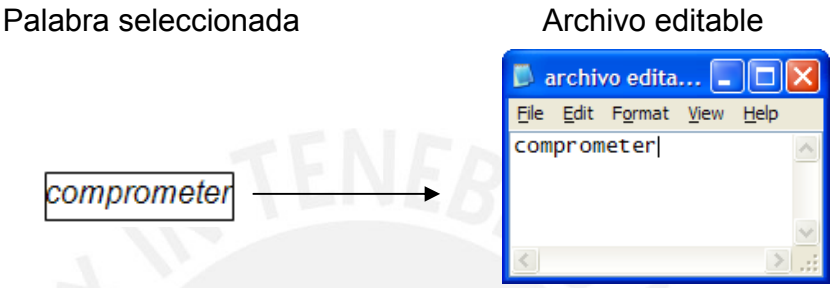


Figura 3.5 Entrega de la palabra en archivo editable

3.4 Implementación de las etapas del sistema de corrección

Para realizar la corrección de los datos generados por el OCR, se desarrolló un programa que recibe como entrada el archivo tipo .TXT, a partir del cual se extrae toda la información que corresponde a los caracteres del texto que se esta digitalizando. Este programa se desarrolló en Visual C++, debido a que mediante este lenguaje de programación se logra implementar eficientes algoritmos de búsqueda, los cuales son vitales para el funcionamiento del sistema.

El sistema implementado y su esquema de funcionamiento se muestran en la Figura 3.6.

El sistema implementado cuenta como parámetro de entrada el archivo tipo .txt, el programa análisis se encargará de leerlo y guardarlo en un formato adecuado para su procesamiento, el cual estará basado en arreglos; una vez almacenado

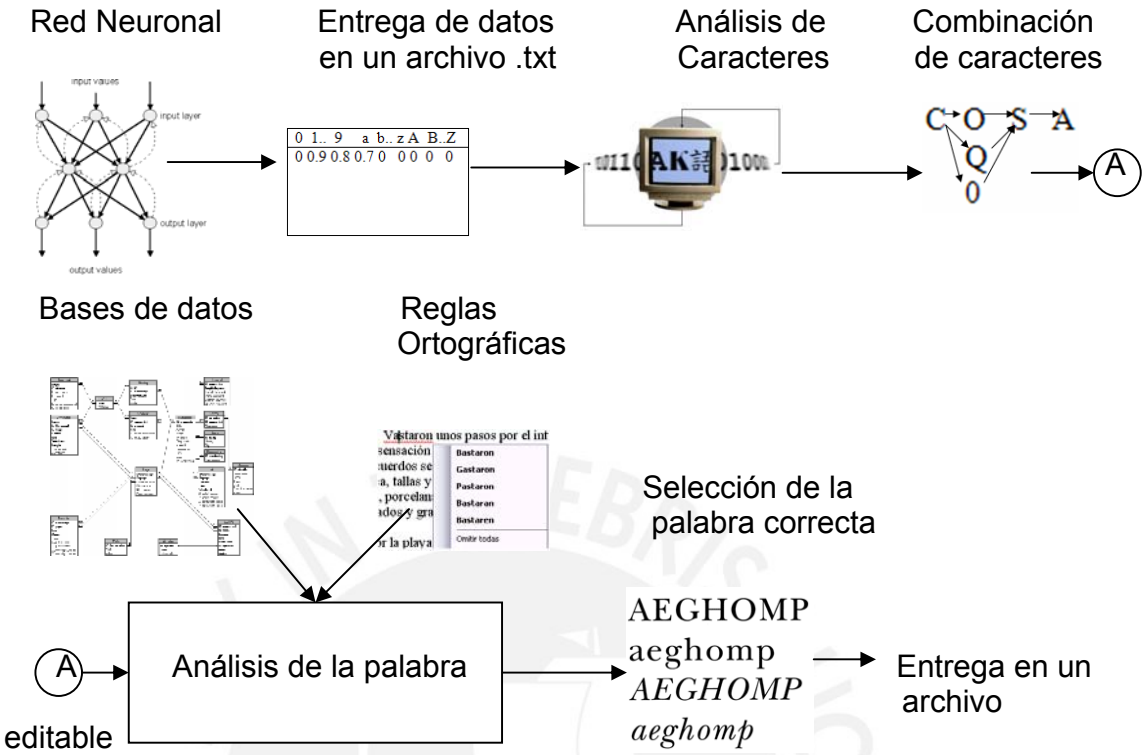


Figura 3.6 Esquema de funcionamiento del sistema implementado

en memoria se procederá a un análisis de la información de cada carácter para poder descartar caracteres que no son parte de la palabra original del texto que se esta digitalizando. Luego de esta etapa se procederá a una combinación de todas la posibles palabras que se pueden generar con los caracteres que han pasado en la primera etapa, para cada combinación de palabras se procederá a un análisis de corrección que se basará en el uso de una base de datos que contiene todo el corpus de la Lengua Española. Finalmente, se seleccionará la palabra que más se asemeja a la palabra del texto procesado.

En la figura 3.7 se muestra el diagrama de flujo del sistema implementado:

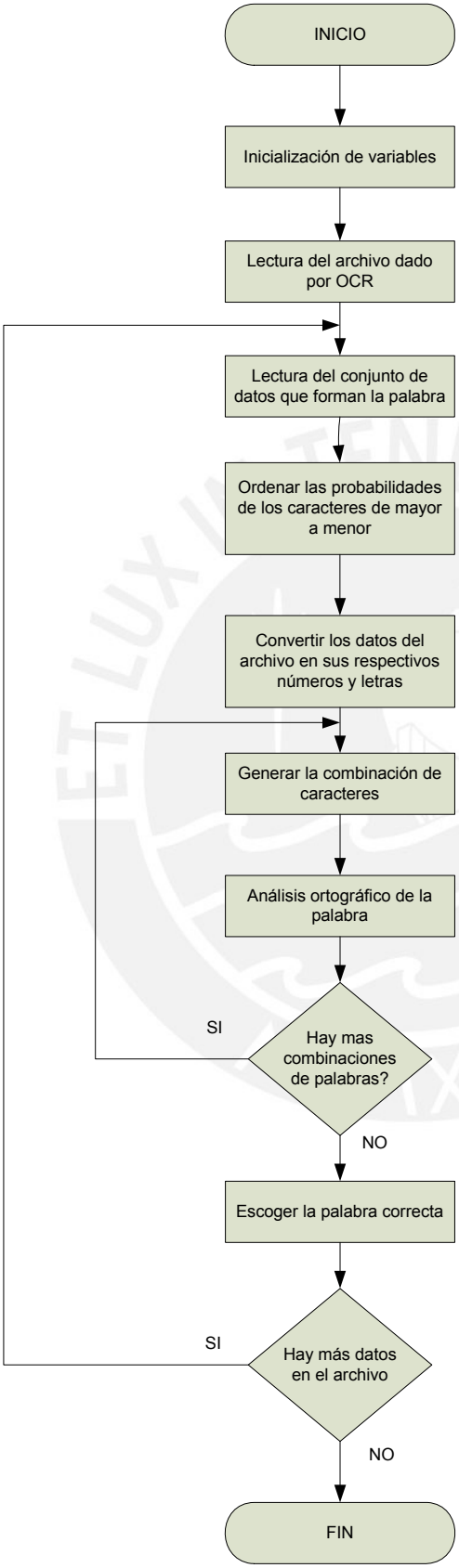


Figura 3.7 Diagrama de flujo del sistema implementado

A continuación se muestra el diagrama de flujo de las funciones usadas por el programa principal:

Función que ordena las probabilidades de los caracteres de mayor a menor:

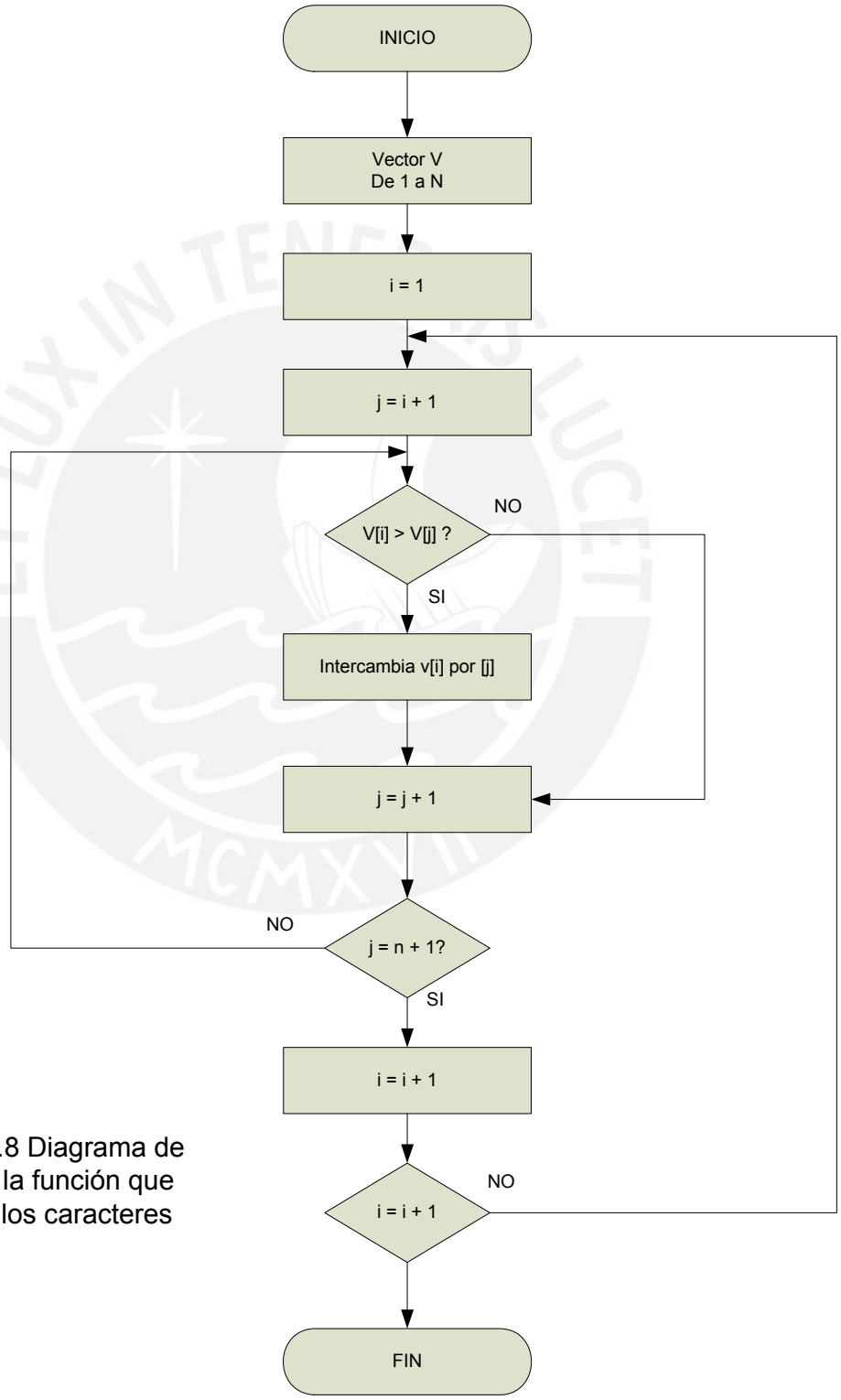


Figura 3.8 Diagrama de flujo de la función que ordena los caracteres

La lógica del diagrama anterior es el siguiente: el algoritmo recibe en un arreglo de datos las probabilidades de reconocimiento de los caracteres reconocidos por el software OCR, luego de leerlos procederá a ordenarlos en forma descendente, de tal manera que se obtenga otro arreglo de datos con las probabilidades ordenadas. Este orden descendente es importante porque aquí se seleccionan los cinco caracteres con mayor probabilidad de reconocimiento.

A continuación se muestra el diagrama de flujo de la función que se encarga de la combinación de caracteres:

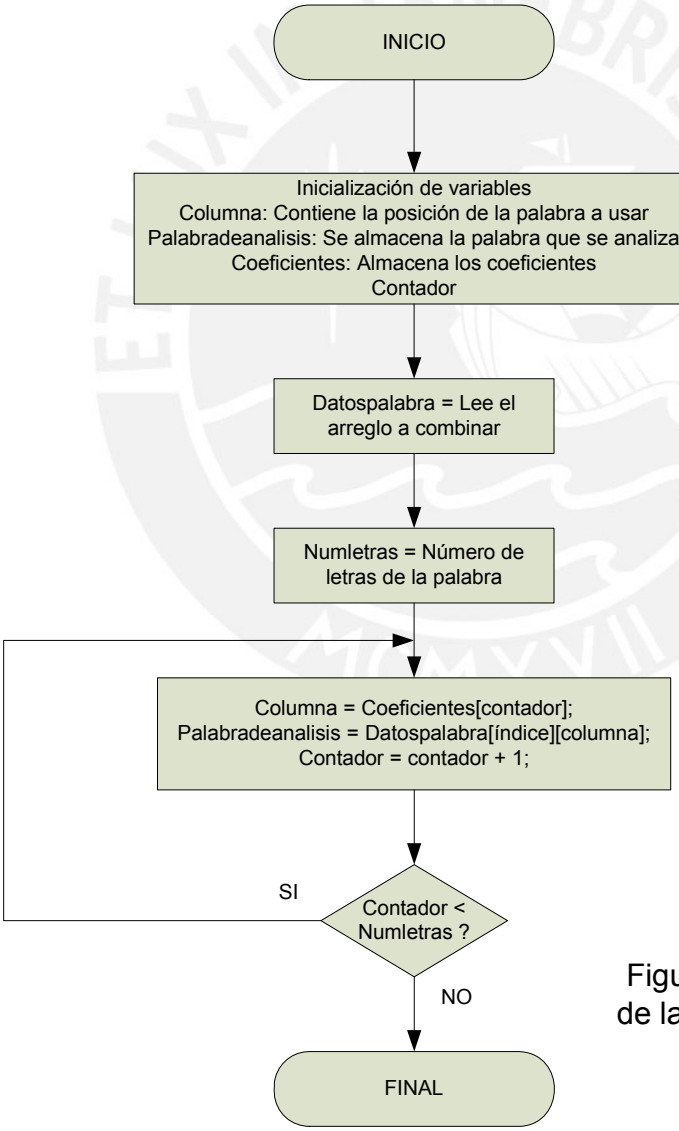


Figura 3.9 Diagrama de flujo de la función que combina los caracteres

3.5 Casos analizados por el sistema de corrección

A continuación se describen los 4 casos que analizará el sistema de corrección ortográfica:

Tipo de análisis	Condiciones de la palabra a analizar	Resultados del análisis ortográfico
Caso 1	Tienes todas sus letras completas	Hay dos o más palabras correctas
Caso 2	Tienes todas sus letras completas	Hay una palabra correcta
Caso 3	No tiene todas sus letras completas	Hay una palabra correcta
Caso 4	No tiene todas sus letras completas	Hay dos o más palabras correctas

3.5.1 Caso I

El primer caso considera que la palabra que se va a analizar tiene todos sus caracteres completos; es decir, la red neuronal ha reconocido por lo menos un carácter para cada letra de la palabra a analizar:

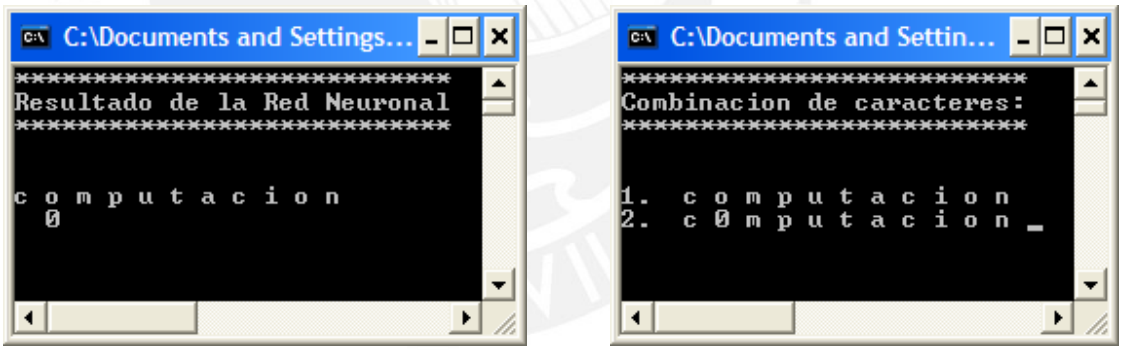


Figura 3.10 Ejemplo para el caso 1

En la etapa *Combinación de caracteres* se puede observar que la palabra correcta se encuentra en la combinación 1, entonces el sistema etiquetará esta combinación de la siguiente forma:

Combinación	Palabra	Prioridad	Error de semejanza
1	Computación	2	15 %

Para la combinación 2 se puede observar que esta combinación no existe en el diccionario por lo que el sistema buscará la palabra que más se asemeja dando como resultado la siguiente información:

Combinación	Palabra	Prioridad	Error de semejanza
2	Computación	1	40 %

Finalmente, una vez terminado el análisis de las combinaciones se usa la información de cada tabla y se selecciona la palabra correcta. Para la selección de la palabra correcta siempre se analiza primero la prioridad de cada palabra, en donde la prioridad 2 es la más importante, y así se descartan ciertas combinaciones. Por otro lado, si dos o más palabras tienen la misma prioridad se procederá a analizar el error de semejanza y de este análisis se seleccionara el que tiene el menor error. Para el presente ejemplo el sistema seleccionará la combinación 1, por tener la más alta prioridad.

3.5.2 Caso II

El segundo caso considera que la palabra que se va a analizar tiene todos sus caracteres completos; es decir, la red neuronal ha reconocido por lo menos un carácter para cada letra de la palabra a analizar:

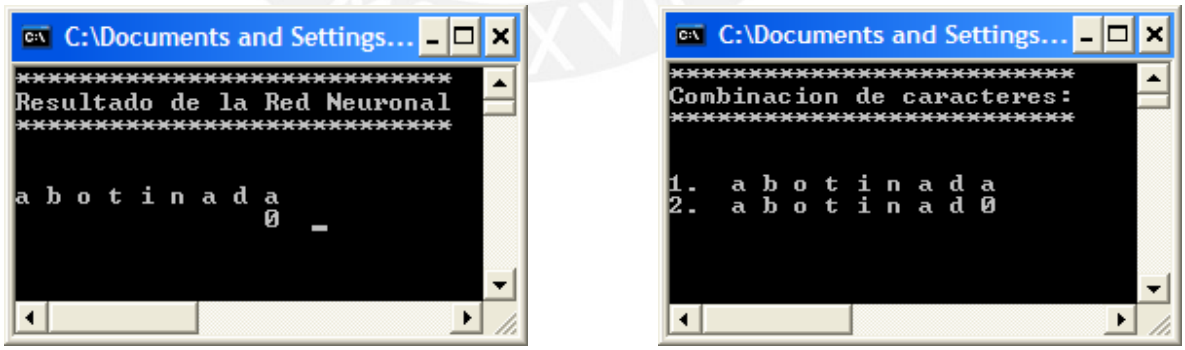


Figura 3.11 Ejemplo para el caso 2

Para esta palabra, una vez realizado el análisis de las combinaciones de los caracteres el sistema dará los siguientes resultados:

Combinación	Palabra	Prioridad	Error de semejanza
1	Abotinada	2	25 %

Para la combinación 2, se puede observar que esta combinación no existe en el diccionario, por lo que el sistema buscará la palabra que más se asemeja, dando como resultado la siguiente información:

Combinación	Palabra	Prioridad	Error de semejanza
2	abotinado	1	39 %

A diferencia del caso anterior, el sistema ha dado como resultado dos palabras que existen en el diccionario. En este caso, para la selección de la palabra correcta el sistema analizará primero la prioridad y como la combinación 1 tiene la más alta prioridad entonces escogerá esta palabra.

3.5.3 Caso III

El tercer caso considera que la palabra que se va a analizar no tiene todos sus caracteres completos; es decir, la red neuronal no ha podido reconocer algunos caracteres que forman la palabra a digitalizar, dejando este lugar en vacío. Aquí el sistema de corrección busca la palabra más cercana en base a los caracteres dados. Para este caso el sistema da como salida solo una palabra:

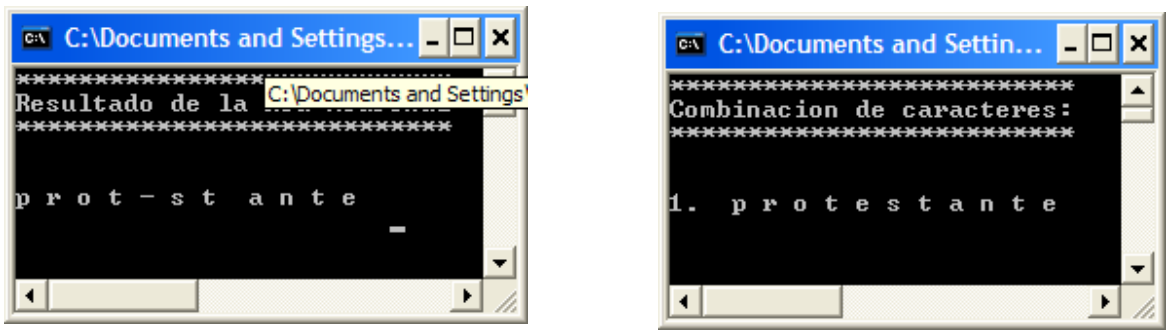


Figura 3.12 Ejemplo para el caso 3

Para esta palabra, como no se puede hacer ninguna combinación de caracteres lo que hará el sistema de corrección es realizar una búsqueda de patrones para buscar la palabra más semejante y así nos dará el siguiente resultado:

Combinación	Palabra	Prioridad	Error de semejanza
1	protestante	1	38 %

Para este caso, donde la red neuronal no reconoce algunos caracteres de la palabra el sistema de corrección utilizará la técnica de búsqueda por patrones para encontrar la palabra más cercana a los caracteres dados por la red neuronal y para el presente ejemplo seleccionó la palabra *protestante* en base a los caracteres dados.

3.5.4 Caso IV

El cuarto caso considera que la palabra que se va a analizar no tiene todos sus caracteres completos; es decir, la red neuronal no ha podido reconocer algunos caracteres que forman la palabra a digitalizar, dejando este lugar en vacío. Aquí, el sistema de corrección busca la palabra más cercana en base a los caracteres dados. Para este caso el sistema da como salida dos o más palabras:

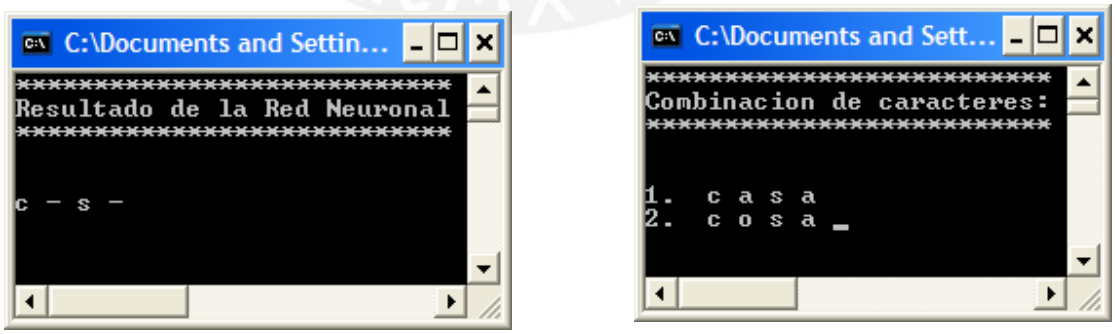


Figura 3.13 Ejemplo para el caso 4

Para esta palabra, como no se puede hacer ninguna combinación de caracteres, lo que hará el sistema de corrección es realizar una búsqueda de patrones para buscar la palabra más semejante y para el presente ejemplo nos da el siguiente resultado:

Combinación	Palabra	Prioridad	Error de semejanza
1	Casa	1	30
2	Cosa	1	30

Para este caso, como el sistema ha seleccionado dos palabras diferentes con la misma prioridad y error de semejanza, la salida del sistema de corrección considerará las dos palabras como correctas y para ello el usuario será quien decida que palabra es la correcta de acuerdo al contexto.

3.6 Conclusiones

El archivo generado por el OCR debe tener un formato adecuado para que pueda ser interpretado por el sistema.

La selección de los caracteres es una parte importante para formar las palabras que luego serán analizadas con las ayuda de una base de datos que contiene la mayoría de las palabras del vocablo español.

Mientras más grande sea la base de datos donde se almacenan las palabras del diccionario se requerirá más tiempo de procesamiento al momento de hacer el análisis respectivo.

Finalmente, el algoritmo de búsqueda de patrones que se esta usando permite la búsqueda de las palabras se realice alrededor de 0.1 segundos en una base de datos de cien mil palabras, lo que facilita un procesamiento mas rápido de los documentos.

CAPITULO 4:

EVALUACIÓN DEL SISTEMA DE CORRECCIÓN IMPLEMENTADO

4.1 Análisis de los resultados de las etapas del sistema:

A continuación se mostrarán los resultados obtenidos por cada etapa del sistema de corrección:

4.1.1 Lectura del archivo *.txt

La figura 4.1 muestra el archivo generado por el OCR y la figura 4.2 muestran la lectura del archivo con un programa diseñado para esta tarea.

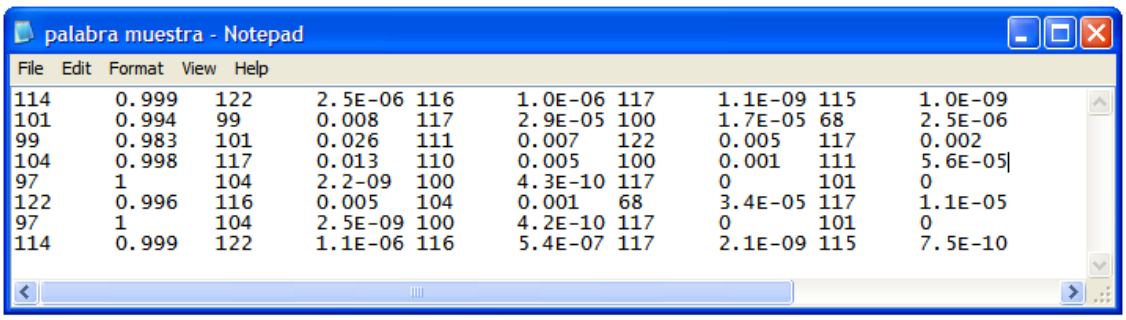


Figura 4.1 Archivo generado por el OCR

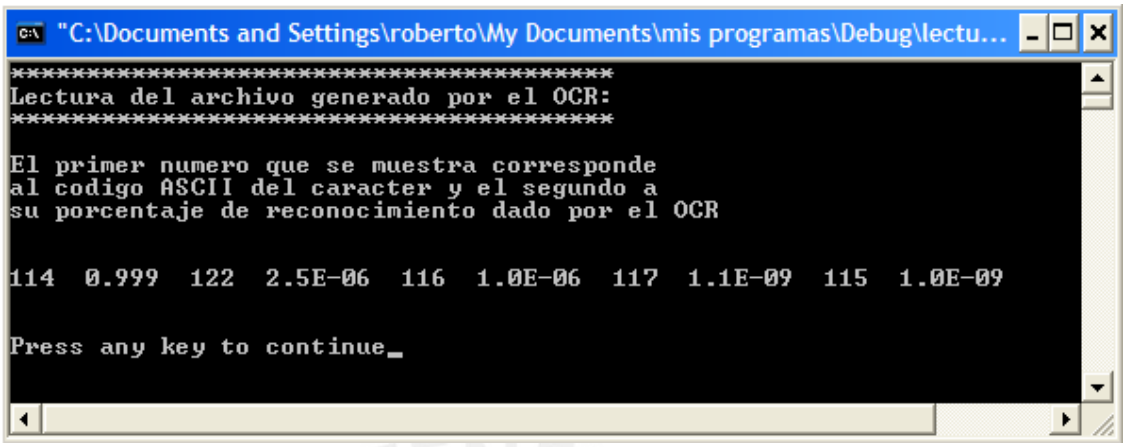


Figura 4.2 Lectura del Archivo generado por el OCR

Los datos leídos por el programa serán almacenados en arreglos para luego ser procesados por algoritmos del programa.

4.1.2 Análisis de caracteres

En La figura 4.3 se muestra los cinco caracteres que tienen las mayores probabilidades de ser el caracter del texto que se esta digitalizando:

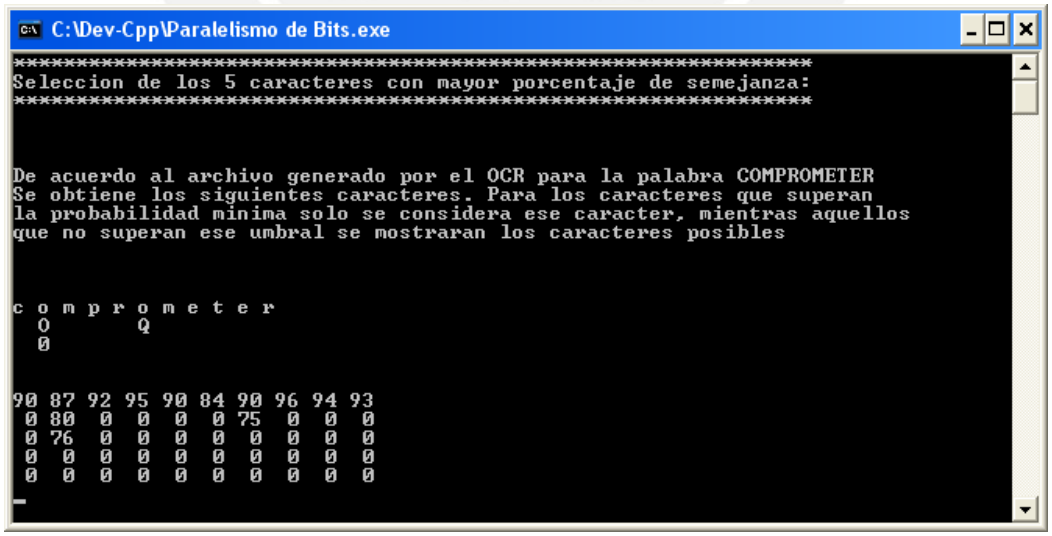


Figura 4.3 Caracteres seleccionados

Esta gráfica muestra el resultado del algoritmo que se encarga del análisis de los caracteres en el cual muestra los caracteres que superan el umbral mínimo de reconocimiento. Para este ejemplo, los caracteres c, m, p, e, t, r superaron el umbral, mientras los caracteres o, O, 0, Q no lo superaron.

4.1.3 Combinación de caracteres

Figura 4.4 Combinación de caracteres

Como se puede observar en la figura 4.4 se logra obtener la combinación de todos los caracteres seleccionados en la etapa anterior. Las combinaciones que se obtienen son: comprometer, comprQmeter, cOmpromenter, cOmprQmeter, c0mprometer, c0mprQmeter.

4.1.4 Análisis de la palabra

En esta etapa se muestra el resultado del análisis ortográfico que se hace a cada una de las combinaciones obtenidas en la etapa anterior. Si la palabra

existe en el diccionario, ésta se indicará con un mensaje tal como se hace con la palabra *comprometer*

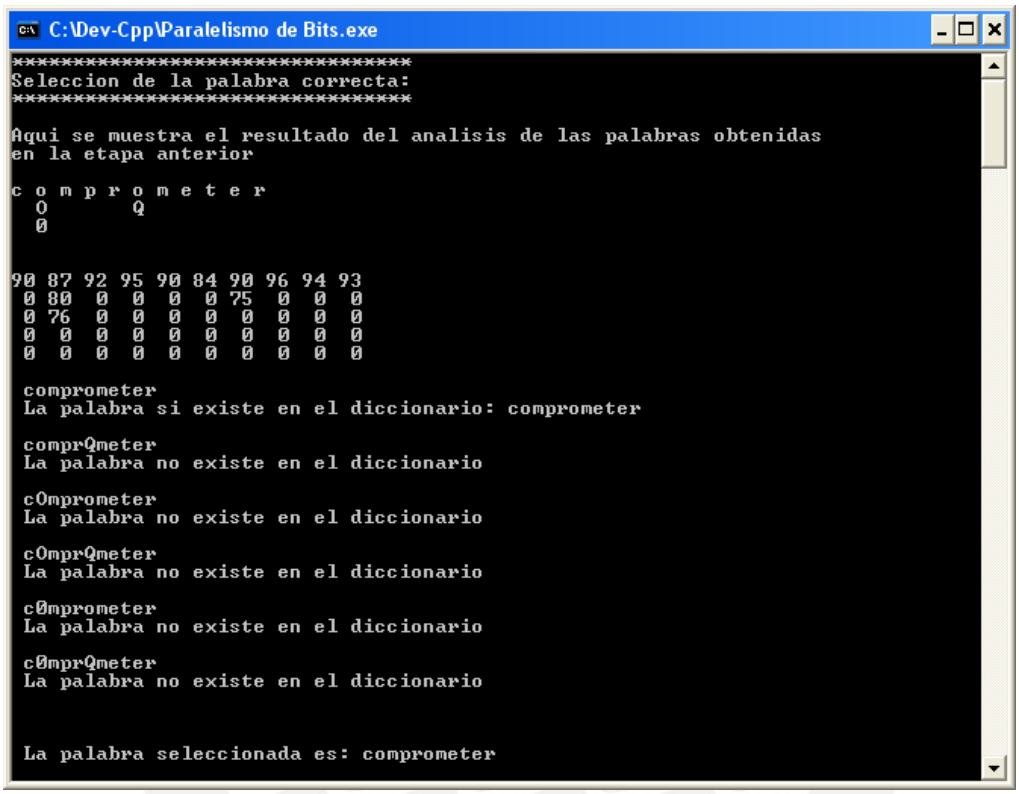


Figura 4.5 Análisis de Corrección

4.1.5 Entrega de la palabra seleccionada en un texto editable

En la figura 4.6 se muestra el resultado final, en donde se entrega la palabra seleccionada en un archivo editable.

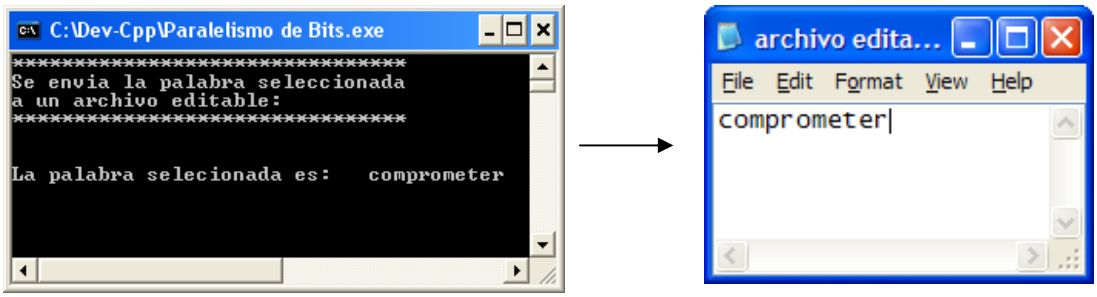


Figura 4.6 Entrega en archivo editable

4.1.6 Evaluación del rendimiento del sistema

A continuación se muestra una tabla de rendimiento del sistema en base al número de caracteres que tiene la palabra:

Número de palabras	Número de Caracteres	Palabras corregidas	Palabras no corregidas
20	1 < 4	95%	5%
20	5 < 6	92%	8%
20	7 < 8	90%	10%
20	9 < 10	79%	21%

Los patrones con los que se experimentó estas pruebas se muestran en el anexo 1.

A continuación se muestra la eficiencia global del sistema:

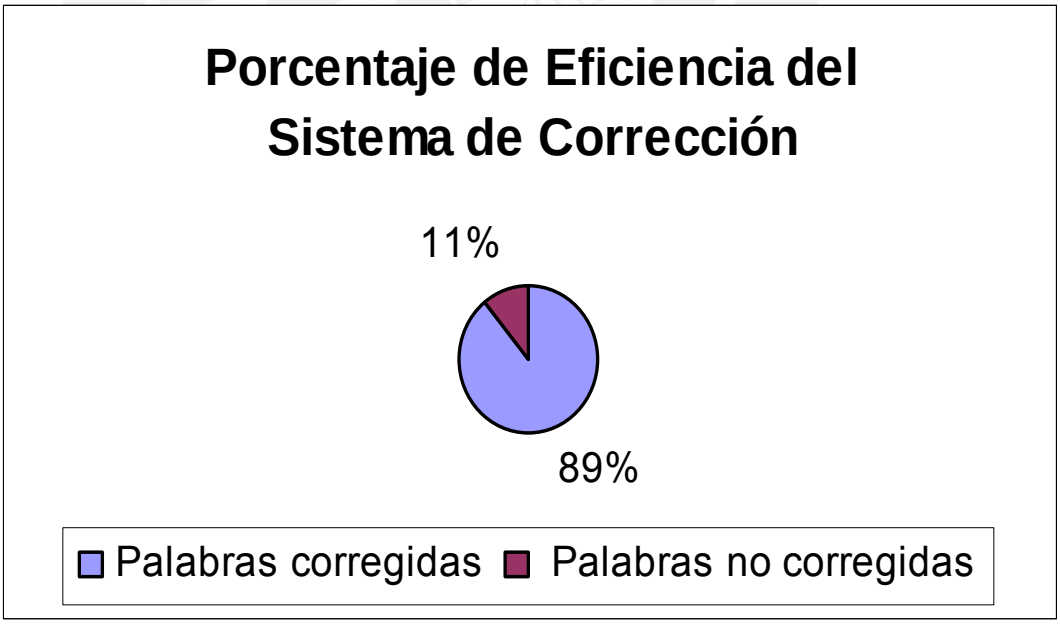


Figura 4.7 Rendimiento del sistema

4.2 Evaluación del algoritmo de búsqueda

Para la ejecución de los experimentos que se realizaron en este trabajo, se utilizó una computadora exclusivamente para este propósito. Las características del hardware y software son: Procesador Intel Pentium IV a 1700 MHz, 512 MB de RAM, 80 GB de disco duro, Sistema Operativo Windows XP. Para probar el algoritmo se utilizó un archivo tipo .txt que contiene 96 314 palabras del vocablo español, los cuales han sido obtenidos del diccionario del Diccionario de la Real Academia Española.

Los patrones se escogieron de forma aleatoria y se experimentó con longitudes del patrón de 4 hasta 10 caracteres. El algoritmo de búsqueda aproximada se implementó en Visual C++, los resultados que se obtuvieron se compararon con un algoritmo de búsqueda lineal.

Las siguientes figuras muestran los tiempos obtenidos en la búsqueda de patrones de longitud menor a 10 caracteres. Como se puede observar, el algoritmo propuesto en este trabajo es hasta 13 veces más rápido que la búsqueda lineal. Esta comparación se obtuvo de la siguiente manera:

Palabra a buscar: ***rechazar***

Usando el algoritmo lineal (Figura 4.8): 2.6 seg.

Usando paralelismo de bits (Figura 4.9): 0.2 seg.

Dividiendo los tiempos de cada algoritmo se obtiene lo siguiente:

$$\frac{Vparalelismo}{Vlineal} = \frac{2.6}{0.2} = 13$$

M < 10

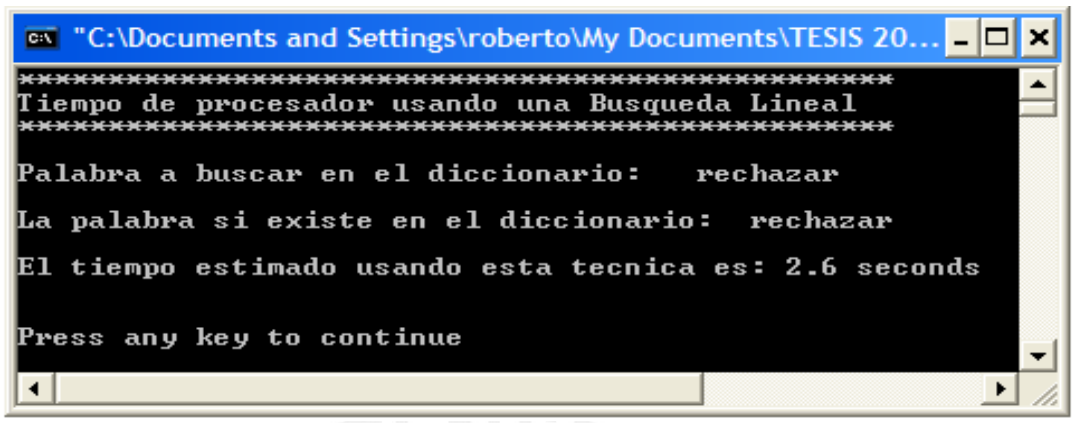


Figura 4.8 Tiempo de búsqueda lineal, para patrones con menos de 10 caracteres.

M < 10

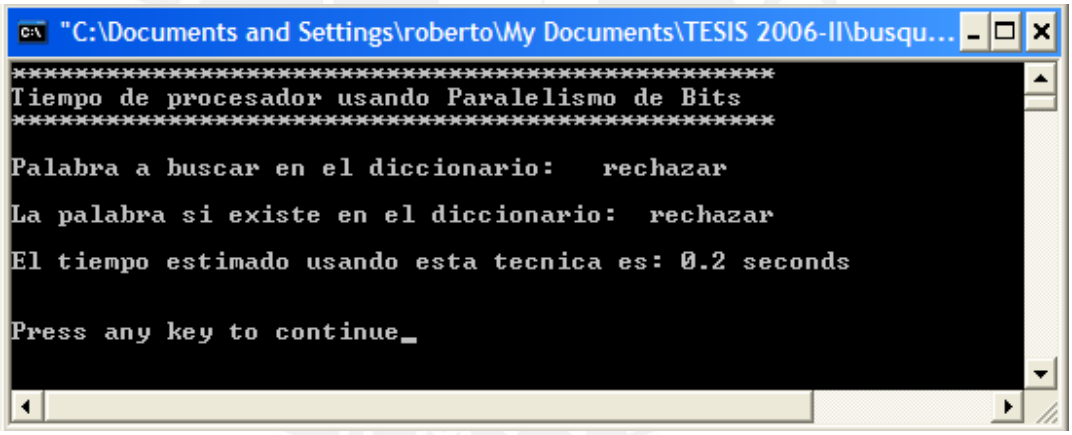


Figura 4.9 Tiempo de búsqueda usando paralelismo de bits, para patrones menores a 10 caracteres.

Los patrones con los que se experimentó las pruebas anteriores se muestran en el anexo 1

A continuación se muestra otra perspectiva del comportamiento del algoritmo en comparación con la búsqueda lineal. Las gráficas muestran cómo el algoritmo mantiene una alta eficiencia al aumentar el tamaño de la base de datos.

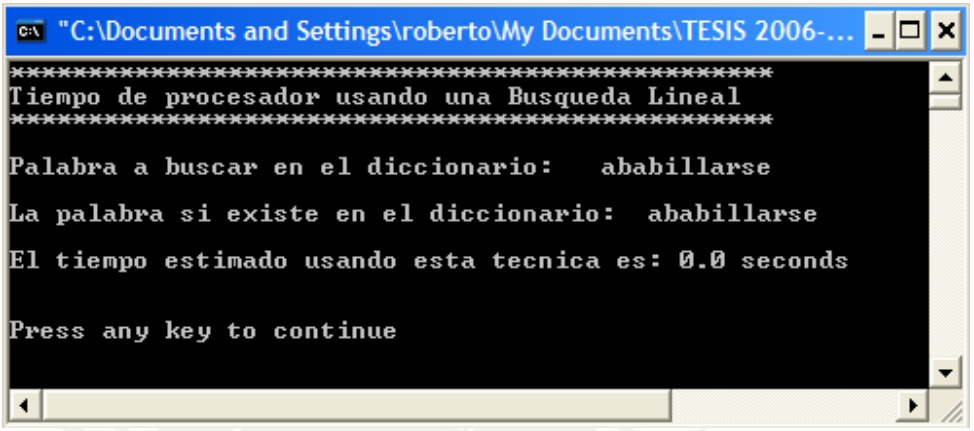


Figura 4.10 Tiempo de búsqueda lineal para un archivo de 1000 palabras

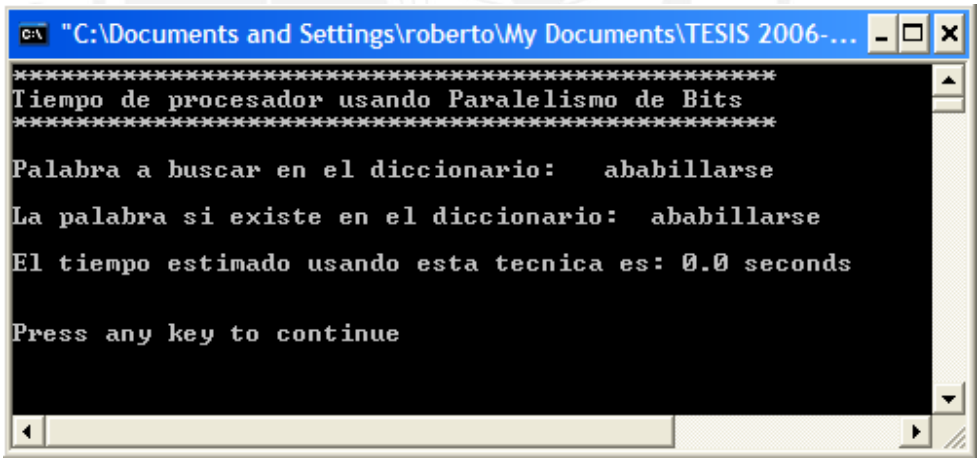


Figura 4.11 Tiempo de búsqueda usando paralelismo de bits para un archivo de 1000 palabras.

Para una base de datos pequeño ambos algoritmos de búsqueda tiempo el mismo tiempo de procesador, pero cuando la base de datos es más grande se empiezan a notar las diferencias.

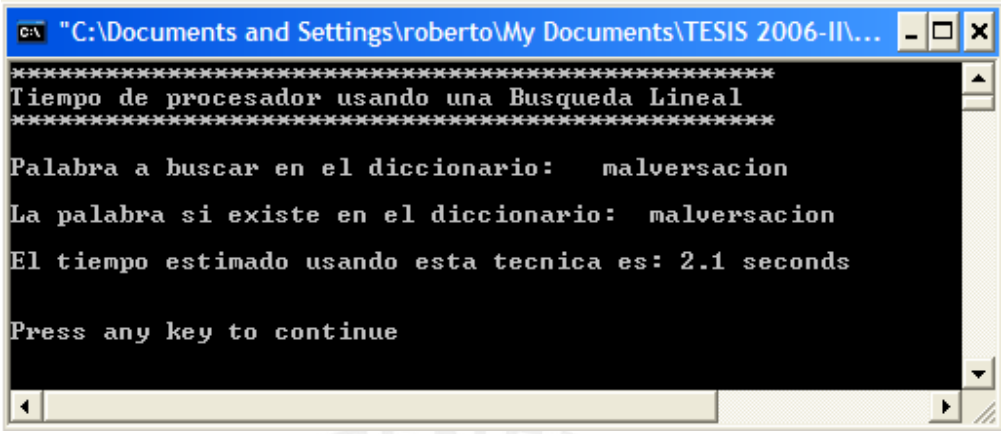


Figura 4.12 Tiempo de búsqueda lineal para un archivo de 62 000 palabras

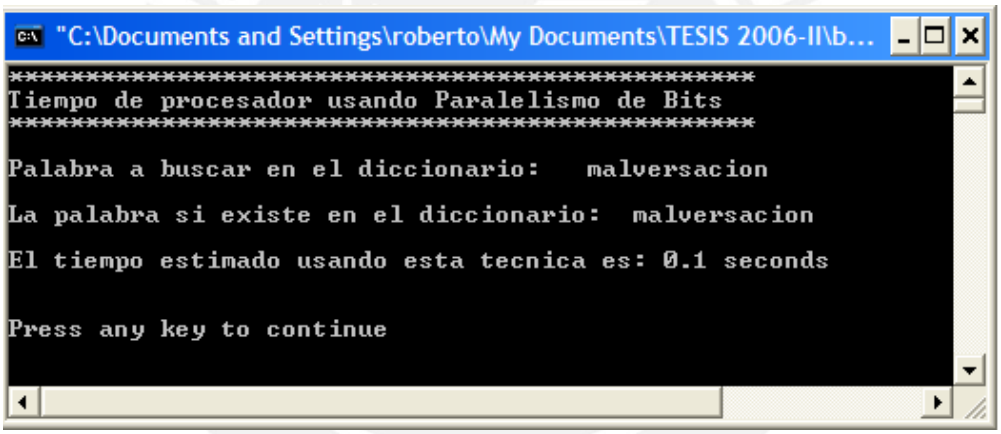


Figura 4.13 Tiempo de búsqueda usando paralelismo de bits para un archivo de 62 000 palabras.

Cuando se tiene un archivo de 62 000 palabras se puede notar que el algoritmo de búsqueda lineal es 21 veces mas lento (2.1/0.1) que el algoritmo de paralelismo de bits, notándose claramente la eficiencia de este último.

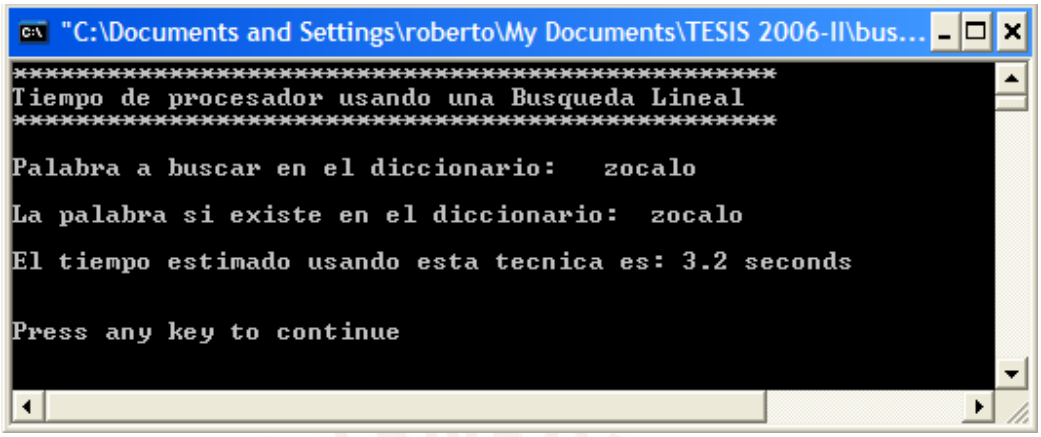


Figura 4.14 Tiempo de búsqueda lineal para un archivo de 96 314 palabras

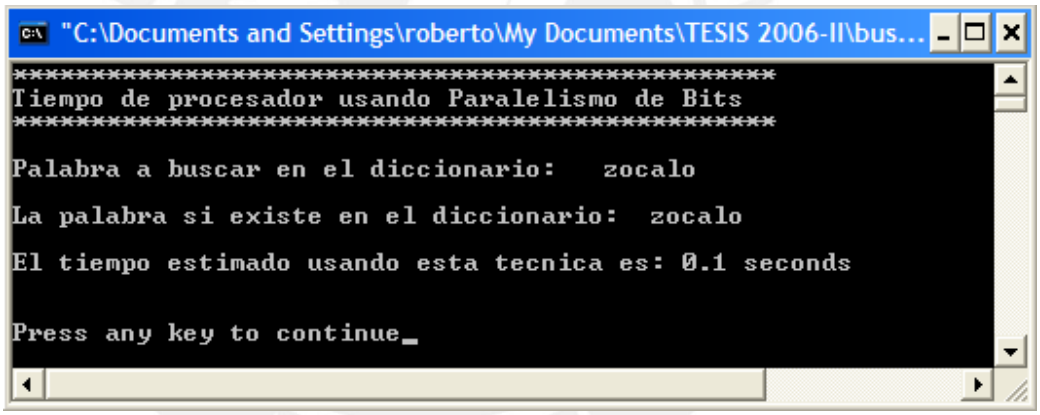


Figura 4.15 Tiempo de búsqueda usando paralelismo de bits para un archivo de 96 314 palabras.

Finalmente, se considera un archivo de 96 314 para hacer la comparación y nuevamente se nota que el algoritmo de paralelismo de bits es mucho más eficiente que el de búsqueda lineal.

4.2.1 Rendimiento de los algoritmos de búsqueda

A continuación se muestra la comparación en cuanto a rapidez de los algoritmos de búsqueda:

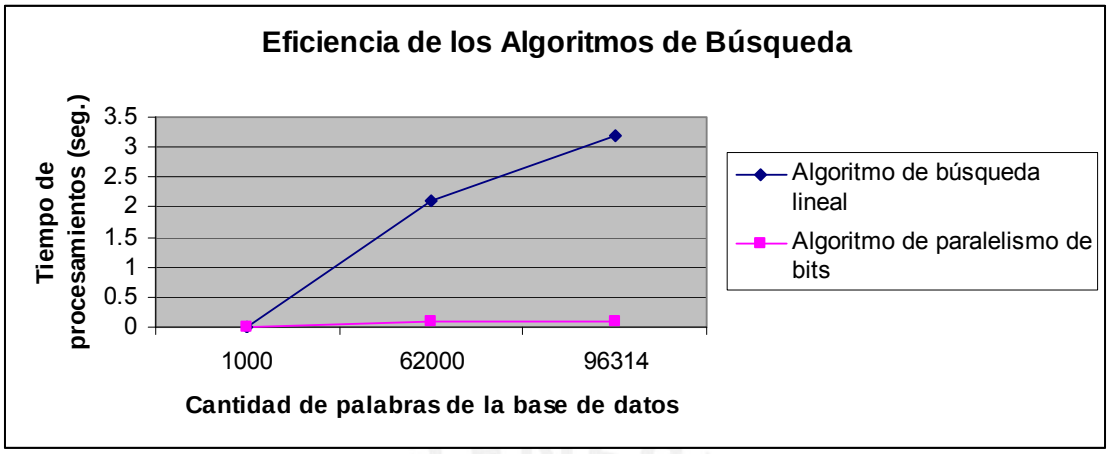


Figura 4.16 Comparación de los algoritmos de búsqueda

4.3 Precio del software y desarrollo del sistema de corrección

4.3.1 Precios de actuales software para OCR:

A continuación se muestran los precios de algunos softwares OCR que se usan actualmente en la digitalización de documentos:

Software OCR con licencias:

Nombre del Software	Precio ($)
Readiris Pro 9	129.99
FineReader Pro 7.0	171.98
CuneiForm OCR 3.1	49.00
SimpleCopier 4.07	30.00
CopyNook 1.10	19.95

Software OCR libre:

Nombre del Software	Proveedor
OCROPUS	Google
OCRE Proyecto	Proyecto GNU
Tesserat OCR	Google

4.3.2 Precio del sistema de corrección

A continuación se realizará la cotización acerca de cuánto costará realizar la implementación de un proyecto de corrección ortográfica. Para la implementación del proyecto se contará con dos fases de desarrollo: la fase 1 se encargará de cotizar el valor de cuánto cuesta realizar el proceso de análisis y diseño del software de corrección ortográfica, el cual estará enfocado en los textos deteriorados; la fase 2 se encargará de cotizar el valor de cuanto cuesta realizar el desarrollo y pruebas del software. A continuación se muestra una tabla con la cotización de ambas fases:

Etapa	Número de recursos (personas)	Horas por mes	Número de meses	Número de horas-hombre
Fase 1 Análisis y diseño	2	20	3	120
Fase 2 Desarrollo y pruebas	2	40	3	240

De acuerdo a los datos mostrados, el número total de horas-hombre en el proyecto es:

Total horas-hombre = 120 + 240
Total horas-hombre = 360

Una vez realizado el análisis de cada fase se procede a calcular el costo de todo el proyecto:

- Cálculo de cuanto cuesta el proyecto
  Para calcular el costo del proyecto se requiere multiplicar el número total de horas-hombre por el valor que cuesta contratar a una persona por una hora de su trabajo, esto es:

Costo = 360 x S/.20 = 7200 soles

- Cálculo del precio de venta

  Para calcular el precio de venta se requiere multiplicar el número total de horas-hombre por el valor al cual vendo una hora de trabajo de mis recursos, esto es:

Precio = 360 x S/.30 = 10800 soles

Con este cálculo se obtiene el precio que costaría el desarrollo del sistema de corrección ortográfica que se aplicaría a los documentos deteriorados.

4.4 Conclusiones

En este trabajo se ha presentado una solución al problema ortográfico que presentan los OCR cuando los documentos que se están analizando tienen poca nitidez o están deteriorados; por otro lado, también se ha presentado un eficiente algoritmo de búsqueda de palabras. Con los resultados que se obtuvieron de las pruebas realizadas, se puede concluir que con el algoritmo propuesto se puede efectuar la búsqueda aproximada de patrones simples en forma más rápida que con el algoritmo de búsqueda lineal.

CONCLUSIONES

1. De acuerdo a los resultados, el sistema propuesto ha mejorado la eficiencia del software OCR en un 15% cuando es usado con documentos deteriorados.

2. El rendimiento del sistema se mantiene alto al crecer el tamaño de la base de datos a diferencia del la búsqueda lineal en donde su rendimiento es alto con archivos pequeños pero cae rápidamente cuando la base de datos crece, en donde el tiempo necesario para buscar una palabra puede aumentar de 0.1 seg. A 3.2 seg.

4. De acuerdo a los objetivos planteados el sistema propuesto tiene una velocidad de procesamiento bastante alto y esto debido al uso del algoritmo de paralelismo de bits, requiriéndose un tiempo total de 0.1 seg. para buscar una palabra dentro de un archivo que contiene 96 314 palabras.

5. Los sistemas basados en lingüística computacional han permitido incrementar la eficiencia de los sistemas de reconocimiento de caracteres haciendo un análisis contextual de las palabras.

RECOMENDACIONES

1. Para que el sistema de corrección sea más robusto y de mejores resultados, sería recomendable aumentar una etapa más al sistema, el cual se encargaría de hacer un análisis contextual de la palabra que se está analizando.

2. Para la ejecución del algoritmo se debe contar con el equipo adecuado: computador. Para el computador es recomendable que sea por lo menos un Pentium IV, 1700 MHz, 512 MB de memoria RAM y 60 GB de disco duro.

3. Se debe verificar que el archivo generado por el OCR se encuentra en la dirección correcta dentro del computador para su lectura mediante el programa propuesto.

4. El sistema de búsqueda que se ha implementado es de propósito general; es decir, si se quiere usar este algoritmo en otras aplicaciones es recomendable darle el formato adecuado para su análisis.

FUENTES

[1] Dye, Jessica
2006 Scanning the stacks. EContent [en línea], 29 (1). [consultado 2005/04/01] EBSCO Academic Search Premier

[2] Zimmermann, Matthias; Chappelier, Jean-Cédric; Bunke,
2006 Offline Grammar-Based *Recognition* of Handwritten Sentences**,** *IEEE Transactions on Pattern Analysis & Machine Intelligence*,[en línea], Vol. 28 (5),). [consultado 2005/04/03] EBSCO Academic Search Premier,

[3] By: Jantz, Richard
2006 Abbyy OCR Pushes Paper Proficiently**,** *PC World*, ,[en línea], 24 (1) , [consultado 2005/04/03] EBSCO Academic Search Premier,

[4] Ma, Matthew Y; Guo, Jinhong K; Wang, Patrick S P
2004 From pixels to true xml structures in digital document images International Journal of Pattern Recognition and Artificial Intelligence. Vol. 18, no. 6, pp. 1057-1069.

[5] Optical recognition of hand-printed character of any size, position and orientation [en línea].
http://www.research.ibm.com/journal/rd/363/ibmrd3603K.pdf

[6] Kelly, Matt
2005 OCR system puts DOT on course., eWeek [en línea] 22 (45), p16-17, 2p, [consultado 2005/04/03] EBSCO Academic Search Premier

[7] Design of logic for recognition of printed character by simulation [en línea], [consultado 2005/04/20]
http://www.research.ibm.com/journal/rd/011/ibmrd0101C.pdf

[8] Emerging Technologies Knowledge Base
[en línea], 24 (1) , [consultado 2005/04/20]
http://emergingtech.ittoolbox.com/topics/t.asp?t=310&p=332&h2=332&h1=310

[9] Biblioteca Virtual Miguel de Cervantes
[en línea], 24 (1) , [consultado 2005/05/16]
http://www.cervantesvirtual.com/

[10] Segmentation methods for recognition of machine-printed character
[en línea], 24 (1) , [consultado 2005/04/26]
http://www.research.ibm.com/journal/rd/152/ibmrd1502J.pdf

[11] Búsqueda aproximada en texto comprimido
[en línea], [consultado 2005/04/10]
http://ccc.inaoep.mx/Reportes/CCC-04-007.pdf

[12] Automated forms-processing software and service [en línea].
[en línea], 24 (1) , [consultado 2005/04/26]
http://www.research.ibm.com/journal/rd/402/gopisetty.pdf

[13] Nishida, Hirobumi
2005 Restoring high-resolution text images to improve legibility and OCR Accuracy Proc. SPIE. Vol. SPIE-5676, pp. 136-147.

[14] Namane, Abderrahmane; Arezki, Madjid; Guessoum, Abderrezak

2005 Sequential neural network combination for degraded machine-printed character recognition, Proc. SPIE. Vol. SPIE-5676, pp. 101-110

[15] An algorithm for separating patterns by ellipsoids
[en línea], 24 (1) , [consultado 2005/05/02]
http://www.research.ibm.com/journal/rd/266/ibmrd2606M.pdf

ANEXO N°1

Palabras Usadas en el Sistema de Corrección

1. abanico
2. acerbo
3. ahogado
4. angloamericana
5. baraja
6. barquero
7. bebedora
8. brumoso
9. cangrejo
10. caridad
11. elegido
12. embate
13. empaquetar
14. faena
15. foraneo
16. futuro
17. glacialmente
18. goteo
19. imprimir
20. informar
21. interrupción
22. librar
23. madera
24. maletin
25. octava
26. oratoria
27. pagano
28. paralela
29. psicologa
30. reciproco
31. ruego
32. santulon
33. sana
34. sede
35. secundario
36. sintonia
37. sugestionable
38. tartamuda
39. teatro
40. termómetro
41. trigonometría
42. uniformidad

43. universal
44. vaporable
45. velada
46. vicepresidenta
47. zodiacal
48. zoquete
49. zorzal
50. zurrumba