BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

FACULTAD DE CIENCIAS DE LA COMPUTACIÓN

MAESTRÍA EN CIENCIAS DE LA COMPUTACIÓN

Técnicas para la normalización de vocabularios en textos cortos.

Tesis que para obtener el grado de

Maestro en Ciencias de la Computación

Presenta:

Narce Francisco Rosales Leyva

Asesor: Dr. David Eduardo Pinto Avendaño

Co-asesor: Dra. Darnes Vilariño Ayala

Febrero, 2015

Puebla, Puebla, México

AGRADECIMIENTOS

A Dios…

Por estar conmigo en todo momento y darme la oportunidad de vivir, por ayudarme a pasar los buenos y malos momentos, por darme una gran familia y la amistad de aquellas personas que estuvieron en mí camino.

A mis padres…

Noemi Leyva Rivera, Miguel Rosales López

Por ser el pilar fundamental en todo lo que soy, en la formación que me otorgaron, tanto academia como en la vida, este trabajo ha sido posible gracias a ustedes. Gracias por todo.

A mis hermanos…

Dalia Rosales, Miguel Rosales, Zaqueo Rosales y Marcos Rosales

Por estar siempre ahí, apoyándome en este camino que emprendí y logre gracias a su completo y constante interés de mi persona. Por todo el cariño brindado.

A mis familiares y amigos…

Por el tiempo compartido de estos dos años, por brindarme su amistad y a poyo para lograr este éxito profesional. Gracias por aquellas palabras de aliento y de fe.

A mis compañeros de la maestría…

Por el poco o mucho tiempo compartido a lo largo de estos dos años, por brindarme su amistad. En especial para el maestro Saúl, por su amistad y apoyo a lo largo de la maestría.

A mis asesores…

Dr. David Pinto y Dra. Darnes Vilariño

Por la orientación y ayuda que me brindaron a lo largo de la maestría y en la realización de esta tesis.

Resumen

Esta tesis comprende el estudio desarrollo de un modelo para la normalización de vocabularios en textos cortos. Se refiere a tomar un mensaje mal escrito, o que contenga palabras fuera de nuestro idioma, en este caso el español, y poder transcribirlo a un mensaje lo más correctamente escrito.

Este trabajo comprende de cuatro etapas básicas. En la primera es hacer un estudio exhaustivo de las principales técnicas empleadas para normalizar textos. En la segunda se muestra la creación de un sistema el cual permitió la creación de un corpus de mensajes SMS, el cual sirvió como corpus de pruebas. La tercera, se hace un análisis de los mensajes cortos, los diferentes aspectos lingüísticos que engloban a dicho mensaje y por ende los caracterizan. Por último se hace una serie de propuestas de resolver el problema de normalizado, para esto se va explicando en forma de cascada, hasta llegar a la propuesta final. Donde se contempla desde la elección del conjunto candidato, filtros para reducirlo o considerar más que coincidan con dicha palabra, posteriormente a la elección de la candidata ideal, donde se observa que hay ambigüedad y se procede a la aplicación de una técnica sintáctica. En esta etapa también se muestra la evaluación de los diferentes experimentos y se observan que tanto influyo la propuesta a mejorar y obtener una buena transcripción del mensaje.

Índice General

Índice de Figuras

Índice de Tablas

INTRODUCCIÓN

En los últimos años el crecimiento exponencial en la generación de textos cortos o comúnmente conocido como texto social, incluidos los mensajes de texto de los teléfonos móviles (SMS), los comentarios de los sitios web de medios sociales como Facebook[1] o Twitter[2], entre otros, así como las plataformas de comunicación en tiempo real como SKYPE[3], Hangouts[4], etc. Este tipo de textos cortos tienen algo en particular que los caracteriza de otros textos; primero son mensajes muy pequeños, tomando como referencia a los SMS, los cuales son mensajes de no más de 160 caracteres y que por su bajo costo de envío, los dispositivos móviles o celulares han hecho de este mecanismo indispensable para su uso, en gran parte ya que en años recientes ha habido un gran crecimiento en servicios basados en móviles, derivando principalmente de la masificación de los mismos; del cual las personas ya se acostumbraron a usar sublenguajes o lenguajes fuera del vocabulario estándar y a cometer diferentes tipos de errores de escritura del idioma español, y en específico de México, con el fin de ahorrar en lo más posible caracteres que no pasen del rango que imponen dichos mensajes.

Otros mensajes cortos, son los que se observan en las diferentes redes sociales, es una forma de comunicación textual que ha impuesto moda. Por ejemplo, [1] hasta agosto del 2013 hay 500 millones de usuarios en Twitter, de los cuales en promedio 400 millones de tweets son enviados todos los días. Se observa la cantidad exorbitante que este tipo de red social arroja de mensajes día con día, así que el poder llevar a cabo un análisis de estos mensajes se ha vuelto un estudio interesante para muchos investigadores de procesamiento de lenguaje natural, pero para ello debe existir un proceso el cual me permita aproximar lo que se quiere dar a entender en un vocabulario estándar, en este caso en el del idioma español de México.

[1] www.facebook.com
[2] www.twitter.com
[3] https://www.skype.com/es/
[4] http://www.google.com/hangouts/

Una de las razones es que la mayoría de las personas se comunican con mensajes cortos y que al escribir en cualquiera de estos medios de comunicación textual, lo hacen en un lenguaje que no se adhiere a la gramática convencional, las reglas de puntuación y de pronunciación usuales, es decir, tienden a faltas de ortografía, uso de abreviaturas no-estándar, acrónimos, sustituciones y omisiones de palabras, transliteraciones fonéticas e incluso neologismos, por lo que hace difícil la comprensión de lo que realmente intentan dar a entender. Principalmente los jóvenes son los que tienden a cometer este tipo de errores ya que no tienen o están en proceso de tener buenos fundamentos de escritura como aquellas personas que ya tiene un grado académico mayor, además que son mayoría en el uso de estos medios.

Esta tendencia atrae una gran cantidad de investigación con el fin de extraer información valiosa y conocimiento de estos datos. Desafortunadamente, las herramientas tradicionales de Procesamiento de Lenguaje Natural (PLN) a veces tienen un mal desempeño en el tratamiento de este tipo de texto.

Desde el punto de vista de los sistemas de recuperación de información, es importante explotar el potencial existente en el número de usuarios que hacen uso de estos tipos de servicios, sin embargo, lo anteriormente expuesto deja claro la dificultad existente en la construcción de un sistema de recuperación de información que contemple como texto de consulta (entrada) a un texto corto, incluso para preguntas cuyas respuestas están bien documentadas como son las preguntas más frecuentes ó Frequently Asked Questions (FAQ, por sus siglas en inglés). Por lo que para tener buenos resultados en la recuperación de información, es muy importante aplicar antes, un pre-procesamiento del mensaje para producir una mejor representación léxica y que la calidad del programa de análisis de texto mejore considerablemente, este paso, llamado normalización, el cual se encarga de tomar *tokens* que se encuentren en forma no estándar, los cuales son creados intencionalmente o involuntariamente por los usuarios y poder restaurarlos a su forma correcta de entre N posibles candidatos de salida; sin embargo estos *tokens ruidosos* o comúnmente conocidos como palabras fuera del vocabulario (*OOV*, por sus siglas en inglés), no son fáciles de definir, es decir, no puede ser considerada solo como una palabra que no existe en el vocabulario estándar, ya que la naturaleza de los mensajes cortos es muy dinámica; por lo que hay que saber reconocerlas primeramente.

En nuestro trabajo nos enfocaremos en trabajar y crear nuevas técnicas de normalización a partir de las que ya existen (proponiendo modificaciones) las cuales nos permitan reducir el ruido que arrojan comúnmente los textos cortos y tratar de convertirlo a un texto el cual satisfaga el uso correcto del vocabulario y las reglas gramaticales; se pretende proponer un método que ataque a este problema. Se va a trabajar en mensajes de textos cortos en el idioma español. Para esto, antes se debe crear un corpus el cual se apliquen aquellas técnicas propuestas, y así poder llevar acabo las evaluaciones correspondientes que permitan elegir la mejor técnica de normalizado.

1.1. Planteamiento de la investigación

En esta sección se precisa el problema de la investigación a resolver y se definen los objetivos del proyecto. Del mismo modo se plantea la propuesta de solución. Por último se describe la organización de la tesis.

1.1.1. Problema a resolver

La tarea a abordar en este trabajo consiste en la elaboración de algoritmos y métodos para llevar acabo la normalización de vocabularios de textos cortos en el idioma español de México. Con esta metodología se pretende, en un futuro, sea usada como posible pre-procesamiento hacia una tarea de Recuperación de Información (RI), obteniendo un mejor desempeño de ésta, lo cual nos arrojen resultados más precisos o que se acerquen a lo que uno pretenda obtener. En la Figura 1 se puede observar gráficamente el problema a resolver.

Al tener que tratar con textos cortos los cuales casi siempre vienen escritos con un sub-lenguaje, el cual es dinámico, es decir, siempre está en constante cambio; el sistema debe ser capaz de comprender que palabras están dentro del vocabulario estándar (*IV,* por sus siglas en inglés*)* y cuales están *OOV.* Por lo que el sistema necesita:

1. Localizar la lista de palabras *OOV* en el texto corto.
2. Para cada palabra *OOV*. Aplicar alguna técnica de reducción de palabras candidatas, así en vez de tener que buscar la palabra que mejor se

acople, en todo mi diccionario de palabras *IV* solo aplicar la técnica de normalizado a un número reducido.

3. Aplicar la técnica de normalizado a cada palabra *OOV* con cada lista reducida correspondiente, de la cual se elegirá la que mejor se acople y se convierta como la candidata ideal.

Figura 1. Descripción gráfica del problema

Cabe destacar, que este tipo de mensaje corto, a menudo contiene palabras combinados con dígitos u otro carácter no alfanumérico, y que en si no se encuentran en nuestro diccionario de palabras *IV*, por lo que el sistema debe ser capaz de detectarlos y etiquetarlos, de igual modo detectar formatos de hora, fecha, paginas, correos, entre otros.

1.1.2. Objetivos de la investigación

- **Objetivo General**

  - Crear un método el cual me permita llevar a cabo la normalización de vocabularios en textos cortos basados en el idioma español de México.

- **Objetivos Específicos**

  - Estudio de las diferentes propuestas de normalizado del estado del arte.
  - Creación de un corpus paralelo de mensajes cortos del idioma español de México.
  - Experimentar con diferentes algoritmos propuestos en los diversos artículos del estado del arte.
  - Evaluación de las diferentes técnicas de normalizado.
  - Plantear una o más posibles propuestas de normalizado.

1.1.3. Justificación de la investigación

El pre- procesamiento, o comúnmente conocido como normalización, enfocado a textos cortos, es una tarea la cual con la mejor interpretación que arroje de un texto corto que contenga palabras *OOV*, se obtendrá una mejor respuesta al aplicarlo antes de realizar tareas de RI.

El hecho de trabajar con textos cortos, nos da pie al uso y análisis de un gran cantidad de ellos, por lo que hoy en día se vive, hablando tanto de los mensajes SMS, que a pesar de los años, siguen siendo un pilar de comunicación textual muy popular y que a diferencia de otros mecanismos de comunicación a través de cualquier medio, estos se han estandarizado en los dispositivos móviles. Otro tipo de textos cortos y que ya se mencionó en los apartados anteriores pero que hay que volver a recalcar, son los que nos ofrecen las redes sociales, que no tienen mucho que salieron a la luz y que por su crecimiento exponencial del manejo de cualquiera de ellas, el poder hacer análisis de ellos, se ha vuelto un reto para muchos investigadores y por ende para nosotros.

También, por mencionar que este tipo de mensajes, contienen un alto grado de términos que no existen en los diccionarios normales, las palabras que más se usan en éstos, son nuevas terminologías las cuales regularmente tiene que ver con contracciones de palabras originales o con una representación ortográfica de su correspondiente fonética. Cabe mencionar que dicha terminología cambia dependiendo al rango de edad de cada persona y a su rango académico,

recalcando que las personas jóvenes son las que suelen tener el mayor uso de representaciones fonéticas, lo cual puede ser derivado del hecho que ellos no dominan aún el vocabulario de su lenguaje nativo.

Esta línea de investigación hay mucho material para el idioma inglés y no para el idioma español y mucho menos para el español de México, el cual lo que se logre obtener va ser algo en donde pondremos un punto de partida para futuros trabajos en este idioma. Además, que de donde se pretende tomar y evaluar los mensajes son de mecanismos de comunicación textual que tienen hoy en día un fuerte auge y del cual se puede obtener mucha información que va ser posible analizar.

Tomando en cuenta el preámbulo anterior, surge la motivación e interés en esta investigación. En este sentido es importante porque se propone el análisis de las diferentes técnicas propuestas por otros investigadores, así como las que se pretenden proponer.

1.2. Organización de la tesis

Este trabajo de investigación se estructura en 6 capítulos distribuidos de la siguiente manera:

- **Capítulo 1.** Introducción. En este apartado se detalla el problema a resolver, los objetivos y la justificación de la investigación.

- **Capítulo 2.** Estado del arte. En este apartado se presentan un conjunto de trabajos de investigación relacionados con el ámbito de Normalización de textos cortos.

- **Capítulo 3.** Creación de un Corpus paralelo de mensajes cortos en español. En este apartado se describe la idea de cómo se llevó acabo la creación de un corpus paralelo.

- **Capítulo 4.** Aspectos Lingüísticos al tratamiento de mensajes SMS. En este apartado se explican los diferentes aspectos lingüísticos comunes, encontrados en los mensajes cortos.

- **Capítulo 5.** Modelo de Normalización de Textos Cortos. En este apartado se explica la propuesta del modelo para resolver el problema de normalización.

- **Capítulo 6.** Evaluación de la propuesta y Resultados. En este apartado, concluye con la evaluación de los diferentes experimentos y se describen las características de los corpus empleados y de los resultados obtenidos.

CAPÍTULO 2

ESTADO DEL ARTE

En este capítulo se presenta una visión general del estado del arte de las técnicas de normalización en textos cortos. Se describen los trabajos más relevantes que se han desarrollado. Se comentan y se discuten los enfoques, técnicas, procedimientos y herramientas que se han propuesto e implementado.

2.1. Descripción de los métodos utilizados por diferentes investigadores

Para tratar los diferentes tipos de errores y usos de sub-lenguajes que acarrean los textos cortos, en los últimos años se ha tratado de afrontarlos proponiendo diversas formas de normalizarlos, existen varias propuestas de diferentes investigadores, entre los cuales unos trabajan sobre los errores de transliteraciones fonéticas que la mayoría de las personas cometen a la hora de escribir palabras tal y como suenan, proponen diferentes algoritmos fonéticos los cuales obedecen a la necesidad de recuperar información que tiene una semejanza sonora; y cuya representación a través de la palabra escrita pueda diferir de su pronunciación.

Hay ciertos sonidos que forman el núcleo del idioma inglés, y cuyos sonidos son inadecuadamente representados por letras del alfabeto, como un sonido puede en algunos casos ser presentados por más de una letra o combinación de letras, y viceversa, una letra o combinación de letras puede ser representado por dos o más sonidos.

Cuando aún no existían los ordenadores Robert Russell y Margaret Obell [6] diseñaron *Soundex*, el objetivo de este algoritmo era codificar de la misma forma los nombres con la misma pronunciación, el cual dejo mucho que desear pero hay que juzgarlo a su medida, sin embargo, se realizaron e implementaron versiones posteriores como *American Soundex* y *Soundex Daitch-Mokotoff.* La recuperación de información eran los nombres más cercanos según las comparaciones de

codificaciones de apellidos almacenados. También existen versiones según el idioma como *Soundex 2 y Phonex, orientados al francés.*

Posteriormente, se desarrollaron algoritmos como *Double Metaphone y Levenshtein* que utilizan reglas de codificación más extensas, añadiendo nuevos caracteres, codificaciones para diferentes pronunciaciones de una sola palabra y técnicas para transformar una cadena en otra.

En la facultad de *Ciencias de la Computación, BUAP,* por parte de los doctores del laboratorio de *Recuperación de Información* [7], hicieron adaptaciones al algoritmo fonético *Soundex* para la codificación de SMS, usada para la representación de textos SMS, la recuperación de información y normalización basada en SMS. Los cambios mejoraron el *performance* y mejoraron el modelo probabilístico, dando mejores posibles traducciones, además de mejorar el *matching* entre los textos SMS y sus correspondientes texto en inglés o en español; uno de sus experimentos, fue aplicar antes un filtro, checar abreviaturas mediante un diccionario creado manualmente, el cual obtuvo mejores resultados.

En [8], proponen un nuevo enfoque de normalización, el cual combina técnicas de traducción léxicas y fonéticas, con algoritmos de desambiguación en dos niveles: léxico y semántico; tratan tanto los símbolos especiales, abreviaturas fonéticas y la desambiguación en dos niveles; lo separan en tres módulos, el primero llamado "pre procesamiento" divide las palabras correctamente, teniendo en cuenta algunas características de los mensajes SMS, como la posible usencia de espacios en blanco, el segundo módulo, llamado "traducción", como su nombre lo dice, obtiene todas las posibles traducciones de las palabras SMS del idioma español a un español convencional, para ello tiene que liderar con tres tipos de palabras: las abreviaturas fonéticas, no fonéticas y palabras reales, del cual para cada tipo se ocupa un diccionario diferente, la salida de este módulo es la unión de las listas de traducciones posibles extraídas desde el diccionario SMS y el diccionario fonético en español; para esto ocupa una modificación de la distancia de levenshtein ponderada, asignando costos dependiendo de la operación a realizar (inserción, eliminación o sustitución de caracteres) y usando un umbral para la selección de mejores traducciones; por último el módulo de desambiguación, el cual elige la traducción correcta para cada palabra entre todas las posibles traducciones dadas por el modulo anterior.

En [9], se trata de manera particular el problema de trabajar con textos tipo SMS. La normalización la ven como un problema de traducción automática del idioma ingles de mensajes SMS al inglés convencional; entonces proponen una adaptación de un modelo estadístico basado en frases. Se argumenta que los SMS son muy diferentes de los textos escritos normales, debido al estilo particular de escritura de éstos, y la alta frecuencia de ocurrencia de términos no estandarizados, usualmente en versiones cortas, resumidas, truncadas, o fonéticamente transliteradas.

Hay muchas abreviaturas y símbolos no estándar en los mensajes SMS y Twitter, para atacar este tipo de problemas de los textos cortos, [10] propone diferentes combinaciones de técnicas de normalizado con el fin de aprovechar las fortalezas que cada uno propone, usan un enfoque para segmentar o dividir palabras en bloques de caracteres en función de sus símbolos fonéticos y aplican modelos de *machine translation* y etiquetado de secuencias a nivel de bloque, además del uso de la técnica de normalización *jazzy spell checker.* Realizan las diferentes combinaciones para juntar la lista de candidatos que arroja cada técnica obteniendo una lista resultante la cual contendrá los mejores candidatos. Los conjuntos de datos utilizados para los experimentos fueron los mismos utilizados en otras investigaciones para comparar los resultados, con base en ello, los resultados mejoran la presión con otros trabajos realizados anteriormente.

En [11], proponen un decodificador de normalización de texto de redes sociales del idioma ingles y chino y después hacen la traducción de un idioma a otro, en trabajos previos se centraron principalmente en la normalización de las palabras mediante la sustitución de una palabra informal a su forma convencional, en este trabajo proponen una mejora, añadiendo el tratamiento de la recuperación de aquellas palabras perdidas que usualmente los usuarios no suelen poner por ejemplo para el idioma ingles omiten mucho el conjugado del verbo "be", además añaden el tratamiento de la corrección de signos de puntuación, es decir, quitan, sustituyen o añaden estos signos.

Para tratar los signos de puntuación ellos proponen un modelo DCRF de dos capas, la primera capa proporciona las etiquetas de puntuación reales: ninguna, punto, coma, signo de exclamación e interrogación; la segunda capa da el limite o borde de la sentencia, es decir, indica si la palabra actual se encuentra al principio de (o dentro) una sentencia declarativa, pregunta o frase.

Para tratar aquellas palabras que a menudo se omiten, proponen un modelo CRF para llevar acabo la recuperación. Entonces, para saber cuál de las técnicas de normalización es bueno aplicar, es decir cual hipótesis es la más óptima, su algoritmo realiza dos sub-tareas, la primera produce nuevas hipótesis a nivel de la sentencia en la pila actual a través de un productor de hipótesis. La segunda evalúa las hipótesis y retiene las buenas, esto se debe a que ocupan una función de característica la cual maneja puntuaciones para saber cuáles son hipótesis buenas y malas, y por último, todas las funciones de característica son combinadas mediante un modelo lineal y obtener el mejor score para una hipótesis dada.

También las técnicas de normalización son ocupadas para textos de un dominio especifico, por ejemplo, los textos médicos en donde hay un gran número de abreviaturas que se utilizan de forma rutinaria a lo largo de dichos textos y la identificación de su significado es crítica para la comprensión del documento, pero es bien sabido que este tipo de abreviaciones o acrónimos pueden tomar diversos significados, por lo que se vuelve un caso especial de la desambiguación de palabras (WSD), entonces es muy importante tomar el contexto como tal; en [12] proponen un enfoque semi-supervisado basado en máxima entropía (ME) para poder normalizar las abreviaturas y acrónimos de textos médicos, al utilizar ME eliminan la ambigüedad del "significado" que pueda tomar una abreviatura; propone dos modelos LCM(Modelo de contexto local) y CM (modelo combo), el primero se construye para el entrenamiento sobre el contexto a nivel de sentencia, el cual toma las dos primeras y las dos últimas palabras que acompañan a la expansión de la abreviatura, y el segundo entrena sobre una combinación de sentencia y contextos a nivel se sección, el cual asigna simplemente el título de la sección en la que se encontró la expansión de la abreviatura. Así entonces el uso de modelos de ME, demostró que para la desambiguación de abreviaturas es una vía prometedora de investigación, así como para implementarla en las tareas de normalizado de textos relacionados con abreviaturas.

La recuperación de información va más allá de una sencilla traducción. Las respuestas a preguntas frecuentes (*FAQ*) pueden ser información recuperada desde una aplicación. Originalmente las *FAQ* se refería a la pregunta realizada con frecuencia en sí misma, y la compilación de estas preguntas se les llamaba *Lista de FAQ.* Más tarde sencillamente *FAQ* hace referencia a la lista y a preguntas que no necesariamente son frecuentes pero si sobresalientes o importantes; tanto que las

FAQ trascendieron de los correos a Internet e incluso a estar presentes en los folletos de artículos de consumo.

En [13] y [14], por ejemplo, se implementa un servicio web que permite consultar una base de datos de *FAQs* escritas en lenguaje Hindi mediante consultas ruidosas *SMS* en idioma inglés. La propuesta consiste en formular el criterio de similitud en el proceso de búsqueda como un problema combinatorio, en donde el espacio de búsqueda consiste de todas las combinaciones de las variaciones del diccionario de los términos de la consulta y sus N mejores traducciones. Lamentablemente el corpus usado en los experimentos no se encuentra disponible y por tanto, no es posible usarlo con fines de comparación.

Con lo anterior el laboratorio de Recuperación de Información de la facultad de ciencias de la computación, BUAP; aplicaron un análisis del uso de técnicas de traducción automática para sistemas de recuperación de información monolingüe, crosslingüe y multilingüe, al trabajar sobre estas tareas, emplearon técnicas de normalización automática, el cual aquellos términos que no estén el vocabulario común, se propone sustituir cada término de la consulta por la traducción más cercana ofrecida por medio de un diccionario estadístico bilingüe, el cual se creó usando pares de frases, alineando los SMS y las preguntas correspondientes de las *FAQs*.

Para encontrar la similitud entre los términos de los mensajes (SMS) y cada una de las preguntas de las *FAQs*, utilizaron la similitud de *Jaccard*.

En la mayoría de los mensajes de textos cortos, normalmente contienen un colección de palabras fuera del vocabulario, lo que sugiere que implementar un aprendizaje convencional supervisado no funcionaría bien debido a la escasez de datos; además muchas palabras mal formadas son ambiguas y requieren de analizar su contexto para así eliminar la ambigüedad. Por ejemplo "*goooood*", puede referirse a "*good*" o "*god*" según sea el contexto. Por lo que [15], optaron por proponer un método el cual no requiera datos de entrenamiento etiquetados, pero que sea capaz de tomar en cuenta contexto para llevar acabo la normalización; En este artículo, se trabaja colectivamente a los casos individuales de errores ortográficos, abreviaturas ad hoc, ortografía no convencional, sustituciones fonéticas y otras causas de desviación léxica como "palabras mal

formadas". Propone un método en cascada para detectar y normalizar palabras mal formadas del idioma inglés.

En primer lugar generan su lista de formas léxicas canónicas que sean candidatas de palabras fuera del vocabulario (*OOV*, por sus siglas en inglés) llamado conjunto confusión, usando un conjunto de estrategias (distancia de edición, reducción de palabras y el uso de del algoritmo *doblemetaphone*) que trabajan en el estudio de la morfología y variación fonética; también, en este paso, realizan una clasificación de los candidatos sobre la base de un modelo de lenguaje de trigramas, entrenando con un conjunto de datos limpios del *twitter*, es decir, *tweets* que se componen de todas las palabras *IV*. Para saber si son *OOV* o *IV* hace uso de un diccionario en inglés.

Luego entonces, detectan si una palabra dada que este *OOV*, es en realidad una palabra mal formada o no, con respecto al conjunto confusión; para esto, primero se obtienen las características basada en dependencia, usando un clasificador *SVM* de *kernel* lineal, el cual lo entrenan con un conjunto de textos cortos limpios de *twitter*, el cual no contiene palabras *OOV*; cada palabra está representada por palabras que están *IV* integradas en una ventana de contexto de tres palabras, el cual contendrá dos palabras y su posición de dependencia entre ellas (palabra1,palabra2,posicion). Para predecir si un candidato es o no una palabra mal formada, se crea un ejemplar por cada uno de los candidatos del conjunto confusión, y se extrae las características de dependencia. Si todos estos candidatos son pronosticados a ser negativos por el modelo, marcan a este como correcto; en otro caso, lo marcan como mal formado, y se pasa todos los candidatos a la etapa de selección de candidatos.

Ya estando en la etapa de selección de candidato, la palabras que son predichas como mal formadas, se selecciona el candidato más probable del conjunto confusión, con las bases de normalización. La selección final de las palabras *OOV* que no están mal formadas, se aplican: distancia de edición léxica, distancia de edición fonética y sub-cadena prefijo, sub-cadena sufijo, y la sub-secuencia común más larga; con estas técnicas se logra capturar la similitud morfofonémica.

La mayoría de los sistemas antes mencionados limitan su alcance de procesamiento a categorías determinadas (por ejemplo, faltas de ortografía, abreviaturas basadas en supresión), o requieren un corpus de anotaciones humanas de gran escala para entrenamiento, lo que dificulta en gran medida la escalabilidad del sistema.

[16] Proponen un sistema de normalización de texto robusto con "amplia cobertura", es decir, para cualquier *token* ruidoso o no estándar creado por el usuario, el sistema propuesto trata de restaurar a su palabra correcta dentro de un top N de candidatos; el cual, integra diferentes perspectivas humanas en la normalización de *tokens* ruidosos, incluyen la transformación de letras mejorado, primado visual y similitud fonética/léxica.

Para esto, proponen el uso de un umbral el cuál indicara que tan amplia es "la cobertura" que se abordara para los posibles candidatos de un *token* ruidoso; a medida que se aumente la cobertura, es decir, crezca el umbral, implicaría que el conjunto de candidatos crecería mucho y sería más difícil elegir el correcto, al igual que si la cobertura es pequeña, tendería a un límite de desempeño indeseable para el planteamiento de *reranking* de candidatos. Ellos proponen abordar el problema de normalización de textos desde una perspectiva cognitivo-sensible e investigan razones humanas por las cuales se comenten estos errores, es decir, sostiene que existen un conjunto de patrones de transformación de *tokens* que los seres humanos usan para descifrar los *tokens* no-estándares.

El uso de primado visual, importante en la comprensión humana de los *tokens* ruidosos, donde, la primera o primeras letras de la palabra sirven como un estimo visual muy importante; por ejemplo, si una persona lee una lista de palabras en la que está la palabra "*table*", y después tenga que a completar una palabra en donde empiece con "*tab*", es muy probable que responda "*table*", ya que la persona es primada.

La "transformación de letras mejorado", cabe destacar lo de mejorado ya que integran dos aspectos novedosos: primero, un conjunto de características basadas en límite de fonema, silaba, morfema y palabra que caracterizan efectivamente el proceso de formación de los *tokens* no-estándar, usan BILOU para llevarlo a cabo; posteriormente usan el modelo de canal ruidoso para obtener los candidatos; Y entrenamiento de selección de pares (*word, token*) con base en su contexto.

Para la normalización con base en similitud fonética/léxica usan Jazzy, el cual ya integra *DobleMetaphone* y distancia de *Levenshtein* para esta tarea.

Al aplicar estas tres perspectivas a un *token* arbitrario ruidoso, independientemente, cada una de ellas sugieren sus N candidatos más confiables.

Los candidatos considerados para cada *token* ruidoso lo evalúan a nivel de palabra y de mensaje; A nivel de palabra, proponen dos estrategias de combinados heurísticamente, primero combinar los tres sistemas de normalizado, entonces se tiene 3N candidatos; el segundo solo toma N candidatos en total, dando prioridad a las técnicas que obtuvieron mayor precisión (trasformación, corrector ortográfico y primado visual). También llevan a cabo la evaluación a nivel de mensaje, donde la información de contexto local es utilizada para seleccionar el mejor candidato.

2.2. Conclusiones del capítulo.

En este capítulo se abordaron las diferentes propuestas más relevantes por diferentes investigadores, los cuales permiten tener una visión general del panorama de las diferentes técnicas empleadas, la mayoría de ellas orientadas solo para el idioma inglés, de aquí la importancia de poder enfocarlos al idioma español.

De lo que se resalta, es que muchos investigadores hacen uso de más de una técnica de normalización, además de que es muy importante primero poder localizar aquellas palabras OOV, técnicas tanto para transliteraciones fonéticas, léxicas como semánticas, la combinación de una con otra se observó que se obtiene un grado más de precisión en la posible interpretación de un mensaje corto.

Con esto nos damos cuenta que el uso de una sola técnica de normalización no es suficiente para obtener una mejor traducción del mensaje corto, además de que se debe saber cuándo usar una técnica de otra.

Muchos investigadores, se fueron de lado de la Fonética, argumentando que el ser humano, tiende a escribir tal y como lo dice, al referirse con ser humano, se orienta más de lado de la población más joven, aquella la cual le falta ese grado de madurez, hablando de la escritura, de la cual no les importa como lo escriban siempre y cuando se den a entender.

Pocos artículos, mencionan el proceso de reducción de posibles candidatas, esto es un paso crucial ya que el poder reducir en lo mínimo mi conjunto candidato, se logra un mejor desempeño en cuanto a tiempo de ejecución.

CREACIÓN DE UN CORPUS PARALELO DE MENSAJES CORTOS EN EL IDIOMA ESPAÑOL.

En este capítulo, se expone como se creó un corpus paralelo en español, el cual servirá para llevar acabo los diferentes experimentos de propuestas planteadas para la normalización de textos cortos. Se detalla el sistema creado para dicho propósito.

3.1. Planteamiento de creación del corpus paralelo

Se creó un corpus paralelo de 2500 mensajes SMS. Para ello se planteó crear un sistema para el público en general el cual permita mandar SMS gratis a números TELCEL, y así poder recaudar los mensajes SMS enviados.

3.2. Estructura del sistema

Como se necesita crear un corpus paralelo, se debe obtener dos corpus, uno que contenga los mensajes enviados vía SMS, y otro el cual serán los mensajes correctamente escritos, este último Será creado de manera manual, por la persona a cargo, se observara cada mensaje SMS, y se procederá a escribirlo con palabras *IV.*

Entonces, el sistema solo se encargara de crear el corpus de los mensajes SMS, debe pedir el SMS escrito tradicionalmente como si lo mandaran de un celular, es decir, con errores ortográficos, tipográficos, omisión y sustitución de palabras, abreviaciones, etc. El mensaje debe ser guardado en nuestra base de datos y posteriormente enviado a la persona destino. En la Figura 2 se muestra el sistema gráficamente y a continuación se describe cada módulo del sistema de envío de SMS gratis.

- **Modulo Cliente**. Se creó una página WEB con un domino especifico (www.enviarsmsgratis.zapto.org), se describía el ¿por qué? de la página, la confidencialidad de sus datos, que solo era de dominio académico. Y posteriormente el módulo de envio de mensaje, en el cual se le pide al usuario el mensaje y numero para poder enviarlo.

  Para esto, se hizo uso de varios lenguajes de programación; HTML y CSS para la estructura y diseño de la página web, al mismo tiempo que JAVASCRIPT para la verificación del lado del cliente.

- **Modulo Servidor**. Se encarga de verificar nuevamente que haya ingresado el mensaje y numero correctamente, posteriormente se almacena en una BD creada en MySQL haciendo esta comunicación con PHP, un lenguaje de programación de lado del servidor. Además es el encargado de ejecutar una serie de comandos AT los cuales me permitan solicitar al modem el envío del SMS al número proporcionado por el cliente, para esto usamos PYTHON que nos permitió ejecutar estos comandos como si estuviéramos desde consola.

- **Modulo Modem**. Realiza la tarea de envío del SMS. Se hizo uso de un modem Cinterion modelo TC65, el cual fue proporcionado por el laboratorio de Recuperación de Información de la Facultad de Ciencias de la Computación[5]. Con este modem y el uso de un chip se pudo llevar acabo la comunicación del envío.

[5] www.cs.buap.mx

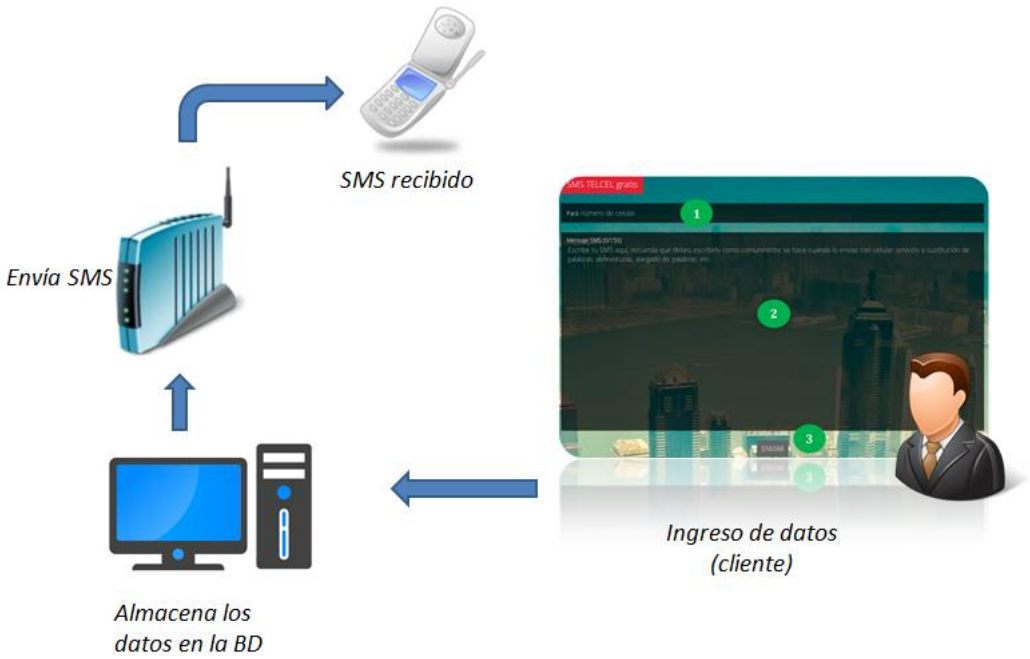

Figura 2. Sistema encargado de la creación del corpus paralelo

3.3. Diagramas del Sistema

En esta sección se describen algunos diagramas que son usados en ingeniería de software para la explicación y uso del sistema creado, en nuestro caso el de enviar SMS gratis a cualquier número TELCEL. Se describen de manera rápida cada uno de ellos, no se hace tanto énfasis en los diagramas del sistema ya que solo esto es una parte de lo que abarca esta investigación.

3.3.1. Diagrama de casos de uso

En la Figura 3 se muestra gráficamente los casos de uso usados en este sistema. En realidad solo se explicara el caso de uso enviar SMS, el cual caso principal y de interés del sistema.

Caso de uso: Enviar SMS. El usuario que es cualquier persona del público en general que quiere enviar un SMS, lo único que tiene que hacer es entrar a la página e ingresar los datos que se piden: **Escribir el número** de celular a la persona a la que envía; **Escribir SMS,** este mensaje, como se explica en la sección de arriba, es escrito como se escribiría desde un celular.

Por último enviar SMS, es cuando el usuario da clic en el botón para el envío del SMS, así se comunica con el servidor y este envía la solicitud al modem.

En cuanto a los casos de uso por parte del administrador, no se explican, ya que estos casos el encargado solo monitorea el proceso de llenado y de envío de SMS sean correctos.

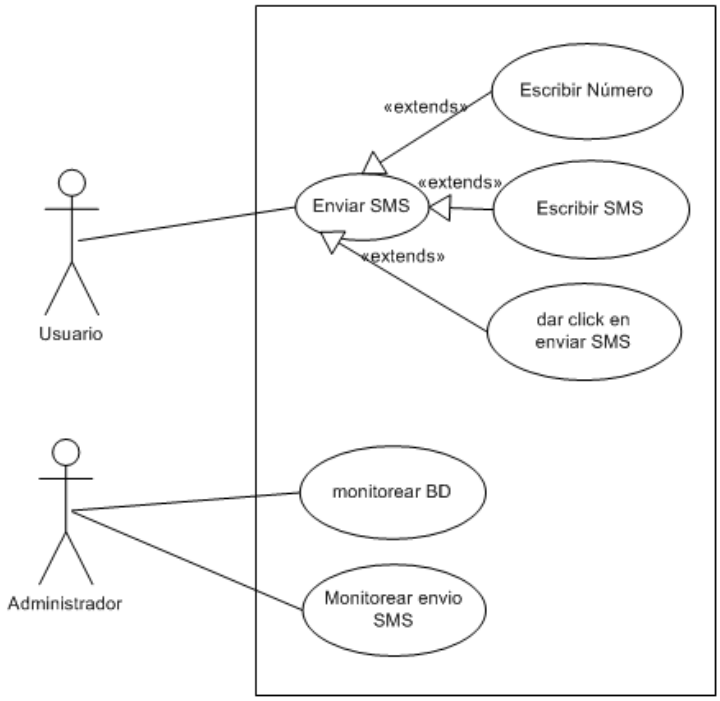

Figura 3. Diagrama de Casos de Uso del Sistema de Envío de SMS Gratis

3.3.2. Diagrama de Entidad /Relación.

En la Figura 4 se muestra la única entidad con sus atributos o visto desde BD la única tabla, como solo nos interesa que almacene los mensajes sin hacer otras tareas, solo se necesitó una tabla, los datos que se guardan son los mismos que se piden para el envío del SMS, excepto el número, es decir, como no interesa a quien envía, el número no sirve. Se agregó un campo el cual es un INDEX que nos permite llevar el conteo automático de la cantidad de SMS almacenados en la BD.

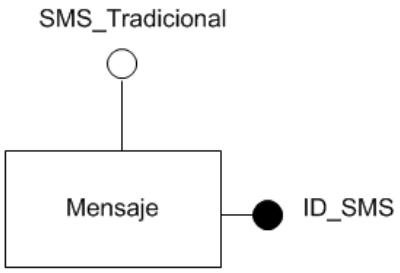

Figura 4. Diagrama Entidad/Relación del sistema de envió de SMS

3.3.3. Diagrama de Actividad

Para tener más claro el sistema, se muestra otro diagrama, en este caso el de actividad, el cual explica de otra forma como el usuario puede mandar un SMS desde el sistema. Se observa en la Figura 5 dicho diagrama.

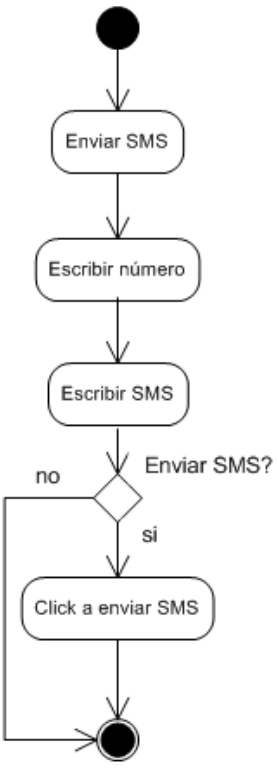

Figura 5.Diagrama de actividad "enviar SMS"

3.3.4. Diagrama de secuencia

En la Figura 6 se muestra el diagrama de secuencia de envío de un SMS, en este se observa como interactúa el usuario con el sistema, y al mismo tiempo la interacción de los diferentes componentes del mismo.

Se puede ver, que tanto del lado del cliente como del servidor se hace la verificación de los datos, para posteriormente almacenarlos en la BD, después el servidor ejecuta los comandos correspondientes para la comunicación con el modem, el cual se encargará de enviar el SMS con el número proporcionado por el cliente. Este diagrama es en el caso de que todos los datos sean ingresados de acuerdo a lo que se pide. Si alguno de ellos no es ingresado correctamente, dependiendo del campo se muestra el mensaje de error. A continuación se describe más detalladamente cada uno de los posibles errores que arrojaría en caso de datos incorrectos.

- **Numero de celular:** debe cumplir en tener 10 dígitos, además que sean puros números, si no cumple con alguna de ellas arroja el error correspondiente.
- **SMS:** Debe tener mínimo dos palabras en el mensaje.
- **Otro:** Cuando se da clic en el mensaje de enviar, hace la verificación de todo de nuevo. Ahora del lado del servidor.

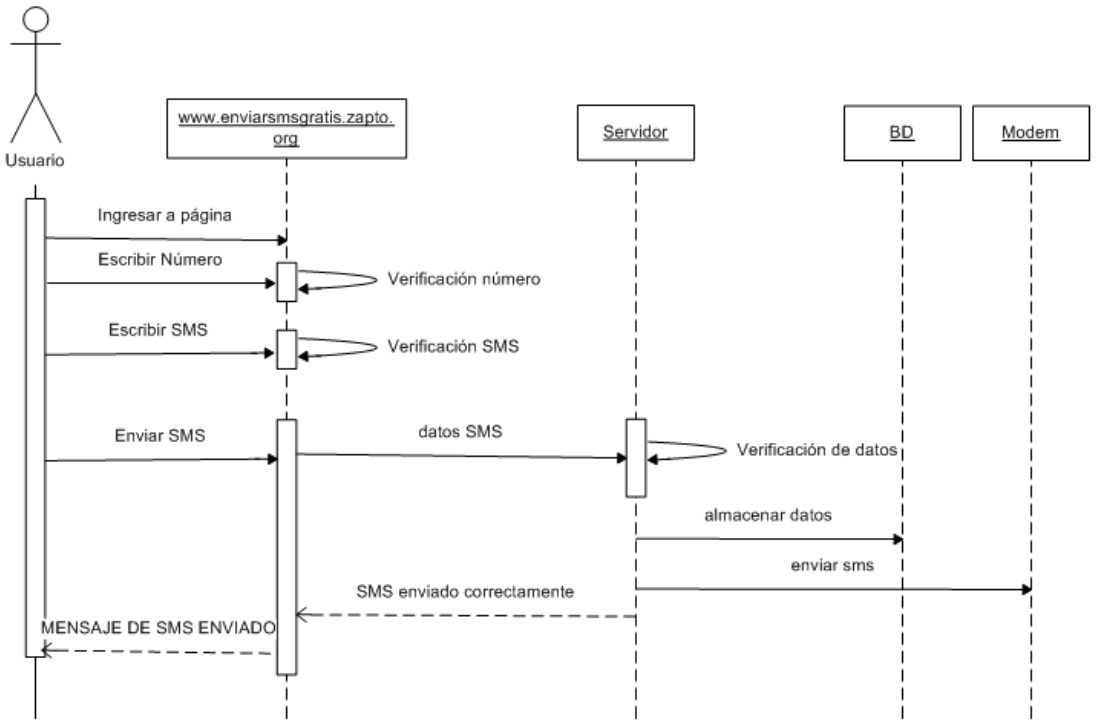

Figura 6. Diagrama de secuencia "envio de SMS"

3.4. Conclusiones del capítulo.

Con la creación de este corpus paralelo de mensajes SMS, se podrán llevar acabo las diferentes pruebas para obtener los métodos o técnicas que mejores resultados arrojen. También como no se cuenta con un corpus de tal magnitud y del idioma español de México es un gran aporte para futuras investigaciones.

Se observa que, la idea que se propuso favoreció tanto para la investigación, como aquellas personas que apoyaron en la creación del corpus, ya que se beneficiaron en un tiempo para enviar SMS gratis a cualquier número TELCEL.

Hacer énfasis, de porque no se pidió también el mensaje correctamente escrito, y así no hacer la parte de corpus limpio manualmente. Se probó un mes de

los tres que corrió el sistema. El cual no se obtuvo buenos resultados, en ese mes se logró juntar una cantidad de 200 mensajes, la idea era que proporcionaran los usuarios, tanto el mensaje escrito SMS como el bien escrito, así como datos de rango de edad y sexo. Por el bajo porcentaje de mensaje, se optó por quitar estos campos y únicamente pedir el número y mensaje a enviar.

Con este corpus ya completado, sirvió para observar los diferentes modos de escritura de las personas, se observa que hay escritura que alcanza hasta un 80% de palabras *OO.* Así como mensajes que no tenían ni un solo error de ortografía. Por supuesto estos últimos fueron desechados.

ASPECTOS LINGÜÍSTICOS AL TRATAMIENTO DE MENSAJES SMS.

En este capítulo, se expone los diferentes aspectos lingüísticos el cual está explícito en la formulación de mensajes cortos, por ejemplo: aspectos fonéticos, abreviación de palabras, sustitución y omisión de palabras, etc. Todo esto crea una nueva forma de escritura que no respeta las normas, tanto ortográficas como gramaticales con respecto al idioma español estándar.

El objetivo es comprehender mejor que mecanismos de transcripción aplicar para llegar a tener una mejor normalización del mensaje, por lo que primero se analizan los diferentes procesos lingüísticos que me permitan llevar una mejor orientación.

Se enfoca el estudio a los mensajes obtenidos del corpus de SMS creado en lenguaje español.

Este capítulo fue basado de una tesis realizada por Anissa, una alumna de la Universidad de Paris 13, basada en el estudio de mensajes SMS de los estudiantes de dicha universidad [1]. La cual principalmente se relaciona con el idioma francés, pero engloba a más lenguas, de ahí que los diferentes procesos se adaptaron para el idioma español.

Dicho estudio se basó de un trabajo realizado por Jacques Anis, el cual trata de las diferentes técnicas que describen los tipos de formación de las unidades de texto de mensajes SMS y abreviación tenemos:

- Reducción a un esqueleto consonante.
- Reducción de variantes fonéticas.
- Reducción de silabo-gramas y en jeroglíficos.
- Sustitución y simplificación de diagramas y de trigramas.
- Truncamiento en apocope o aféresis.

Cada una se muestra a continuación de forma más detallada y con un ejemplo representativo de dicha técnica empleada en el mensaje. Después se mencionan algunas particularidades morfo-léxicas.

4.1. La neografía.

Representa una nueva forma de escritura usando modificaciones distintivas y que han adquirido una forma ya establecida. De origen griego neos que significa nuevo y grafía que significa escritura.

La neografía, representa una palabra o expresión, que desde el punto de vista de la ortografía, representa lo contrario a la manera usada en el lenguaje base, en nuestro caso el idioma español de México.

Jacques Anis, la define de la siguiente manera:

La neografía es un término que se utiliza para designar, sin juicio de valor, ni positivo ni negativo de los gráficos que se apartan deliberadamente de la norma ortográfica. Deliberado por la relevancia de los procesos tales como la abreviatura, la simplificación fonética, la transcripción lejos del idioma tradicional (francés).

La neografía se presenta en diferentes formas que se tratan a continuación.

4.1.1. La neografía fonetizante

Es un proceso que permite un efecto de oralidad, es decir, un gráfico reducido a simples formulaciones fonéticas. Así que la comunicación escrita, en nuestro caso mensajes cortos, se realiza usando una escritura fonética, remplazando silabas y letras, además de acortar palabras a su más simple expresión.

4.1.1.1. Quitar <e> inestables

La limitación de un mensaje corto, hacen que sea necesario la brevedad y la extrema simplicidad de la expresión, por lo que ha hecho que desaparezcan letras no pronunciadas, en este caso la letra <e> que se encuentra al final de algunas palabras. Se observa en la Tabla 1, algunos ejemplos comunes.

Escritura SMS	Escritura Normal
est	este
desd	desde
t	te
qed	quede
cuidat	cuídate
tard	tarde
dejat	déjate
salt	salte
cambiat	cámbiate

Tabla 1. Ejemplos de omisión de la letra <e>

Cabe mencionar que, no solo hay omisiones al final de las palabras, sino en cualquier posición de la palabra donde haya una letra <e>, por ejemplo: "dbrias" significa: deberías.

4.1.1.2. La sustitución de <k> a <c> y a <q>

Se observó en los mensajes, presencia de un cambio común de la letra <k> representando tanto a la letra <q> como <c>, aunque no reduce el tamaño de la palabra cuando se aplica sobre la consonante <c>, es muy común remplazar a ésta. Por lo contrario, cuando hace el remplazo por la letra <q>, viene acompañado con la vocal <u>, así que se reduce una letra menos con este cambio. Se observan algunos ejemplos en la tabla 2.

Escritura SMS	Escritura Normal
kasa	casa
ke	que
K	que
saka	saca
aki	aquí
kon	con
musik	música
kieres	quieres
komo	como

Tabla 2. Ejemplos de remplazo de la letra <c, q> por <k>

4.1.1.3. La reducción junto con compactación

Representa aquellas palabras condensadas, ya sea por palabras ya establecidas o palabras fonéticas, también por la desaparición de guiones que representan la formación de neo-gráficos por la combinación de dos o más unidades léxicas. Algunos ejemplos encontrados en los mensajes recopilados se muestran en la tabla 3.

Escritura SMS	Escritura Normal
tqm	te quiero mucho
FCC	Facultad de ciencias de la computación
xq	por que
tq	te quiero
k	que
pal	para el
ntc	no te creas

Tabla 3. Ejemplos de palabras reducidas con compactación

4.1.1.4. Sustitución de <i> a <y> y viceversa

La sustitución de <i> y <y> y viceversa muy vista en los mensajes SMS, las personas que escriben estos mensajes, suelen considerar el mismo sonido a <i> por <y> y viceversa, lo que lleva de manera voluntaria al cambio de estas letras. Se observan en la tabla 4 algunos casos.

Escritura SMS	Escritura Normal
hoi	hoy
fuy	fui
voi	voy
i	y
ia	ya
oiie	oye
mui	muy
kari o kary	Karina

Tabla 4. Ejemplos de sustitución de la letra <i> por <y> y viceversa

4.1.1.5. Omisión de la letra <h>

Como sabemos en el idioma español la letra <h>, representa una letra muda en muchas palabras, por lo que en muchos mensajes, tienden a suprimirla para reducir espacio. Observando algunos casos en la tabla 5.

Escritura SMS	Escritura Normal
ola	hola
oi	hoy
asta	hasta
ermanito	hermanito
oy	hoy
acer	hacer
aya	haya
as	has
ubo	hubo

Tabla 5. Ejemplos de Omisión de la letra <h>

Cabe mencionar que en otros casos, la <h> junto a otra consonante, toma un sonido único, por ejemplo junto a la consonante <c>, quedaría <ch>, donde se suele reemplazar ambas letras por una <x>. Un ejemplo seria la palabra <chido>, escrita en mensajes cortos como <xido>.

4.1.2. La variación de palabras

En el corpus se observó que las unidades léxicas que son transcritas, tienen diferentes usos dependiendo del que la escribió. Sin embargo hay personas que no utilizan siempre los mismos gráficos. Por ejemplo <k>, <ke>, <qu> y <q> que representan la unidad léxica <que> en el español tradicional.

Este aspecto sucede entre personas que se conocen, es decir, el que recibe el mensaje debe conocer el modo de escritura de la persona emisora del mensaje, y así poder dar la mejor interpretación. Otro caso, que la persona tenga ya un amplio conocimiento de interpretar el mensaje.

4.1.3. Polivalentes y Polisemia

Este es un procedimiento o el mismo signo grafico a la posibilidad de tener múltiples lecturas. Las letras individuales permiten la transcripción de dos o más unidades con diferentes significados. En la tabla 6 se observan algunos casos.

Hay casos donde, una palabra escrita en lenguaje SMS, suele considerarse una preposición o una conjugación de un verbo, para poder diferenciar, se debe tomar en cuenta el contexto para poder dar la mejor transcripción del mensaje. Otro caso, es cuando se suprime aquella vocal que te puede decir si esa palabra es de categoría masculina o femenina, es decir, <ls>, representa un pronombre, lo que hay que saber a cuál <las, los, les>, para esto de igual mente se debe saber de qué trata el mensaje y así dictaminar cual es la mejor propuesta de transcripción. Por ultimo está el caso de representar por un digito o numero una palabra, suele ocurrir que se puede tomar más de una posible transcripción.

Escritura SMS	Escritura Normal
aya	alla, haya
xq	porque, por que
d	de, da, di
l	la, el, lo, le
ls	los, las, les
1	uno, una
+	mas(palabra) o signo
-	menos(palabra) o signo
cuant	cuanto, cuanta

Tabla 6. Ejemplos de Polivalentes y Polisemia

4.1.4. Logogramas

Los logogramas conciernen las transcripciones enteras de una palabra por uno o más signos. Suele tomarse más tiempo de transcripción para personas que tienen conocimiento de este tipo de lenguaje en mensajes cortos. Algunos casos comunes mostrados en la tabla 7.

Escritura SMS	Escritura Normal
sl2	saludos
srt	suerte
sl	solo
esta2	estados
1	Una
Est3	Este
1s	unos
1er	Primer
Alg1	Algún
100tos	cientos

Tabla 7. Ejemplos de Logogramas

También, en el análisis se observa que se hace el uso de palabras que son reducidas a sus letras iniciales, algunos ejemplos en la tabla 8. Se observa que escribir solo la letra inicial conlleva a la polisemia.

Escritura SMS	Escritura Normal
v	ve, va, ves
t	te, tu
q	que
m	me, mi
d	de
s	Se, si
l	la, el ,al, le

Tabla 8. Ejemplos de palabras reducidas a letras iniciales

4.1.5. Esqueleto consonante

Se refiere a palabras que están formadas solo con consonantes, esto se manifiesta debido a economizar el número de letras a utilizar. También resaltar que en la mayoría se mantiene la primera y última consonante de la palabra cuando es una palabra corta.

Jacques Anis, menciona que las consonantes juegan un papel más importante en comparación a las vocales. Se ha sabido por mucho tiempo gracias

a la teoría de la información, las consonantes tienen un valor informativo más fuerte que las vocales.

En la tabla 9, se conservan algunos casos, en donde palabras se transforman radicalmente a solo consonantes, este proceso de quedarse solo con consonantes, va más a palabras que se podría decir, que ya están como un escritura estándar para este lenguaje.

Escritura SMS	Escritura Normal
mnn	mañana
cn	con
q,k	que
x	por
vcs	veces
msj	mensaje
bss	besos
tqm	te quiero mucho
vz	vez

Tabla 9. Ejemplos de palabras formadas de consonantes

4.1.6. Siglas de sintagmas preposicionales o incluso de las expresiones escritas

Estas son formas comunes de palabras o de expresiones con mayor número de coincidencias en los mensajes del corpus. Se podría decir que son la base de este nuevo lenguaje de comunicación escrita. Algunas de ellas se observan en la tabla 10.

Escritura SMS	Escritura Normal
xq	porque
q	que
x	por
msj	mensaje
tqm	te quiero mucho
t	te
kasa	casa
djo	dejo
vz	vez

Tabla 10. Ejemplos de palabras comunes en los mensajes cortos

4.1.7. Técnicas de jeroglíficos

Definimos como jeroglífico, como una mezcla de signos, dígitos y letras para el remplazo de una palabra por el valor fonético de sus sonidos. Es uno de los tipos de los cuales cuesta dar la mejor transcripción posible. Algunos ejemplos se observan en la tabla 11.

Escritura SMS	Escritura Normal
sl2	saludos
100tos	cientos
1ro	primero
Esta2	estados
2da	segunda
=	igual
seg1	según
1s	unos
=mente	igualmente

Tabla 11. Ejemplos de jeroglíficos

4.1.8. Heterogeneidad

Como se observa claramente en la mayoría de los ejemplos antes mencionado, engloba a más de un proceso, es decir, hay combinaciones que forman a los tipos de palabras encontradas en los mensajes cortos.

Escritura SMS	Escritura Normal	procesos
sl2	saludos	jeroglífico + reducción + logograma
oi	hoy	sustitución + reducción
tqm	te quiero mucho	logograma + reducción+ compactación
k	que	sustitución + reducción
q	que	reducción + logograma
ntc	no te creas	reducción + compactación + logograma

Tabla 12. Ejemplos de combinación de procesos

4.2. Particularidades morfo-léxicas

Enfocado a la correcta escritura del mensaje, donde en el caso de SMS, no respetan este tipo de reglas. También se engloba aquellas palabras que por el contenido de ellas, representan un estado de ánimo, tienden a ser deformadas, del mismo modo la combinación de caracteres tanto alfanúmero como no, forman una nueva palabra la cual representa un estado de ánimo.

4.2.1. Errores de transcripción ortográfica

Los mensajes cortos (SMS), no obedecen a las normas ortográficas del idioma español tradicional, por lo que sucede a menudo errores de este tipo, de los cuales se dividen en dos tipos:

- Errores espontáneos: Se conocen como errores tipográficos, o mejor conocidos como "errores de dedo", fuertemente relacionados por el tamaño del celular, la presentación del mismo teclado o por la rapidez de escribir el mensaje.
- Errores voluntarios: son con respecto a que se escribe la mayoría de veces como se pronuncia la palabra, no se hace uso correcto de las normas ortográficas.

Algunos casos, se observan en la tabla 13. Destacar que puede ser considerada la escritura de la palabra por ambos tipos de error (idea propia), debido a que ambos encajan en algunas palabras, ya sea por presionar ambas teclas al mismo tiempo, escribir rápido o por instinto.

Escritura SMS	Escritura Normal	Tipo de error
ola	hola	Voluntario
uqe	que	Espontaneo
ceanr	cenar	Espontaneo
benir	venir	Voluntario
saka	saca	Voluntario
srte	suerte	Espontaneo | Voluntario
dja	deja	Voluntario | Espontaneo
llehas	llevas	Voluntario
dpositar	depositar	Voluntario | espontaneo

Tabla 13. Ejemplos de tipos de errores

En la mayoría de los mensajes, es común omitir los acentos, en parte que hay celulares que su teclado están en codificación del idioma inglés, por lo que los acentos no vienen; otra es porque un acento ocupa un carácter de más, que conlleva a reducir un espacio el tamaño libre de escritura.

4.2.2. Signos de Puntuación

Son los signos que marcan los descansos o aclaran el sentido o la modalidad de un enunciado, de los cuales están los puntos, comas, signos de admiración e interrogación, etc. Su omisión de éstos ocurre muy frecuente en los mensajes cortos, son signos de menor importancia y que para el usuario representaría reducción de mi tamaño del mensaje.

En otros, casos suelen ocuparse para representar estados de ánimo.

4.2.3. Interjección

Se refiere a una palabra o expresión, la cual expresa por si sola un estado de ánimo o reacción emocional de la persona en ese momento. En este caso, en el momento que está escribiendo el mensaje.

Escritura SMS	Escritura Normal
holaaaa	hola
te amus!!	te amo
tqm	te quiero mucho
chingona!!	chingona
ammm	ammm
Jajajaja	jaja
Mmm	mmm
Haaaaa	ha
Heee	he

Tabla 14. Ejemplos de palabras que representan algún estado de ánimo

4.2.4. Onomatopeya

Son aquellas palabras alargadas debido a rasgos de características de oralidad. Representan diferentes emociones. Algunas palabras más frecuentes, se observan en la tabla 15. Se puede ver que hay palabras que no pertenecen al vocabulario estándar aun si se transcribiera, son expresiones que su contenido representa una emoción en ese momento.

Escritura SMS	Escritura Normal	Tipo de emoción
Holaaaa	hola	Felicidad
mmmm	mmm	Pensativo
Uffff	uf	Aliviado
auchss	auchs	desconcierto
Ammm	amm	Pensativo
felizzzz	feliz	Felicidad
Jajajaj	Jaja	Felicidad
nuevooo!!	nuevo!	Felicidad
ahhhh	he	Tristeza

Tabla 15. Ejemplos de Onomatopeyas

4.2.5. Truncamiento

Se aplica mucho en mensajes cortos, así se compacta más el mensaje y economiza para no sobrepasar el límite permitido en cuanto al número de caracteres permitidos.

Existen dos tipos de truncamiento:

- La eliminación de las primeras letras se llama truncamiento aféresis.
- La eliminación de las últimas letras se llama truncamiento apocope.

En la tabla 16. Se muestran palabras que tienden a aplicar algún tipo de truncamiento.

Escritura SMS	Escritura Normal	Tipo truncamiento
Ola	hola	aféresis
Prueb	prueba	apocope
Encontrat	encontraste	apocope
Tard	tarde	apocope
Stas	estás	aféresis
Libramieent	libramiento	apocope
Stancia	estancia	aféresis
Fuist	fuiste	apocope
Qed	quede	apocope

Tabla 16. Ejemplos de truncamiento de palabras

4.2.6. Emoticones

Es un neologismo que proviene de emoción e icono, es una secuencia de caracteres ASCII para formar diversas expresiones en forma imágenes, comúnmente caritas las cuales muestran una emoción. Hay desde emociones positivas, como negativas. Algunos emoticones más conocidos se muestran en la tabla 17.

emoticón	Emoción
:)	Alegre
;)	Alegre
:(	Triste
O.O	Sorprendido
¬¬	Pensativo
n_____n	Feliz
:-$	Enfermo
^-^	Feliz
:'(	Triste

Tabla 17. Ejemplos de emoticones

4.2.7. Letras capitales y supresión de ellas.

Son las letras mayúsculas de inicio de un párrafo o que va después de un punto. En los mensajes cortos, cuando se escribe un mensaje SMS, inicialmente automáticamente pone la primera letra en mayúscula, lo que conlleva a que la mayoría de mensajes contiene la primera letra de esta forma, pero hay quienes

suprimen éstas, puede explicarse que ocurre debido a una dificultad de dactilografía o incluso por la falta de dominio del celular.

4.3. Conclusiones del capítulo.

En este capítulo se observaron más a detalle la escritura empleada en los mensajes SMS. De la cual se observa que es una escritura dinámica, cada usuario puede dar más de un significado a una palabra o escribirla de diferentes maneras. Resaltar que en la mayoría de procesos aplicados, se realizan para economizar en lo más posible el tamaño disponible y no sobrepasarlo del mensaje permitido, pero entre más compacto sea el mensaje se hace más difícil de interpretar para personas que no estén familiarizados con esta escritura, lo que lleva a que el que recibe el mensaje debe conocer la forma de escritura del emisor, para dar la mejor interpretación del mensaje.

Conociendo los diferentes procesos lingüísticos que realizan las personas al escribir un mensaje, se obtiene un mejor dominio para aplicar técnicas que permitan revertir este proceso y obtener la normalización que más se apegue a lo que da a entender el mensaje en un formato que aplique las normas ortográficas y gramáticas del idioma español estándar.

Las diferentes formas lingüísticas que se presentaron, fueron elegidas de entre otras, tratadas de la tesis con la que se apoyó en este capítulo. Debido que en el idioma francés, tienden a presentarse de gran ocurrencia así como en el idioma español. También se añadieron otras observaciones vistas en el corpus creado, donde son más particulares al idioma español, como la ausencia de la letra <h>.

Cada una de ellas se abordaron, mediante diferentes técnicas de normalizado, para así llevar el proceso de traducción a su correcta forma escrita a palabras *IV.*

Algunos procesos coinciden al describir el tipo de palabra, se puede decir, que es otra forma de etiquetar a ese tipo de palabras, el punto, es observar la transformación de la palabra y la posible transcripción, en este caso se muestran las ideales, pero más adelante se verá que no siempre se logra obtener la candidata ideal que remplace a la palabra escrita en lenguaje de mensaje corto.

MODELO DE NORMALIZACIÓN DE TEXTOS CORTOS

En este capítulo se establecen los diferentes procesos consecutivos implementados para representar el modelo propuesto, el cual, viene dado por un conjunto de técnicas léxicas y sintácticas, las cuales se van a describir más a detalle a continuación. También se mencionan los recursos léxicos que apoyaron a mejorar la propuesta.

5.1. Recursos Léxicos

Antes de empezar a describir el modelo, primero se hace énfasis de recursos léxicos utilizados como un proceso antes de aplicar las diferentes técnicas léxicas y sintácticas.

5.1.1. Vocabulario de palabras

Se creó un diccionario de palabras en el idioma español de aproximadamente 111,353 palabras.

5.1.2. Diccionarios de emoticones y abreviaciones comunes.

Se creó un diccionario de abreviaciones de un total de 73 palabras, se podría decir que "establecidas" o más comunes, en este tipo de mensajes, y que nuestro modelo no fue capaz de expandirla a su forma correctamente escrita. Entre las que se encuentran son abreviaciones que representan a más de una palabra, ejemplo: <ntc>, su transcripción seria <no te creas>. Otras, son abreviaciones de un porcentaje menor o igual al 50% de la transcripción correcta, ejemplo: <oax>, su correcta escritura es <Oaxaca>.

Un diccionario de emoticones con un total de 599, los cuales también contiene su descripción representativa [17].

5.2. Creación del índice de palabras

Trabajar con todo el vocabulario recopilado, por cada palabra de un mensaje corto, consume mucho tiempo y trabajo; pensando ello, se dan dos propuestas de crear un índice con estas palabras, haciendo que el tiempo y

trabajo se vea considerablemente reducido. A continuación se menciona cada una de estas propuestas:

5.2.1. Índice basado en trigramas

Se necesita hacer una reducción de mi campo de palabras candidatas, para esto, como primera propuesta, se hace uso de n-gramas de caracteres, en específico trigramas.

Definición Formal de n-grama [18]: Sea una secuencia S de elementos ordenados $s_1 s_2 s_3 \dots s_k \dots$. Se denomina n-grama a cualquier sub secuencia $A = s_{i+1} s_{i2} \dots s_{i+n}$ donde i es un valor entre 0 y $|S|$-n para garantizar que la longitud de A sea siempre n o lo que es lo mismo $|A| = n,\ n > 1$. Entonces, definimos a trigramas como una sub secuencia de tres caracteres consecutivos. El número de trigramas por palabra corresponde a $n - 2\ trigramas\ donde, n = numero\ de\ carcateres\ de\ dicha\ palabra.$

En la tabla 18 se puede observar algunos ejemplos de palabras con sus correspondientes 3-gramas.

Palabra	Trigramas
Hola	Hol + ola
Mañana	Mañ + aña + ñan + ana
Escuela	Esc + scu + cue + uel + ela
Camara	Cam + ama + mar + ara

Tabla 18. Descomposición en trigramas de caracteres

Se crea un índice de trigramas invertido con todo el vocabulario, asignando a dicho trigrama, el conjunto de palabras que lo contienen. Ejemplo: trigrama <aut>: se relaciona con las palabras <auto, pauta, flauta, autorizo, autónomo, etc.>. Teniendo esto en cuenta, se crea un archivo con 5,204 trigramas generados por el vocabulario.

Así, si buscamos una palabra por ejemplo: 'hola' se generarían dos trigramas 'hol' y 'ola', por lo que el total de palabras que engloban ambos trigramas es de un total de 234, por consiguiente un mensaje que contenga esta palabra, trabajaría sobre este número y no sobre todo el vocabulario.

Una desventaja de índice basado en trigramas, dado que en mensajes cortos hay muchos errores, mencionados en el capítulo anterior. Es común que escriban variaciones de una palabra escrita correctamente, visto mejor con el ejemplo de la palabra 'hola', variaciones como 'ohla', la cual genera dos trigramas 'ohl' y 'hla', que al buscar en nuestro índice nos regresa un conjunto de palabras, de entre las cuales no se encuentra 'hola'. Así, al pasar a la siguiente fase no tendrá oportunidad de remplazar 'ohla' por 'hola'.

Aquellas palabras de longitud menor a 3 caracteres, se optó por tenerlas representadas por un índice, etiquetadas por <long>.

Cabe mencionar que se hicieron variaciones a estos índices, por ejemplo sacábamos los trigramas solo al cuerpo consonántico, pero no se obtenía mejores resultados. Del mismo modo se trabajó con bigramas, pero aumentaba el número de candidatas y por consiguiente el tiempo de ejecución, y los resultados no mejoraban mucho.

5.2.2. Índice basado en fonética.

Otra propuesta de índice, fue basado en fonética, como se mencionó en el capítulo 2, en el estado de arte, se habla del uso del algoritmo fonético Soundex, el cual es basado en fonética para el idioma Ingles, y trata, en asignar números del 0-6 a un conjunto de consonantes que suenan de forma similar. (Cero, para las vocales, 'y', 'h' y 'w'). Se consideraba solo tomar los primeros 4 dígitos generados al sustituir por su correspondiente, y los que no llegaran a esa longitud, se rellenaban con ceros [19]. En la tabla 19, se observa el código fonético soundex original para el idioma inglés.

Código	Letras
1	BFPV
2	C,G,J,K,Q,S,X,Z
3	D,T
4	L
5	M,N
6	R

Tabla 19. Código Fonético Soundex Inglés

Lo que se propone primero es un código fonético nuevo para el idioma Español, en [20] hacen una propuesta, de la cual sirvió para basarse, del mismo modo al observar el comportamiento en los mensajes cortos, se crea un código fonético, visto en la tabla 20.

Código	Letras
0	AEIOUH
1	CKQSXZ
2	BGJVW
3	DT
4	L,LL,Y
5	MNN
6	FPR

Tabla 20. Propuesta de código fonético para el idioma Español

De la misma forma que el algoritmo fonético, una palabra la representamos por este conjunto de dígitos, considerando las siguientes cuestiones, apoyado con un ejemplo, la palabra <hola>, se muestra el algoritmo de conversión a código fonético de una palabra:

1. Primero se cambia a mayúsculas la palabra <HOLA>
2. Se convierte a código fonético, quedaría <0040>, a diferencia de el algoritmo original, también se cambia la primera letra a su digito correspondiente.
3. Se quitan los ceros que me representan las vocales y la <H>, queda <4>
4. A diferencia del proceso original, donde se rellenaba con cero hasta completar una longitud de 4, en este caso si la longitud es menor a 4 se deja tal cual, en caso de ser mayor a 4, se toman los primeros 4 dígitos que me representarían esa palabra. En el caso particular, para palabras en donde sus caracteres son todos <0>, ejemplo: <ahí>, entran en una categoría nueva, más adelante se menciona como se distinguirá de las demás.

Siguiendo este código fonético, se crea un índice con 927 códigos para el vocabulario de palabras.

Retomando el ejemplo de la palabra ‘<hola>, se obtiene el código 4, que representa esta palabra, del cual contiene un conjunto de 78 palabras que también contienen el mismo código. Se observa que el conjunto es muy reducido al obtenido con índice de trigramas. El cual soluciona el problema de la palabra mal escrita <ohla>, ya que obtiene el mismo código que es 4. Y por consiguiente, en la fase posterior podrá asignarla correctamente a <hola>.

Una desventaja es que no se soluciona los erros tipográficos, por ejemplo ‘<holap> es una variación de la palabra <hola>, y tiene un código fonético de <46> diferente a <4>.

5.3. Análisis Previo

Para poder explicar el proceso de elección de candidatas para una palabra de un mensaje corto, primero se hacen varias observaciones, se mencionan diversas características que se consideran para la siguiente fase y se obtenga una mayor precisión de elección de la candidata ideal.

Para empezar se hizo una limpieza del mensaje:

1. Se eliminan acentos, la <ñ> se cambia a <n>.

2. Se localizan los formatos de hora, fecha, página, correo y los emoticones que se localizan en cada mensaje y se etiquetan:

   - **Formato emoticón.** TAGEMOTICON + [número de emoticón localizado en el diccionario], compara los emoticón disponibles en el diccionario, y al encontrar uno, este ro reemplaza por su etiqueta correspondiente. Resaltar que el diccionario de emoticones se encuentra ordenado de mayor a menor número de caracteres (debido a que hay emoticones contenidos en otros, así que se da mayor importancia a los de mayor longitud).

   - **Formato hora.** TAGHORA, el formato de hora es:
     [Numero][Numero]?[:][numero][numero]?

   - **Formato correo.** TAGCORREO, el formato de correo es:
     [a-zA-Z0-9-__]+@[A-Za-z]+[.][c][o][m]

   - **Formato Página.** TAGPAGINA, el formato de página, se tienen tres:

(1) http://[a-zA-Z0-9.:,/#$=?_@]+
(2) https://[a-zA-Z0-9.:,-/#$=?_@]+
(3) www.[a-zA-Z0-9.:,-/#$=?_/@]+

3. Buscar abreviaciones comunes y sustituirlas, apoyándose del diccionario de abreviaciones.4. Eliminar caracteres no alfanuméricos: puntos, comas, puntos suspensivos, punto y coma, dos puntos, signos de interrogación y exclamación, etc.

4. Palabras formadas con letras y los números 1 y 2 se pasó a su representación escrita, de igual manera números pequeños se convirtió a su correspondiente representación.
   Ejemplo: <esta2> a <estados>; <2> a <dos>.

5. Signo <+>, <-> y <=>, se sustituye por <mas>, <menos> y <mas>.

6. Palabras con letra <w> se le agrego la vocal <u>, <wu> esto para obtener mejor precisión en la siguiente fase.

7. Palabras que contienen <x> junto con alguna vocal <aeiou> se agrega una letra <h> entre las dos.

8. Palabras que contienen la letra <q> y no tiene la vocal <u> seguida, se añade quedando <qu>

9. Palabras que expresan algún estado de ánimo como <haaaa>, <heeeee> <mmmmm>, <jejejeje>, <jajajajaja>, <hoooo>, <uffffff>, se establece un formato estándar a 2,3 o 4 letras dependiendo de cuál se trate: <ha>, <he>, <mmm>, <jeje>, <jaja>, <ho> y <uf>.

10. Se buscan las abreviaciones más comunes, vistas en nuestro diccionario de abreviaciones, antes mencionado y se sustituyen por su correspondiente palabra escrita correctamente.

Todos estos cambios, se observaron de los mensajes, y ayudaron a obtener una mayor precisión o reducción de palabras candidatas a elegir.

5.4. Elección del conjunto de candidatas

En esta fase, se procede a elegir un conjunto de palabras candidatas, de entre las cuales puede estar la palabra ideal que remplace a la del mensaje corto.

Para esto, se procede a convertir las palabras de un mensaje corto a su correspondiente código fonético, apoyado del propuesto en la fase de creación de índices.

Para cada código obtenido, dependiendo de su longitud se aplica la elección de candidatas:

1. Para los de longitud 0 añadimos a candidatas las palabras que tienen código de longitud cero en el índice, corresponden aquellas palabras relacionadas con el índice etiquetado como <0>, además de todas las palabras con código de longitud 1.
2. Para los de longitud 1, añade las palabras con códigos de longitud 0 más los de longitud 2 que contengan al de longitud 1, también los de longitud 3 que coincidan su primer digito.
3. Para los de longitud 2, añadimos los de longitud 0, los de longitud 1 que contengan alguno de los dos dígitos; se añaden los de longitud 3 que contienen a los dos dígitos consecutivos; también a los de longitud cuatro que coincidan los dos dígitos con los dos primeros consecutivos.
4. Para los de longitud 3, se sacan dos dígitos consecutivos, posición (1,2) y (2,3), se añaden las palabras con esos dos códigos; añadimos los de longitud 4 que contengan a ese código de longitud 3.
5. Para códigos de longitud 4, se sacan combinaciones de dígitos de longitud 2: (1,2), (2,3), (3,4) y se buscan coincidencias de estos, en los códigos de longitud 3 añadiendo las palabras relacionadas a esos códigos al conjunto candidato.

   También, se consideran aquellas de longitud 4 que coincidan en combinaciones a nivel de bigramas. Se ilustra con un ejemplo en particular: palabra <URGETNE>, su código fonético es <6532>. Sus combinaciones consecutivas se muestran en la tabla 21. Se generan 5 combinaciones, se toman las palabras del índice que coincidan, y se consideran parte del conjunto candidato.

   Observar que la candidata a la que se quiere llegar es la palabra <URGENTE> con código fonético <6523>, coincidiendo con la combinación 2.

PALABRA : URGETNE	CODIGO : 6532
Combinación 1	6532
Combinación 2	6523
Combinación 3	5623
Combinación 4	5632
Combinación 5	6352

Tabla 21. Combinaciones a nivel de Bigramas del código <6532>

En el ejemplo, anterior, se da el caso cuando los cuatro dígitos son diferentes, se obtienen 5 posibles combinaciones; cuando hay un digito repetido, se obtienen 4 combinaciones y cuando hay 2 dígitos repetidos quedan tres combinaciones.

Al hacer las combinaciones en cada caso mencionado anteriormente y considerando cada aspecto desde códigos de longitud cero a cuatro, se logra en la mayoría de veces resolver el problema de errores tipográficos a sabiendas que mi conjunto candidato contiene la que sería mi candidata ideal. Un inconveniente que se puede claramente observar es que mi conjunto de candidatas aumenta.

Posteriormente, se aplica un filtro, eliminando palabras que estén repetidas. Obteniendo un conjunto reducido y listo para la siguiente fase.

5.5. Elección de la candidata ideal

En este proceso ya contamos con un conjunto de candidatas reducido, a continuación, se necesita elegir de entre ellas, cuál sería la mejor transcripción de la palabra *OOV.* Se aplica distancia de Levenshtein, cadena más larga sobre esqueleto consonántico y por ultimo un modelo de lenguaje basado en bigramas.

5.5.1. Variante del algoritmo de Levenshtein

La distancia de Levenshtein, conocida como distancia de edición, fue creada e implementada por Vladimir Levenshtein a mediados del siglo XX, con el propósito de medir la diferencia entre dos secuencias de símbolos [23]. En [21], dan una demostración formal, sobre las operaciones que realiza dicha distancia.

La distancia de Levenshtein es el número mínimo de operaciones requeridas para transformar una cadena de caracteres en otra, las operaciones que maneja son: sustitución, eliminación y agregación. El costo por operación por lo regular es de 1; va recorriendo carácter por carácter, y tomando el mínimo de operaciones con forme va avanzando. El algoritmo original se observa a continuación:

Recibe como entrada, dos palabras: str1 y str2

```
Distancia_Levenshtein (str1 [1... lenStr1], str2 [1…lenStr2])
        // se crea una matriz de tamaño  lenStr1+1 filas y lenStr2+1 columnas
        Declara d[0..lenStr1, 0..lenStr2]
        // i y  j son usadas para iterar sobre str1 y str2
        Para  I de 0 a lenStr1
          d[i, 0] := i
        para j de 0 a lenStr2
          d[0, j] := j

        para i de 1 a lenStr1
          para j de 1 a lenStr2
            si str1[i] = str2[j] entonces costo := 0
                        si no costo := 1
            d[i, j] := mínimo(
                        d[i-1, j] + 1,    // eliminación
                        d[i, j-1] + 1,    // inserción
                        d[i-1, j-1] + cost   // sustitución
                      )
       //se regresa la distancia que hay entre las dos palabras
        Regresa d[lenStr1, lenStr2]
```

Se propone una variación del algoritmo de levenshtein, modificando el costo que se le da a cada operación, se ha observado, que el costo es proporcionado por un número fijo, definido antes de correr el algoritmo, suele ser 1, la propuesta es variar este costo, apoyándose de la fonética, para esto se hace uso del código fonético soundex propuesto para el idioma español.

A continuación se describe los diferentes casos posibles para asignar el costo:

1. Cuando se aplica alguna de las operaciones de una vocal a una consonante, se da un peso de 0.5
2. Cuando es de una consonante a una vocal, se da un peso de 1.0
3. Si pertenecen ambas letras al mismo código fonético se da un peso de 1.0
4. En caso contrario, sino tienen el mismo código fonético se da un peso de 1.5

Como en el algoritmo original, aplica cada una de las operaciones, para letras iguales el peso es de 0, en caso contrario se revisa en que caso de los antes mencionados pertenece y de entre las tres operaciones se toma el mínimo que lleva hasta el momento cada operación.

Se obtiene la distancia de Levenshtein de cada candidata con respecto a la palabra del mensaje.

5.5.2. Cadena más larga sobre esqueleto consonántico

También se añade otro factor discriminativo, considerar la cadena más larga consecutiva de la palabra, pero solo considerando el cuerpo consonántico de la palabra.

Por ejemplo la palabra <LAS> que se encuentra en el conjunto candidato de la palabra escrita en el mensaje <LS>, tiene una longitud de cadena más larga de 2.

Al obtener tanto la distancia de Levenshtein del ejemplo que es <1>, unimos el factor de la cadena más larga <2>, haciendo una diferencia.

$$\text{Factor discriminante} = DL - CML$$

DL, Distancia de levenshtein

CML, cadena más larga

Del ejemplo, por consiguiente tenemos un factor discriminante de -1. Se observa, que aun apoyándonos del factor de cadena más larga tenemos que la distancia mínima a veces corresponde a más de una palabra, ejemplo en la tabla 22.

PALABRA DEL MENSAJE: <LS>

CANDIDATA	DISTANCIA
LOS	-1
LAS	-1
LES	-1

Tabla 22. Ejemplo aplicando Levenshtein y cadena más larga

5.5.3. Modelo del Lenguaje Basado en Bigramas.

Así que hasta este punto, se sigue teniendo en algunos casos, un conjunto candidato, muy reducido, pero con ambigüedad; entonces, en vez de ya tomar de entre ellas la candidata ideal, se procede a concentrarse en un conjunto reducido, obtenido con base a el mínimo valor de factor discriminativo por cada palabra, y sobre un rango de distancia de 0.5, y sobre él, se propone aplicar un modelo de lenguaje basado en bigramas.

Un modelo de lenguaje, es un mecanismo para definir la estructura del lenguaje, es decir, para restringir adecuadamente las secuencias de unidades lingüísticas más probables.

La probabilidad de que la palabra w_i venga después de la palabra w_{i-1} en una oración viene dada por la cantidad de veces que aparece la dupla: $w_i\, w_{i-1}$ en un corpus, dividida por el total de veces que aparece la palabra w_{i-1} .

$$P(w_i \mid w_{i-1}) = \frac{frecuencia(w_i\, , w_{i-1})}{frecuencia(w_{i-1})}$$

Para obtener las frecuencias, se hace uso de un recurso nuevo, que no se mencionó antes. Se creó un diccionario de bigramas, apoyándonos del conjunto de mensajes escritos correctamente; se van obteniendo bigramas consecutivos y su probabilidad.

Al conjunto reducido de candidatas se le aplica el modelo del lenguaje, se van tomando dos conjuntos de candidatas, por cada par de palabras consecutivas, y por cada candidata de la palabra 1, se hace combinaciones con cada una de las candidatas de la palabra 2, así pues, se busca en el diccionario de bigramas y se obtiene su probabilidad.

Por último, se elige la palabra que tenga la mayor probabilidad, y se convierte en la candidata ideal para esa palabra del mensaje, en caso de no haber coincidencia con algún bigrama, se procede a elegir la de menor factor discriminativo, en caso de haber palabras con el mismo valor, se elige la última.

5.6. Conclusiones del capítulo

En este capítulo, se describió la propuesta de un modelo, el cual es un compuesto de técnicas léxicas y sintácticas, las cuales alimentaron para obtener una mayor precisión, y resolver el problema de normalización de vocabularios en textos cortos.

Primero se mencionan los recursos léxicos utilizados para este trabajo, se fueron recolectando a lo largo del tiempo que se le dedico a este trabajo.

Posteriormente, se presenta dos formas de mostrar el campo de búsqueda de candidatas, no considerar todo el vocabulario disponible, sino solo un conjunto que concuerda en características fonéticas entre palabras con el mismo código fonético, reduciendo el costo en operaciones y tiempo. El índice creado previo a la aplicación de las técnicas, ayuda a reducir el tiempo de procesamiento.

Como segunda fase, se ve como la fase de supervisión, la cual fue de análisis de los mensajes de textos cortos, su comportamiento lingüístico, apoyándonos del capítulo 4 donde se mencionan diferentes de estos aspectos que se consideran. También al aplicar el diccionario de abreviaciones, apoya aquellos casos en donde la propuesta no pudo resolver, palabras que representaban a más de una, o que se mostraban como máximo un 50% de caracteres al original.

La siguiente fase obtiene por cada palabra *IV,* su conjunto candidato, añadiendo el término de combinaciones, variantes desde códigos fonéticos de log uno a cuatro, para cada caso diferentes formas de obtenerlo. Así con esto se logra atacar el problema de errores tipográficos, que son muy comunes en este tipo de mensajes.

En la siguiente fase, se procede a la elección de la candidata, en esta fase se logra reducir a un más el espacio del conjunto de candidatas, apoyándose de una variación del algoritmo de levenshtein, considerando a la fonética dentro. Además de añadir el factor de longitud de cadena más larga con la variante de aplicarlo solo al cuerpo consonántico. Aun asi hay casos en donde todavía no se puede

elegir la mejor transcripción de la palabra, debido a la existencia de mismos valores discriminativos.

Como aun había más de una elección de la candidata ideal, se procede al uso de modelo del lenguaje basado en bigramas, calculando la probabilidad de una palabra dada su anterior. Se hace uso de un diccionario de bigramas, el cual fue creado previo a la aplicación de esta técnica.

Así al final tomar la de mayor probabilidad, y obteniendo la candidata ideal para esa palabra.

Todo este proceso, es secuencial, cada una de las técnicas empleadas, fueron elegidas, debido a que son las que mejor comportamiento y resultados se obtuvieron, se observa claramente, que el seguir alimentando mis recursos léxicos y sintácticos, ayudara a ir obteniendo mayor precisión de transcripción de un mensaje escrito en lenguaje SMS, a su correspondiente forma estándar.

CAPÍTULO 6

EVALUACION DE LA PROPUESTA Y RESULTADOS.

En este capítulo se procede a describir tanto el corpus empleado y los diferentes experimentos realizados, y dar a conocer el porcentaje de precisión del modelo propuesto con respecto al baseline y a cada fase del proceso.

6.1. Corpus Paralelo

Un corpus paralelo, contempla dos partes, la primera son los mensajes escritos correctamente, y la otra son mensajes escritos en lenguaje SMS.

Para poder trabajar y evaluar las técnicas utilizadas, se hizo uso de dos corpus paralelos, para llevar realizar la tarea de normalización.

6.1.1. Corpus Patera

Este corpus, fue proporcionado por el departamento de Recuperación de Información de la BUAP.

Es una colección de textos cortos, de un libro llamado Patera, el cual tiene la parte correctamente escrita y la parte la cual contiene palabras *IV*. Cabe mencionar que es un corpus en español de España, pero trabajo bien sobre las técnicas propuestas.

En la Tabla 23 se resumen algunas de sus características.

Tamaño del corpus	295 mensajes cortos
Promedio de tamaño del mensaje	102.4 palabras
Porcentaje de palabras mal escritas	54.94%
Porcentaje de palabras bien escritas	45.06%
Promedio de emoticones	0.0
Tamaño de mensaje más pequeño	1 palabra
Tamaño del mensaje más grande	286 palabras

Tabla 23. Características del corpus paralelo Patera

6.1.2. Corpus SMS

Este corpus fue generado por la aplicación propuesta en el capítulo 3. La parte correctamente escrita fue realizada manualmente.

En la Tabla 24 se resumen algunas de sus características:

Tamaño del corpus	2549 mensajes cortos
Promedio de tamaño del mensaje	13.548 palabras
Porcentaje de palabras mal escritas	48.87%
Porcentaje de bien escritas	51.13%
Promedio de emoticones	0.0123
Tamaño de mensaje más pequeño	1
Tamaño de mensaje más grande	79

Tabla 24. Características del corpus paralelo SMS

6.2 Análisis de los índices generados

Como se menciona en el capítulo 5, en la creación de los índices, se hacen dos propuestas, de las cuales cada una se aplicó, y se obtuvo diferentes resultados.

6.2.1 Índice basado en trigramas

Para esta propuesta basada en índice de trigramas, se tienen 5,204 índices.

6.2.2. Índice basado en fonética

Para esta propuesta, usando soundex con variación del código fonético para el español, se tienen 16,482 índices. Mencionar que se aplicó soundex sin restricción, es decir, no se acotaba a longitud 4 los códigos, sino se quedaba con todos sus dígitos.

6.2.3. Índice basado en fonética con truncamiento a longitud 4

Para esta propuesta, usando soundex con variación del código fonético para el español, se tienen 927 índices. Se considera tomar hasta 4 dígitos del código fonético para aquellos que lo sobrepasen.

6.3. Comparación del modelo propuesto

Para poder evaluar los diferentes experimentos con el modelo propuesto, se propuso el uso del coeficiente de similitud de Jaccard, el cual es una medida de similitud entre conjuntos de muestras finitas; definido como la intersección de dos conjuntos dividida entre el tamaño de la unión de los conjuntos de muestra, formalmente descrito en la figura 7.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

Figura 7. Coeficiente de Jaccard

En nuestro caso, un conjunto A, me va a representar los mensajes escritos correctamente, mientras que un conjunto B, es la parte de mensajes que me arrojo al aplicar los diferentes experimentos, llegando al modelo propuesto.

Como baseline, nosotros aplicamos el coeficiente de Jaccard sobre el corpus paralelo de cada uno, para ver qué tan similares son, y que sirva como punto de partida, para mejorar ese coeficiente.

Resaltar, que en cuanto más se acerque el coeficiente obtenido por cada para de mensajes a 1, tienden a ser más similares. También que el promedio de todos los coeficientes obtenidos en cada par de mensajes, me representa la precisión de la técnica implementada sobre ese corpus en concreto.

Se aplicaron diferentes experimentos, para poder comparar el modelo propuesto, también algunos experimentos constan de algunos algoritmos originales aplicados a los corpus o pertenecen a propuestas de artículos, resaltando que algunos se aplican al idioma inglés.

Se aplican a los experimentos, con las observaciones que se hicieron en el análisis, visto en el capítulo 5, y otros sin ellas, para así recalcar, el proceso de análisis lingüístico ayudo en la mejora de la precisión.

6.3.1. Soundex original y propuesta de artículo

Se aplicó soundex con el código fonético original en idioma inglés, y con un código fonético para el español, propuesto en un artículo.

6.3.2. Soundex original y propuesta de artículo con levenshtein

Se aplica 6.3.1, añadiendo también la distancia de Levenshtein del algoritmo original con costo igual a 1 para las tres operaciones.

6.3.3. Trigramas con levenshtein original

Se calcula índice de trigramas para reducir el conjunto candidato y levenshtein para escoger la candidata ideal.

6.3.4. Primera versión. Propuesta de soundex con propuesta de levenshtein

Es la primera versión, solo aplicamos soundex con el código fonético propuesto y también con la propuesta de la variante de Levenshtein. Se hace esta primera versión para ir observando, como ayuda ir integrando una técnica cada vez más.

6.3.5. Segunda versión. Añadiendo longitud de cadena más larga y combinaciones consecutivas

Se aplica 6.3.4, añadiendo el factor de longitud de cadena más larga, usando solo su cuerpo consonántico. También se añadieron a aquellas candidatas que coincidan en su código fonético con las combinaciones.

6.3.6. Tercera versión. Añadiendo modelo del lenguaje

Aquí es donde añadimos como técnica final, el modelo del lenguaje basado en bigramas mencionado en el capítulo 5.

6.4 Resultados

Teniendo en cuenta la forma de evaluar basada en el coeficiente de Jaccard y los diferentes experimentos implementados más las versiones de la propuesta de modelo. En la Tabla 25 se muestran características referentes a aplicarlos al corpus Patera.

Experimentos\evaluaciones	Promedio Jaccard	Tiempo ejecución
BASELINE	0.1812	1.67s
SOUNDEX INGLES	0.056	2.72s
SOUNDEX ARTICULO	0.057	2.71s
SOUNDEX INGLES/LEVENSHTEIN ORIG	0.510	5m24s
SOUNDEX ART/LEVENSHTEIN ORIG	0.518	5m25s
TRIGRAMAS/LEVENSHTEIN ORIGINAL	0.468	6m09s
V1: SOUNDEX PRO/LEVENSHTEIN PRO	0.533	25m56s
VERSION 2	0.610	201m32s
VERSION 3	0.801	225m52s

Tabla 25. Evaluación sobre el corpus Patera

En la Tabla 26 se muestran características referentes al aplicarlos sobre el corpus SMS.

Experimentos\evaluaciones	Promedio Jaccard	Tiempo ejecución
BASELINE	0.225	2.072s
SOUNDEX INGLES	0.066	4.084s
SOUNDEX ARTICULO	0.067	4.067s
SOUNDEX INGLES/LEVENSHTEIN ORIG	0.525	8m1s
SOUNDEX ARTICULO/LEVENSHTEIN ORIG	0.531	8m.4s
TRIGRAMAS/LEVENSHTEIN	0.448	9m32s
V1: SOUNDEX PRO/LEVENSHTEIN PRO	0.553	30m4s
VERSION 2	0.688	175m43s
VERSION 3	0.813	203m83s

Tabla 26. Evaluación sobre el corpus SMS

6.5 Conclusiones del capítulo

Los corpus paralelos, sobre los que se trabajó, contienen características similares a los comunes textos cortos, sirviendo así pues, para evaluar las diferentes técnicas empleadas.

El baseline, consistió en aplicar Jaccard directamente al mensaje limpio con el mensaje sucio, por lo que se observó que tan diferentes son el uno con el otro, ver lo parecidos o no que son ambos, y así poder partir con cada uno de los diferentes experimentos, hasta llegar a la propuesta final.

Se ve claro, que el modelo propuesto fue direccionado hacia la fonética, tomando en cuenta aspectos lingüísticos como apoyo, así como la combinación de técnicas tanto léxicas como sintácticas.

Se observa, que los experimentos implementados del estado del arte, no superan el 50% de buena transcripción. A partir de la primera versión supera a los experimentos implementados del estado del arte sobre ambos corpus y se empieza a ver mejoría por ese camino.

A medida que aplicamos cada una de las versiones se consigue una mejor precisión al transcribir el mensaje corto, en su correspondiente correcta escritura del idioma español.

El tiempo de ejecución de la versión 4 supera considerablemente a las anteriores, se debe a que al aplicar las combinaciones, se obtiene un mayor número de conjuntos candidatas a las cuales aplicarles las diferentes técnicas.

Conclusiones finales y trabajo a futuro

A continuación se presentan las conclusiones finales de este trabajo de tesis. De igual manera el trabajo a futuro para una posible extensión de esta tesis.

Conclusiones Finales

El principal objetivo de esta tesis fue el desarrollo de un modelo para la normalización de vocabularios en textos cortos, en este caso se usaron dos corpus con este tipo de características, mensajes los cuales contaran con palabras *OOV,* de entre las cuales hay palabras recortadas o truncadas, omisiones o sustituciones de letras ya sea porque se escriben tal cual suena, o por un error tipográfico.

Se mencionaron dos tipos de errores, espontáneos y voluntarios, el primero son los comunes errores tipográficos, debido al dispositivo que es muy pequeño, o la rapidez con la se escribe un mensaje; los voluntarios vienen de la mano, con escribir palabras que ya tenemos definidas o por escribirlas tal cual suenan y se hace más fácil.

En este tipo de mensajes, se observa la aparición de emoticones los cuales me representan una emoción, también la existencia de onomatopeyas, las cuales de la misma manera reflejan un estado de ánimo. El poder diferenciar alguno de estos tipos, fue gracias a la observación de dichos mensajes y a la creación de diccionario de emoticones, y así diferenciarlos y etiquetarlos,

El modelo propuesto fue una serie de técnicas en cascada que se acoplaron bien, obteniendo buenos resultados, el uso constante de la fonética, y el camino que se siguió, hizo posible combinar una técnica fonética con la distancia de edición.

Se propuso un código fonético, el cual supero al original y la propuesta de un artículo, ayudo tanto a crear el índice fonético como el costo para las diferentes operaciones de la distancia de Levenshtein.

Se propuso un factor de descremación, obtenido de la distancia de edición y la longitud de la cadena más larga, esta última basada solo sobre el cuerpo consonántico de dicha palabra. Aun así se presentaron casos en donde existían palabras con el mismo valor discriminativo, por lo que se hizo uso de una última técnica la cual permitiera elegir la candidata ideal basada en su probabilidad; el modelo de lenguaje basado en bigramas, ayudo a una mejor precisión.

Trabajo a Futuro

Este trabajo lo que tiene, es que en cuanto se siga enriqueciendo los diferentes recursos léxicos y sintácticos, mejores resultados se obtendrían y se volverá más general para otros corpus. El crecimiento tanto del vocabulario, el diccionario de emoticones, el de abreviaciones y hasta el de probabilidades de bigramas, ayudara en próximas tareas de procesamiento de lenguaje natural. Entonces, seguir con el crecimiento de los diferentes recursos.

El poder aplicar técnicas semánticas, y observar su comportamiento, para este tipo de mensajes, ver las ventajas y desventajas que traería.

Se aplicó el modelo de lenguaje sobre bigramas, ahora poder aplicarlo a trigramas sería un buen experimento.

Probar estas mismas técnicas sobre otros corpus, de otras fuentes como Twitter o Facebook.

Considerar los diferentes signos de puntuación, y un mayor etiquetado para una posterior aplicación de tarea de procesamiento de lenguaje natural.

Bibliografía

[1]"Twitter". Disponible en internet: http://geeksroom.com/2013/09/twitter-en-20-estadisticas-actualizadas-a-agosto-2013/

[2]"*México Proyecciones de Población al 1ro Enero 2013".*Disponible en Internet:*de*http://www.coespomor.gob.mx/investigacion_poblacion/proyeccion_20013/proyeciones%20a%202013_datos%20numericos.pdf

[3]"*CNN México*". Disponible en internet: http://mexico.cnn.com/tecnologia/2013/08/03/mexico-conexion-internet-moviles-redes-sociales-cpmx4-campus-party.

[4]*"Cofetel".* Disponible en internet: http://www.cft.gob.mx

[5] "Facebook". Disponible en: http://www.trecebits.com/2013/08/26/13-sorprendentes-estadisticas-de-facebook-infografia/

[6] *"Soundex"* disponible en internet: http://es.wikipedia.org/wiki/Soundex

[7]"*Publicaciones del Laboratorio de Recuperación de Información".* Disponible en internet: www.cs.buap.mx~dpinto/pubs.html.

[8] *Jesús Oliva, José Ignacio Serrano, María Dolores del Castillo, and Ángel Iglesias*. SMS Normalization: Combining Phonetics, Morphology and Semantics. CAEPIA 2011, LNAI 7023, pp. 273–282, 2011. Springer-Verlag Berlín Heidelberg 2011.

[9] *AiTi Aw, Min Zhang, Juan Xiao, and Jian Su.* A Phrase-based Statistical Model for SMS Text Normalization. In Proceedings of the COLING/ACL on Main conference poster sessions, COLING-ACL '06, pages 33–40, Stroudsburg, PA, USA, 2006. Association for Computational Linguistics.

[10] *ChenLi Yang Liu.* Improving Text Normalization Using Character-blocks based Models and System Combination. *Proceedings of COLING 2012: Technical Papers*, pages 1587–1602, COLING 2012, Mumbai, December 2012.

[11] *PidongWang, Hwee Tou Ng.* A Beam-Search Decoder for Normalization of Social Media Text with Application to Machine Translation. Proceedings of NAACL-HLT 2013, pages 471–481, Atlanta, Georgia, 9–14 June 2013.

[12] *Serguei Pakhomov*. Semi-Supervised Maximum Entropy Based Approach to Acronym and Abbreviation Normalization in Medical Texts. Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL), Philadelphia, July 2002, pp. 160-167.

[13] *Govind Kothari, Sumit Negi, Tanveer A. Faruquie, Venkatesan T. Chakaravarthy, and L. Venkata Subramaniam.* SMS Based Interface for FAQ Retrieval. In Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2, ACL-IJCNLP '09, pages 852–860, Morristown, NJ, USA, 2009. Association for Computational Linguistics.

[14] *Danish Contractor, Govind Kothari, Tanveer A. Faruquie, L. Venkata Subramaniam, and Sumit Negi.* Handling Noisy Queries in Cross Language FAQ Retrieval. In Proceedings of the 2010 Conference on EmpiricalMethods in Natural Language Processing, EMNLP '10, pages 87–96, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[15] Han, B., & Baldwin, T. (2011, June). Lexical Normalisation of Short Text Messages: Makn Sens a# twitter. In *ACL* (pp. 368-378).

[16] LIU, Fei; WENG, Fuliang; JIANG, Xiao. A broad-coverage normalization system for social media language. En Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1. Association for Computational Linguistics, 2012. p. 1035-1044.

[17] "Emoticones". Disponible en internet: http://emoticonos.smilchat.net/.

[18] Christopher D. Manning, Hinrich. Foundations of Statistical Natural Language Processing. MIT Press, 1999.

[19] Else, Willis I. 2002. The Complete Soundex Guide: Discovering the rules Used by the Census Bureau and the Inmigration and Naturalization Service When These organization Indexed Federal Records. Apollo, PA: Closon Press.

[20] Iván Amón, Francisco Moreno, Jaime Echeverri. 2012. Algoritmo fonético para la detección de cadenas de texto duplicadas en el idioma español. Revista Ingenierías Universidad de Medellín. Pp. 127-138. Medellín, Colombia.

[21] González, Abdiel E. Cáceres. La métrica de Levenshtein. Revista de Ciencias Básicas UJAT, 2008, vol. 7, no 2, p. 35-43.

[22] Díaz, M.; Pérez, J.; SANTANA, O. Distancia Dependiente de la Subsecuencia Dependiente más Larga entre Cadenas de Caracteres. Anales de las II Jornadas de Ingeniería de Sistemas Informáticos y de ComputaciÃ³n, Quito (Ecuador), 1993, vol. 117, p. 123.

[23] V. I., Levenshtein. Binary codes capable of correcting deletions, insertions, and reversals. Soviet Physics Doklady 10 (1966):707710. Traducción al inglés de la versión original en ruso publicada en 1965.