FACULTAD DE INFORMATICA DE BARCELONA

ESPECIALIDAD DE CIENCIAS DE LA COMPUTACION

SISTEMA DE TRADUCCIÓN NEURONAL USANDO BITMAPS

Autor: David Aldón Mínguez

Directora: Marta Ruiz Costa-Jussà

Ponente: Lluís Padró Cirera

Agradecimientos

En primer lugar quiero dar las gracias Marta Ruiz Costa-Jussà, la cual se ha encargado de llevar a cabo el seguimiento del proyecto en todo momento. Además, quiero dar las gracias José Adrián Rodríguez Fonollosa el cual tuvo como idea la realización de este proyecto. A todo esto lo que quiero decir es que sin la ayuda de ambos no hubiese sido posible la realización de este proyecto.

También quiero agradecer su disposición a ofrecer los corpus y el servidor ssh para la realización del proyecto, ya que sin dicho material no hubiese podido realizar este estudio.

Finalmente estoy muy agradecido a mi familia, amigos y compañeros de la carrera por haberme apoyado en todo momento y ayudarme siempre que les ha estado posible.

Resumen

Recientemente, los sistemas de traducción basados en redes neuronales están empezando a competir con los sistemas basados en frases. Los sistemas que se basan en las redes neuronales usan representaciones vectoriales de las palabras. Sin embargo, uno de los mayores retos que aún enfrenta la Traducción Automática (TA), se trata de grandes vocabularios y lenguas morfológicamente ricas. Este trabajo tiene como objetivo adaptar un sistema de TA neuronal para traducir del chino al español, utilizando como entrada diferentes tipos de granularidad: palabras, caracteres, fuentes de mapa de bits de las palabras chinas y las fuentes de mapa de bits de caracteres chinos. El esquema sugerido para el modelo TA neuronal mitiga el problema de las palabras de origen desconocido. El hecho de realizar la interpretación de cada carácter o palabra como una fuente de mapa de bits permite obtener representaciones vectoriales más informadas. Los mejores resultados se obtienen cuando se utiliza la información de la fuente palabras del mapa de bits.

Resum

Recentment, els sistemes de traducció basats en xarxes neuronals estan començant a competir amb els sistemes basats en frases. Els sistemes que es basen a les xarxes neuronals usen representacions vectorials de les paraules. No obstant això, un dels majors reptes que encara enfronta Traducció Automática (TA), es tracta de grans vocabularis i llengües morfològicament rics. Aquest treball té com a objectiu adaptar un sistema de TA neuronal per traduir del xinès a l'espanyol, utilitzant com a entrada diferents tipus de granularidad: paraules, caràcters, fonts de mapa de bits de les paraules xineses i les fonts de un mapa de bits de caràcters xinesos. L'esquema suggerit per al model TA neuronal mitiga el problema de les paraules d'origen desconegut. El fet de realitzar la interpretació de cada caràcter o paraula com una font de mapa de bits permet obtenir representacions vectorials més informades. Els millors resultats s'obtenen quan s'utilitza la informació de la font paraula de mapa de bits.

Abstract

Recently, translation systems based on neural networks are starting to compete with systems based on phrases. The systems which are based on neural networks use vectorial representations of the words. However, one of the biggest challenges that Machine Translation (MT) still faces, is dealing with large vocabularies and morphologically rich languages. This work aims to adapt a neural MT system to translate from Chinese to Spanish using as input different types of granularity: words, characters, bitmap fonts of Chinese words and bitmap fonts of Chinese characters. The suggested scheme for the Neural Machine Translation (NMT) model mitigates the problem of the unknown source words. The fact of performing the interpretation of every character or word as a bitmap font allows for obtaining more informed vectorial representations. Best results are obtained when using the information of the word bitmap font.

Índice general

Agradecimientos **I**

Resumen **II**

Resum **III**

Abstract **IV**

Lista de figuras **VIII**

Lista de tablas **IX**

1. Introducción y Estado del arte **1**

1.1. Contextualización 1

1.1.1. Glosario 2

1.1.2. Actores implicados 3

1.2. Estado del arte 5

1.2.1. Motivaciones 6

1.2.2. Diferencias lingüísticas entre español y el chino 8

1.2.3. Traducción por reglas vs traducción estadística 9

1.2.4. Sistema de traducción neuronal 11

2. Alcance, metodología y planificación **15**
2.1. Formulación del problema 15
2.2. Alcance 16
2.3. Posibles obstáculos 17
2.4. Metodología y rigor 18
2.4.1. Método de trabajo 18
2.4.2. Herramientas de seguimiento 19
2.4.3. Métodos de validación 19
2.5. Descripción de las tareas 20
2.5.1. Duración del proyecto 20
2.5.2. Planificación y diagrama de Gantt 20
2.5.3. Estimación de horas 23
2.5.4. Recursos 24
2.5.5. Valoración de alternativas y plan de acción 24
2.5.6. Diagrama de Gantt 25

3. Gestión económica y Sostenibilidad **29**
3.1. Gestión económica 29
3.1.1. Costes directos: recursos humanos 29
3.1.2. Costes directos materiales 29
3.1.3. Presupuesto 31
3.1.4. Control de gestión 31
3.2. Sostenibilidad 32
3.2.1. Sostenibilidad económica 32
3.2.2. Sostenibilidad social 33
3.2.3. Sostenibilidad ambiental 34

4. Descripción teórica **35**
4.1. Descripción del sistema de traducción basado en redes neuronales 36

4.1.1. Corpus 36
4.1.2. Generación de diccionarios 38
4.1.3. Sistema de referencia 39
4.2. Segmentación e inclusón 40
4.2.1. Separación de palabras en caracteres 41
4.2.2. Tratamiento de datos del sistema neuronal 42
4.2.3. Integración de bitmaps 44

5. Experimentación **47**
5.1. Sistema sin atención añadida 49
5.1.1. Sistema de referencia 50
5.1.2. Sistema de referencia usando información de bitmaps . 50
5.1.3. Ejemplos comparativos sin atención 51
5.2. Sistema con atención añadida 52
5.2.1. Sistema de referencia con segmentación de caracteres o palabras 53
5.2.2. Integración de información de bitmaps con segmentación de caracteres o palabras 54
5.2.3. Ejemplos comparativos con atención 56
5.3. Sistema con información de caracteres 57
5.4. Sistema con atención añadida con un corpus más grande . . . 59

6. Conclusiones **62**
6.1. Conclusiones técnicas y personales 62
6.2. Trabajo futuro 63
6.3. Publicación 64

A. Publicación **68**

Índice de figuras

1.1. Mapa mundial donde se muestra hablantes de chino y español como lenguas oficiales y cooficiales. 7
1.2. Decodificador-Codificador. 11

2.1. Estimación de horas. 23
2.2. Recursos Necesarios. 24
2.3. Gantt parte1 26
2.4. Gantt parte2 27
2.5. Gantt parte3 28

3.1. Tabla sostenibilidad. 33

4.1. Descripción del sistema de traducción basado en redes neuronales. 37
4.2. Diccionario de Palabras. 38
4.3. Decodificador-Codificador. 40
4.4. Estrategias de segmentación e inclusión de bitmaps. 41
4.5. Fuente para trabajar con caracteres chinos. 43
4.6. Código para generar bitmaps. 43
4.7. Letra China. 44
4.8. Nuevo sistema de traducción automática neuronal. 45

Índice de tablas

3.1. Costes directos: recursos humanos. 30
3.2. Costes directos materiales. 31
3.3. Presupuesto. 32

4.1. Detalles del Corpus. Número de frases (S),palabras (W), vocabulario (V) 38

5.1. Detalles de los Resultados. Tipo de experimento (T), Porcentaje de traducción correcta (BLEU) 1 palabra bien traducida consecutiva (1PC), 2 palabras bien traducidas consecutivas (2PC), 3 palabras bien traducidas consecutivas (3PC), 4 palabras bien traducidas consecutivas (4PC) 49
5.2. Frases de ejemplo. Origen (Src), sistema de referencia (CH), inclusión de Bitmap (+Bitmap), Objetivo (Ref) 51
5.3. Frases de ejemplo. Origen (Src), Palabras del sistema de referencia (Words), Fuentes de Bitmap (+Bitmap), Objetivo (Ref) 57
5.4. Detalles del Corpus. Número de frases (S),palabras (W), vocabulario (V) 60
5.5. Frases de ejemplo. Origen (Src), sistema de referencia (Words), inclusión de Bitmap (+Bitmap), Objetivo (Ref) 61

Capítulo 1

Introducción y Estado del arte

1.1. Contextualización

Este proyecto se realiza como un Trabajo de Final de Grado de la Universidad Politécnica de Barcelona, Grado de Ingeniería Informática, especializado en la rama de Computación, en la Facultad de Informática de Barcelona.

La idea de este proyecto es sacar el máximo provecho de los sistemas de traducción, basados en redes neuronales ya construidas para el uso de palabras chinas. En este estudio se pretende experimentar sobre qué entradas al sistema podrían ser más interesantes. Se trabajará tanto a nivel de palabra, carácter y bitmaps (caracteres y palabras), por bitmaps entendemos el proceso de convertir texto a imagen y de la imagen obtener la información, mediantes bits de 0 y 1.

En los sistemas de traducción neuronales la información que se recibe juega un papel muy importante, ya que cuanta más información se introduzca, mejor son sus resultados. Precisamente el chino es una lengua que permite tres granularidades de segmentación con mejores condiciones que otras lenguas, esto es debido a que cada carácter chino se representa simbólicamente,

por lo que nos es muy útil para utilizar la granularidad de bitmaps.

Debido a que actualmente estos sistemas están construidos para recibir como entrada palabras. En este proyecto se realizan modificaciones sobre el proceso que se lleva a cabo para la traducción neuronal, mediante distintas entradas al sistema neuronal. Donde se aplicaran técnicas de segmentación y tratamiento de datos.

1.1.1. Glosario

1. **Corpus:** Un corpus lingüístico es un conjunto de textos relativamente grande, creado independientemente de sus posibles fines de uso. Es decir, en cuanto a su estructura, variedad y complejidad, un corpus debe reflejar una lengua o su modalidad de la forma más exacta posible.

2. **Theano:** Theano es una librería de matemática para Python, se usa mucho en Deep Learning.

3. **Red neuronal recurrente (RNN):** Las neuronas pueden estar conectadas hacia adelante o hacia atrás. Puede que la conexión bidireccional esté limitada a neuronas de capas consecutivas (SRN redes recurrentes simples) o que no exista ninguna restricción hasta el punto de que todas las neuronas están conectadas con toda.

4. **Traducción automática (TA):** Es traducción automatizada. Es el proceso mediante el cual se utiliza *software* de computadora para traducir un texto de un lenguaje natural (como el inglés) a otro (como el español). También conocido en ingles como ***Machine Translation*** **(MT)**.

5. **Traducción automática estadística (SMT):** A las aproximaciones estadísticas de la traducción automática se le llama traducción

automática estadística.

6. **Traducción neuronal automática (NMT):** El objetivo de NMT es diseñar un modelo totalmente entrenable de los cuales cada componente se ajusta basándose en el corpus de entrenamiento para maximizar su rendimiento traducción.

7. **Sistema con atención añadida:** Dada una red neuronal, se añade un red bidireccional adicional al sistema, para mayor calidad del procesamiento de información.

8. **Bitmaps:** Vectores de 0 y 1, que se obtienen de la información abstraída de una imagen que contiene los caracteres/palabras chinas.

9. **Deep learning:** Es un conjunto de algoritmos en aprendizaje automático (en inglés, machine learning) que intenta modelar abstracciones de alto nivel en datos usando arquitecturas compuestas de transformaciones no-lineales múltiples

10. **BLEU:** Es un método de evaluación de la calidad de traducciones realizadas por sistemas de traducción automática. Una traducción tiene mayor calidad cuanto más similar es con respecto a otra de referencia, que se supone correcta. BLEU puede calcularse utilizando más de una traducción de referencia. Esto permite una mayor robustez a la medida frente a traducciones libres realizadas por humanos.

1.1.2. Actores implicados

A continuación, identificamos cuáles son las partes interesadas en nuestro proyecto. Es decir, aquellas personas, organizaciones y organismos que puedan tener algún tipo de interés en nuestra investigación. Cada una de las partes tendrá sus propios intereses para alcanzar sus objetivos.

Desarrollador emprendedor

Esta tarea se realiza por mí, ya que soy la única persona que realiza este proyecto. El objetivo principal es que este proyecto sirva como nuevo recurso hacia otras áreas, ya que es algo que actualmente está en continuo desarrollo y carece de muchos recursos debido a que está en fase experimental.

Tutora del proyecto

Este proyecto estará en todo momento dirigido por Marta Ruiz Costa-Jussà, la cual se encargará de supervisar toda la parte técnica de este estudio. Sus principales funciones son verificar que este estudio cumple con la planificación marcada en cada momento del proyecto y verificar que se cumplen los objetivos estipulados. También hay que remarcar que se encargará de guiar el desarrollo de este estudio, ya que es muy importante cada decisión para poder avanzar. Además estará como ponente Lluís Padró.

Lectores

La versión original del código que se utiliza en este proyecto es *open source*, ya que tiene por finalidad que cualquiera pueda aportar mejoras, para así poco a poco ir obteniendo mejores resultados. Por lo que podemos decir que cualquier persona interesada en el uso de redes neuronales sería un lector de este proyecto. El cual tendrá la finalidad de aportar nueva información corrigiendo errores cometidos anteriormente y aportando nueva información que se haya podido pasar por alto, con la intención de aportar nuevas ideas al conjunto de personas que trabajan en estos proyectos a cualquier nivel.

Centros de investigación en sistemas de traducción

Los centros de investigación pueden estar realmente interesados, tanto en resultados fallidos como óptimos, para poder ver posibles mejoras, o bien no invertir tiempo en algo fallido. Es decir, en caso fallido, este estudio sería de gran utilidad, para ver que es un camino que no tiene salida, por lo que no es necesario invertir tiempo ni dinero, tal y como muestra el proyecto. O bien en caso óptimo, con la finalidad de si realmente vale la pena volver a enfocar sus ideas actuales para ver si se pasó algo por alto y así mejorar en sus actuales estudios.

1.2. Estado del arte

Ahora se realizará un estudio del arte, el cual trata de dejar claro la frontera de conocimiento entre las investigaciones anteriormente realizadas con la investigación que se realizará en este proyecto. Es muy importante saber que errores se han cometido, qué cosas podrían mejorarse teniendo en cuenta dichos estudios y que opciones (diferentes métodos/técnicas) tenemos a la hora de decidirnos por un método o el otro. Además hay que destacar que antes de investigar sobre los métodos se debe tener en cuenta las características de la propia lengua para saber sus particularidades.

Comenzaremos preguntando: Qué es la traducción automática? Es el subcampo de la lingüística computacional que investiga el uso de programas para traducir texto o voz entre distintos idiomas [Figueroa et al.2011].

Una vez definido este concepto clave, podemos realizar un estudio del arte sobre la traducción automática (TA). En él, podremos ver las distintas técnicas que actualmente se utilizan para realizar dichas traducciones. Por lo que se intentará mostrar los últimos avances en este campo.

Este estudio del arte se dividirá en cuatro partes principalmente:

En la **primera** hablaremos de las motivaciones que nos han hecho llevar a cabo este trabajo final de grado, donde podremos ver la relevancia que tienen el par de lenguas chino-español actualmente en la sociedad, tanto a nivel político, económico o social.

En **segundo** lugar, hablaremos de las diferencias que existen lingüísticamente, entre dichas dos lenguas, ya que es un factor muy importante a tener en cuenta antes de comenzar a realizar cualquier estudio, ya que en función de ello podremos decidir qué método puede resultar más óptimo dadas unas condiciones iniciales.

Seguidamente trataremos sobre estudios ya realizados hasta el momento, donde se ha decidido mostrar una pequeña idea del funcionamiento de las traducciones estadísticas vs traducciones por reglas, ya que primeramente las investigaciones se realizaban mediante este tipo, y fue más tarde cuando se empezó a llevar a cabo la traducción neuronal. Esto nos puede ayudar a entender mejor la traducción neuronal, respecto a las ventajas /desventajas de un tipo de traducción u otra.

Finalmente, veremos que es lo que se está utilizando hoy en día, es decir, las redes neuronales, las cuales utilizan métodos explicados en el tercer paso, por lo que en este punto nos habrá sido útil haber realizado el estudio de traducciones por reglas y estadísticos. Además, en este punto se encuentra la frontera de conocimiento con nuestro proyecto la cual definiremos en este apartado.

1.2.1. Motivaciones

Hoy en día, disponemos de traductores automáticos que son capaces de traducir entre una gran cantidad de pares de lenguas. Evidentemente, los resultados son mejores o peores dada la complejidad de esas lenguas, es decir, la cantidad de recursos lingüísticos de cada lengua, lo cual indirectamente es

influenciada por el número de habitantes que dominan dicha asociación de lenguas.

Si nos centramos en nuestra situación, la asociación chino-español, e investigamos sobre su número de habitantes lingüísticos en el mundo, podemos observar que son 2 de las lenguas más habladas por todo el mundo, concretamente, el chino tiene la primera posición mientras el español la tercera [Wikipedia2016].

Seguidamente mostraremos los lugares donde se habla chino, y los lugares donde se utiliza el español correspondientemente (como lengua nativa y lengua cooficial), tal y como vemos el la figura 1.1.

Figura 1.1: *Mapa mundial donde se muestra hablantes de chino y español como lenguas oficiales y cooficiales.*

Ahora bien, la traducción que se realiza actualmente en el par de lenguas chino español, es peor que una traducción de un par de lenguas como podría ser catalán y sueco.

Con todo esto queremos remarcar, que investigar en traducción automática en este par de lenguas puede ser de gran interés a muchos niveles, donde podemos ver:

China es la segunda potencia económica del mundo, el primer exportador y posee las reservas de cambio más elevadas [San2016]. Por lo que una buena traducción sería de interés para transacciones, especialmente para lugares entre Pekín y México, Brasil, Perú... [Riaño et al.2015]

Como problemática, hay que destacar que el principal inconveniente de cualquier tipo de traducción para esta asociación de lenguas es la inexistencia, al menos como recurso públicamente disponible [Banchs et al.2006], de un corpus paralelo bilingüe lo suficientemente grande que sea puro, es decir, sin traducciones intermedias, para realizar un entrenamiento con la asociación chino-español.

1.2.2. Diferencias lingüísticas entre español y el chino

Para poder entender un poco lo que supone realizar un entrenamiento de la lengua para la traducción por reglas o bien estadística, haremos un pequeño comentario sobre en qué se diferencia el español del chino, lingüísticamente.

A nivel morfológico, el chino, es una lengua que no presenta inflexiones, es decir, "ordenador y ordenadores". Otro problema habitual, el cual se produce constantemente entre castellano y otra lengua, es que las frase de la otra lengua (chino), no suele aportar la información suficiente para saber a qué inflexión corresponde, por lo que debemos adaptarla a la situación más adecuada [Ángel José Riesgo2016].

A nivel sintáctico, ambas utilizan la metodología Sujeto-Verbo-Objeto, pero como toda lengua,el número de excepciones son prácticamente ilimitadas, lo que desborda a la traducción [Ángel José Riesgo2016].

A nivel semántico, el chino destaca por la capacidad que tiene a la hora de dar un significado u otro según el sonido que se produzca en la pronunciación, por lo que es algo complejo analizar a la hora de realizar una traducción del chino hacia el español [Ángel José Riesgo2016].

1.2.3. Traducción por reglas vs traducción estadística

Ahora que hemos visto un poco los retos lingüísticos que podemos encontrarnos a la hora de realizar una traducción, veremos lo que supone un método u otro de traducción:

Traducción por reglas

La traducción por reglas (automática) se realiza en tres fases: análisis, transferencia y generación [Abaitua2006].

1. En primer lugar, se toma la lengua de origen y se eliminan las posibles inflexiones de manera que se generalice la lengua.

2. En segundo lugar, se toma un diccionario de la asociación de ambas lenguas, y se realiza la transferencia a la lengua destino mediante reglas.

3. Finalmente se pasa de la forma general (lengua destino) a la forma particular, es decir, las correspondientes inflexiones de la lengua.

Traducción estadística

La traducción estadística se realiza mediante muchas metodologías, pero hoy en día, la más usada es la basada en segmentos [Costa-jussà2015].

Esta metodología se basa en tomar un par de textos paralelos a nivel de oración. Estos textos se alinean usando información de coocurrencias a nivel de palabras, es decir, por ejemplo, las locuciones son coocurrencias estables, ya que funciona como una unidad léxica con significado propio, no derivado de la suma de significados de sus componentes, por ejemplo "llueve a cántaros". Es entonces cuando extraemos un modelo de traducción, el cual cada palabra está asociada a una probabilidad de traducción.

Por otro lado se entrena la lengua destino con textos de su propia lengua. Una vez obtenidos estos dos modelos de entrenamiento, se realiza la búsqueda de la traducción destino. Esta búsqueda es realizada por el decodificador, que es un algoritmo de Viterbi [Costa-jussà2015].

Como herramientas de código libre para construir sistemas estadísticos podemos encontrar Giza++, Moses y SRILM [Costa-jussà2015].

Todo esto es conocido como el sistema *phrase-base*, el cual es utilizado actualmente en el traductor de google, donde concretamente para la traducción de chino a español su porcentaje de traducción se encuentra situado en 30 %, aunque actualmente se tiene abandonada la investigación en las técnicas de traducción estadística, ya que es muy difícil obtener mejoras.

Ventajas y desventajas

Algunas de las ventajas y desventajas de dichos sistemas explicados anteriormente podrían ser:

La traducción estadística sólo necesita aprender de un corpus paralelo para generar un motor de traducción. En cambio, la traducción basada en reglas necesita una gran cantidad de conocimiento externo al corpus que solamente expertos lingüistas pueden generar [Pangea2010].

Un sistema de traducción estadística se desarrolla rápidamente. En cambio, un sistema de traducción basada en reglas requiere grandes costes de desarrollo y personalización hasta que alcanza el umbral deseado de calidad [Pangea2010].

La traducción basada en reglas requiere un alto conocimiento lingüístico y muchas horas de dedicación tal y como hemos mencionado en el primer punto. Por eso la aproximación basada en reglas es una inversión a largo plazo [Costa-jussà2015].

Como conclusión, podemos decir que ambos pueden aspirar a obtener

mejores resultados, pero a pesar que uno es peor que otro, quizá puede ser más útil que el otro en un determinado ámbito.

1.2.4. Sistema de traducción neuronal

Actualmente se empiezan a utilizar sistemas neuronales, los cuales tienen la misma filosofía que los estadísticos, es decir, utilizan probabilidades, pero la principal diferencia es que los estadísticos son simplemente métodos de conteo, mientras los neuronales lo que utilizan son un paradigma de aprendizaje y procesamiento automático inspirado en la forma en que funciona el sistema nervioso de los animales. Se trata de un sistema de interconexión de neuronas que colaboran entre sí para producir un estímulo de salida.

Codificador-Decodificador

Todo este proceso se realiza mediante el codificador y un decodificador utilizado en [Cho2015a], tal y como vemos en la figura 1.2.

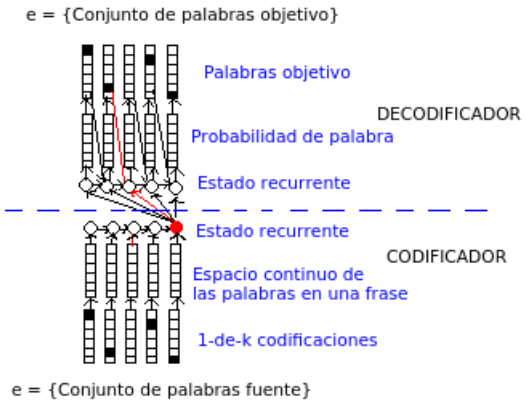


Figura 1.2: *Decodificador-Codificador.*

Este codificador-decodificador, a grandes rasgos, toma una frase origen

(conjunto de palabras sin tener en cuenta el orden), el cual codifica cada palabra a vectores de bits de 0/1 (sólo un bit a 1), seguidamente se multiplican por unos vectores probabilísticos y cada uno de estos vectores informa a un estado, donde sólo uno de ellos informará a todos los estados objetivos, finalmente estos estados aplican el proceso inverso explicado anteriormente obteniendo la frase destino.

El código de este proceso neuronal mediante el codificador y decodificador mencionado anteriormente lo podemos encontrar en esta referencia [Cho2015c]. El cual es totalmente *open source*.

Innovación

Bien, en este punto del estudio del arte hemos definido los distintos estudios que se han ido realizando a lo largo del tiempo, en primer lugar hemos visto las diferencias lingüísticas entre las lenguas, seguidamente hemos visto los primeros métodos de traducción que se empezaron a utilizar y después hemos visto los métodos que se están empezando a utilizar (redes neuronales) y que están influenciados por el tratamiento de imágenes (deep learning).

Además, como tema novedoso, podemos mencionar que cada vez se utiliza más el tratamiento de imágenes para utilizar estas redes neuronales. Esto es debido a que tal y como hemos explicado, a las redes neuronales les es relevante la cantidad de información que reciben.

Ahora vamos traspasar la línea de conocimiento al punto de empezar a definir que abarca este proyecto.

Todos estos estudios previos llevan a la conclusión de que puede ser muy interesante un estudio que se realice mediante distintas granularidades como ya mencionamos en el contexto de este proyecto, es decir, el carácter, la palabra y los bitmaps (tanto bitmaps de carácter y bitmaps de palabra), utilizando una red neuronal, concretamente la misma red neuronal que ya

está implementada para la utilización de palabras, lo cual es un buen punto de partida.

Si tomamos dicha implementación como punto de partida, será necesario adaptar este sistema para que sea capaz de interpretar caracteres en vez de palabras, es decir que cada representación (vector de bits aleatorio) que representa la palabra, ahora sea la representación de un carácter (vector de bits), y el conjunto de palabras que formaban la oración de origen (a cada iteración aleatoria) , ahora sea el conjunto de caracteres que forme la palabra origen a cada iteración.

De esta manera intentamos aportar mucha más información al sistema neuronal, lo cual debería afectar positivamente, ya que una de sus principales características es la mejora que tiene respecto a la cantidad de información que recibe.

Además, se pretende intentar llegar a un nivel un poco más profundo, es decir, si las representaciones de vectores de bits de 0/1 son aleatorias, una buena solución es generar una representación algo más inteligente. Como idea innovadora se intentará aprovechar la representación de los caracteres y las palabras chinas, para transformarlos a imágenes, donde seguidamente se pueda obtener el vector de bits llamado bitmap que representa dicha imagen obtenida de cada carácter o palabra correspondientemente. De esta manera no solo estamos aportando más información al sistema, sino que les estamos contribuyendo información mucho más informativa que un conjunto de valores aleatorios.

Formalización

Por lo que si definimos algo más formal, es decir, el procedimiento técnico que se realizará, podríamos decir que realizaremos un tratamiento de datos previo de la información justo antes de que la información entre al sistema y

se volverá ha implementaremos la entrada del sistema para que se adapte a nuestra nueva entrada de datos.

Por otro lado, este proyecto también intenta servir como nuevo recurso para investigadores, debido a la falta de recursos de la que se dispone en la actualidad, tal y como ya hemos mencionado. Concretamente, pretende tener una visión comparativa de los resultados que tienen las distintas granularidades ya mencionadas, de esta manera, sean resultados óptimos o no, puede dar una idea de que ramas pueden ser interesantes de cara a futuros estudios y cuáles son mejor olvidar.

Capítulo 2

Alcance, metodología y planificación

2.1. Formulación del problema

A pesar de todo el mancomunado esfuerzo colectivo, que se realiza para el desarrollo e investigación de nuevas tecnologías y logros algorítmicos, nos encontramos que a la hora de realizar una traducción automática, existen pares de lenguas que se encuentran totalmente desatendidas, tanto a nivel de investigación como comercial. Un claro ejemplo es la asociación chino-español, cuya traducción es muy utilizada en la actualidad, pero a penas podemos encontrar recursos, o bien investigación directa a este par de lenguas.

La traducción automática permite traducir idiomas instantáneamente. Hasta el momento, se utilizaban sistemas basados en segmentación, que aprendían por coocurrencias de palabras.

Últimamente, han aparecido los sistemas de traducción, basados en redes neuronales, que con representaciones vectoriales de las palabras permiten conseguir resultados similares a los previos obtenidos mediante sistemas de

segmentación.

Este proyecto, está orientado en sacar el máximo provecho de estos sistemas de traducción, basados en redes neuronales ya construidas, y así poder adaptar un sistema neuronal para el par de lenguas chino-español, usando como entrada al sistema diferentes granularidades (palabras, caracteres y bitmaps).

2.2. Alcance

Una vez explicado el problema que se intenta solucionar en este proyecto, podemos pasar a definir los objetivos que se pretenden garantizar.

El principal objetivo de este proyecto es el estudio de la traducción del par de lenguas chino-español, donde se estudiará cómo puede afectar la entrada de palabras y/o caracteres, al sistema neuronal, esto permitirá aprovechar las diferentes posibles granularidades de entrada al sistema mediante en el que incluirá la posibilidad de palabras, caracteres y bitmaps mediante la adaptación del código existente. En cuanto a los objetivos específicos que se abordarán serán:

- La primera toma de contacto con el entorno se basará en experimentación sobre distintos códigos ya construidos, donde:
    - En primer lugar, tomaremos como unidad mínima la palabra y experimentaremos dos cosas:
        - El sistema recibirá como entrada una palabra sin recibir ningún otro tipo de información.
        - El sistema recibirá la palabra, donde además utilizará información adicional, los caracteres de cada palabra introducida en el sistema.

    - En segundo lugar, tomaremos como unidad mínima el carácter, realizando así la segmentación del corpus. De este modo veremos si es mejor una red neuronal con un sistema de atención o bien sin sistema de atención.

- La segunda toma de contacto tratará de introducir una nueva idea al sistema, donde trata de convertir los caracteres en imágenes que posteriormente pasarán a bitmaps, lo que requiere adaptar este sistema ya construido para introducir este nuevo concepto. Nuevamente verificaremos esta entrada con los dos métodos siguientes:
    - Con un sistema de atención (con palabra y caracteres).
    - Sin un sistema de atención.
- Debido a que se trata de sistemas de código abierto para que cualquiera pueda contribuir información, ampliando así los recursos de la red, el tercer objetivo trata de aportar la información a los posibles usuarios mediante un artículo en un congreso internacional y repositorio github, donde para dicha conferencia es necesario realizar una presentación del estudio realizado.

2.3. Posibles obstáculos

Los posibles obstáculos y riesgos a los cuales se tendrá que hacer frente durante el desarrollo del proyecto son:

- Limitaciones de tiempo: La durada del Trabajo Final de Grado es limitada y es posible que durante el desarrollo de la adaptación aparezcan imprevistos técnicos, conceptuales o cualquier otro tipo que haga retrasar de manera inesperada el proyecto. Una buena base conceptual ajudará a evitar este riesgo.

- Limitaciones de conocimiento: Para obtener unos buenos resultados se debe conocer muy bien el funcionamiento de las redes neuronales, además de entender los código ya construidos para poder realizar adaptaciones. Es evidente que se necesita un tiempo de aprendizaje para que las adaptaciones tengan éxito.
- Limitaciones de ssh: Para realizar las traducciones se requiere de acceder a máquinas remotas (ssh), ya que se requiere de GPU, por lo que esto puede dar muchos problemas debido a:
    - La GPU es compartida, y solo una persona puede utilizarla al mismo tiempo.
    - Cada proceso de traducción requiere 4-5 días completos, por lo que un error al final, o ocupación de GPU por otra persona ralentizará el proceso.
    - El tiempo del proyecto es limitado por lo que si cada proceso dura 5 días no disponemos de muchos intentos teniendo en cuenta que es compartido.

2.4. Metodología y rigor

2.4.1. Método de trabajo

Como metodología del proyecto se seleccionará Scrum-Agile, con iteraciones frecuentes que tendrán tareas definidas. El motivo de la selección de esta metodología de trabajo es debido a la incertidumbre de los resultados obtenidos en las adaptaciones, ya que no podemos deducir que tan bueno o malo puede resultar, y si hay que reconducir estas ideas. De esta manera podemos garantizar que las limitaciones mencionadas anteriormente pueden ser superadas.

2.4.2. Herramientas de seguimiento

Para el correcto seguimiento del proyecto se utilizarà un gestor de repositorio online como es GitHub, el cual permitirá que el código del sistema esté disponible en la red. Para la documentación y conexión entre la directora del proyecto y yo se utilizará Google Drive, e-mail y dropbox. Y como herramienta de realización de documentación final Latex, debido a que se trata de un proyecto de investigación.

2.4.3. Métodos de validación

Para el proyecto se utilizará diferentes métodos de validación para comprobar que se avanza correctamente.

En primer lugar, se verificará cada experimento, es decir, se comprobará que los datos que han sido manipulados están completamente bien realizados, ya que cada error supone una pérdida de 6 días (tiempo que dura la ejecución)

En segundo lugar, después de cada iteración se valorará la completitud de la tarea definida y sus posibilidades de alteraciones junto a la directora del proyecto. Además habrá reuniones frecuentes, semanales con la directora de proyecto, con la finalidad de dar la aprobación final y aportación de nuevos cambios.

Finalmente se realizará la calidad de las traducciones mediante un *script* que compara el resultado esperado con el obtenido, viendo así en qué porcentajes mejora o si empeora.

2.5. Descripción de las tareas

2.5.1. Duración del proyecto

Aproximadamente, la duración del proyecto es de cinco meses, desde principios de enero, 11/01/16 hasta 27/06/2016, donde se realiza la lectura final. El proyecto se realizará por una persona, con el soporte de la tutora, la cual se encargará de orientar el proyecto.

2.5.2. Planificación y diagrama de Gantt

Para una buena comprensión del tiempo entre los distintos sprints (o iteraciones) se ha realizado un diagrama de Gantt que engloba todo el proyecto. Es muy importante considerar que a pesar de esta planificación, el tiempo puede variar al que se muestra en el diagrama, ya que la metodología Agile permite adaptarnos a lo que sea necesario en cada determinado momento. Aun así, cabe remarcar que es muy importante haber especificado el tiempo de las tareas con mucho rigor, para asegurar que todo funciona como esperamos o detectar un problema a tiempo. No definiremos un tiempo determinado por semana, sino que más bién marcaremos una tarea entre un periodo de días, y donde para esa tarea hay asignada una cantidad de horas. Hay que destacar que durante el Sprint 4.1 y Sprint 4.2 se dedicaran muchas más horas para poder obtener los resultados mostrados en el diagrama de Gantt.

Seguidamente vamos a definir cada uno de los sprints, donde podemos ver el cálculo por horas, pero el Gantt esta en días. Esto es debido a que entre los días indicados en el Gantt se realizarán estas horas para cada tarea correspondientemente.

Sprint 1: Estudio previo (30 h)

Debido a que se trabajará con técnicas muy concretas de inteligencia artificial, se debe comprender dichas técnicas, individualmente y de manera conjunta, con la finalidad de poder comprender los códigos ya construidos con estas técnicas y posteriormente poder ser capaz de realizar cambios de manera rigurosa.

Sprint 2: Análisis y Entorno (25 h)

Debido a que se utilizan códigos ya construidos, en los que se realizarán modificaciones. Se requiere configurar el entorno para que funcione como es debido en el entorno deseado. Por otro lado, también se deberá comprender cuál es el problema principal y como plantearse una solución.

Sprint 3: Experimentación con distintas granularidades(palabras y caracteres) (52 h)

En este sprint nos centraremos en realizar experimentación, de esta manera comenzaremos a comprender dichos códigos, y analizar los resultados obtenidos.

En este sprint se ha escogido realizar dos granularidades al mismo tiempo, ya que la granularidad de palabras está construida, y la granularidad de caracteres es bastante simple debido a que no requiere un tratamiento de datos de manera rigurosa, con lo que nos resulta útil para comenzar a ver el funcionamiento de nuestro código fuente.

Por otro lado, empezaremos a realizar la documentación de los datos obtenidos y los métodos utilizados.

Sprint 4.1: Introducir la granularidad de bitmaps (140 h) En este sprint nos centraremos en analizar cómo introducir como nueva granulari-

dad las bitmaps e integrarlos en nuestro sistema neuronal. Paralelamente se realizará la documentación de los datos obtenidos y la metodología utilizada.

Se deberá tener en cuenta que durante este sprint, el cual será bastante largo, se realizará otro sprint paralelamente, el 4.2, es decir, el sprint que se define como GEP, por lo que se dedicarán muchas más horas.

Sprint 4.2 : Gestión de Proyecto (GEP) (68.30 h) Esta asignatura asociada al Proyecto Final de Grado se realizará paralelamente con el sprint 4.1 como ya hemos comentado. Sus tareas son los diferentes entregables que hay definidos. Su calendario ha estado extraído de la guía de la asignatura, el cual asegura que la previsión de tiempo es correcta.

Sprint 5: Comparación de las distintas granularidades en redes neuronales con atención o sin. (25 h)

En esta fase nos centraremos en investigar qué granularidad aporta más información. Para ello deberemos utilizar los códigos con atención o sin, y un código que utilice las palabras con información de caracteres. Paralelamente se realizará una documentación sobre esta parte.

Sprint 6: Redactar artículo y subir a github (28 h)

Esta fase se centrará en redactar el artículo el cual se realizará junto con la subida de los códigos en Github, de manera que pueda servir como nuevo recurso en el ámbito de chino-español. Además se realizarán las transparencias para el workshop del contenido del artículo.S

Sprint 7: Documentación final (35 h)

En esta fase se realizará la revisión de todo el documento, donde ya se han ido redactando a medida que se realizaba la parte técnica, además se

revisará cualquier cosa no coherente.

Sprint 8: Transparencias de la presentación final (20 h)

En esta fase se realizará las transparencias que se utilizaran en la lectura de este TFG

Sprint 9: Revisión (15 h)

Este sprint tan largo se utilizará para tener en cuenta posibles retrasos, mejoras, o corregir errores.

2.5.3. Estimación de horas

La estimación de las horas de cada tarea se reflejarán en la siguiente figura 2.1.

Nombre del recurso	Horas de trabajo
Entregable 1: Alcance del proyecto y contextualización	24,50 horas
Entregable 2: Planificación Temporal	8,25 horas
Entregable 3: Gestión económica y sosteniblidad	9,25 horas
Entregable 4: Presentación preliminar	6,25 horas
Entregable 5: Revisión de las competencias de el TFG	1 hora
Entregable 6: Documento final	18,25 horas
Estudio previo	30 horas
Análisis del problema	15 horas
Configuración del entorno	10 horas
Experimentación con palabras	25 horas
Construcción de granularidad de caracteres	12 horas
Experimentación de granularidad de caracteres	15 horas
Construcción de granularidad de imágenes	60 horas
Integración en el sistema neuronal	45 horas
Experimentación de granularidad de imágenes	35 horas
Comparar las granularidades en distintas redes neuronales	25 horas
Redactar un articulo	25 horas
Subir el código al github	3 hora
Redactar el documento final	35 horas
Realizar las transparencias de la presentación final	20 horas
Revisión	15 horas
TOTAL	**438,3 horas**

Figura 2.1: *Estimación de horas.*

2.5.4. Recursos

Seguidamente se definirán los recursos necesarios para la realización de todo este proyecto, se reflejarán en la siguiente figura 2.2.

Recursos humanos	**Finalidad**
Una persona: emprendedor, analísta técnica y desarrollador con una dedicación de 25 h semanales	Desarrollar el proyecto
Recursos materiales	**Finalidad**
Ordenador portátil, Asus I7 con Linux	Desarrollar el proyecto
Terminal remota "ssh" a una GPU	Entrenar los modelos
Microsoft Project	Planificar el projecto (Gantt)
E-mail	Comunicación y seguimiento
TFS (Team Fundation Server)	Planificación del proyecto
GitHub	Control de versiones
Google Drive	Desarrollo de la documentación, herramienta de seguimiento
Softwares de redes neuronales	Desarrollar el proyecto
Corpus	Modelo de entrenamiento

Figura 2.2: *Recursos Necesarios.*

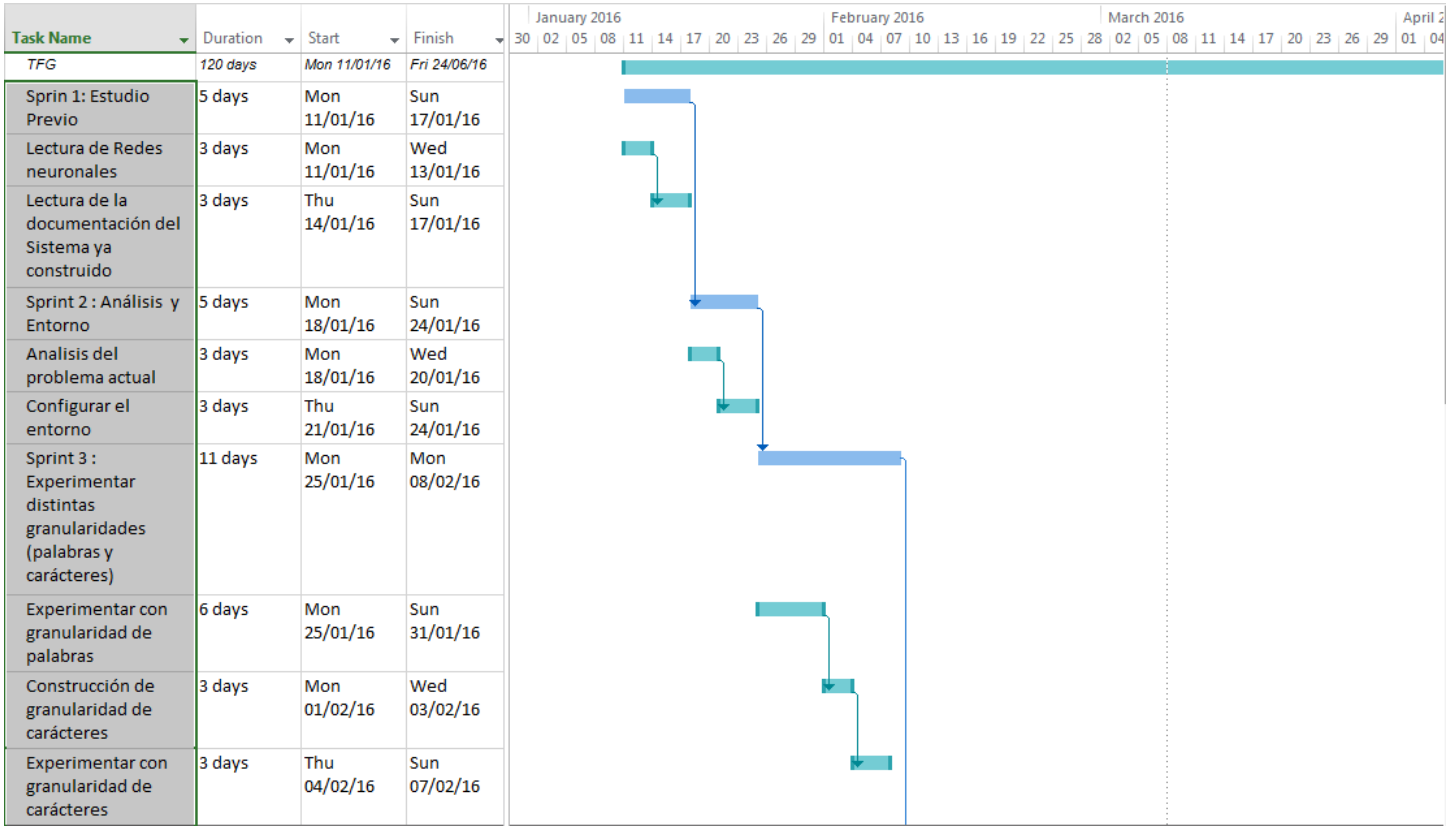
2.5.5. Valoración de alternativas y plan de acción

Como para el desarrollo de este proyecto se utiliza una metodología Agile-Scrum, al final de cada sprint se puede valorar como ha ido el sprint, y si se ha de cambiar alguna cosa de cara al futuro. Por lo tanto, en caso de que aparezcan problemas se podrían solucionar fácilmente y rápidamente. En ese momento, también se podría consultar con el tutor de trabajo que dirección coger y replantear los siguientes sprints.

Hay que tener muy en cuenta posibles modificaciones con los sprints ya que se utiliza GPU donde trabajamos con memoria compartida con otras personas, y además tenemos el factor de que cada entrenamiento requiere entre 5 y 6 días, lo que un error o cualquier posible incidente puede suponer una gran pérdida de tiempo. Debido a que es un problema que se basa en el

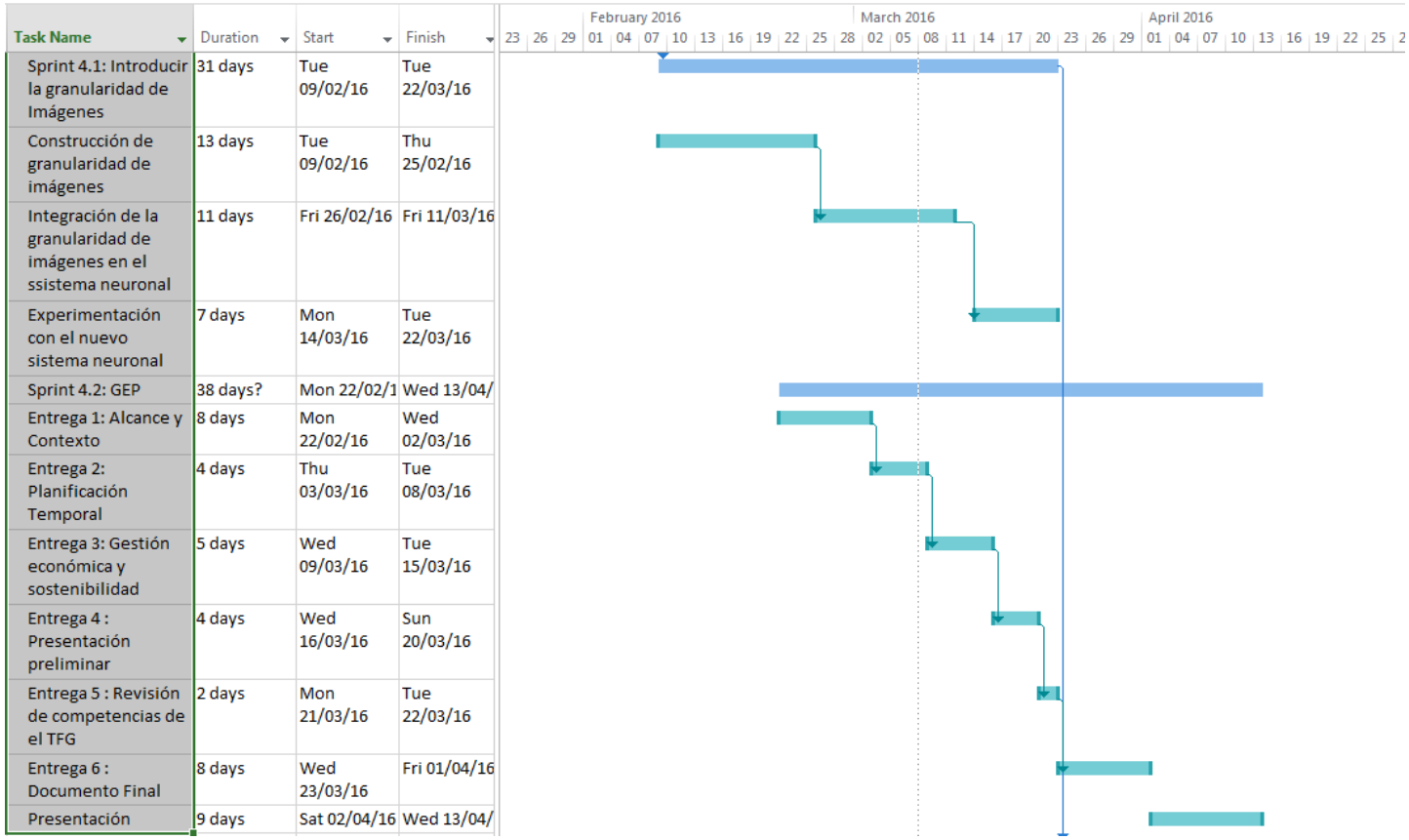
factor tiempo, el cual tiene un porcentaje de riesgo muy elevado, se tomaría la opción de alargar el proyecto a la siguiente lectura. A pesar de esta alternativa, también se ha empezado antes el proyecto con la finalidad de acabar con un margen de tiempo suficientemente bueno para posibles problemas.

2.5.6. Diagrama de Gantt



Task Name	Duration	Start	Finish
TFG	*120 days*	*Mon 11/01/16*	*Fri 24/06/16*
Sprin 1: Estudio Previo	5 days	Mon 11/01/16	Sun 17/01/16
Lectura de Redes neuronales	3 days	Mon 11/01/16	Wed 13/01/16
Lectura de la documentación del Sistema ya construido	3 days	Thu 14/01/16	Sun 17/01/16
Sprint 2 : Análisis y Entorno	5 days	Mon 18/01/16	Sun 24/01/16
Analisis del problema actual	3 days	Mon 18/01/16	Wed 20/01/16
Configurar el entorno	3 days	Thu 21/01/16	Sun 24/01/16
Sprint 3 : Experimentar distintas granularidades (palabras y carácteres)	11 days	Mon 25/01/16	Mon 08/02/16
Experimentar con granularidad de palabras	6 days	Mon 25/01/16	Sun 31/01/16
Construcción de granularidad de carácteres	3 days	Mon 01/02/16	Wed 03/02/16
Experimentar con granularidad de carácteres	3 days	Thu 04/02/16	Sun 07/02/16

Figura 2.3: *Gantt parte1*



Task Name	Duration	Start	Finish
Sprint 4.1: Introducir la granularidad de Imágenes	31 days	Tue 09/02/16	Tue 22/03/16
Construcción de granularidad de imágenes	13 days	Tue 09/02/16	Thu 25/02/16
Integración de la granularidad de imágenes en el ssistema neuronal	11 days	Fri 26/02/16	Fri 11/03/16
Experimentación con el nuevo sistema neuronal	7 days	Mon 14/03/16	Tue 22/03/16
Sprint 4.2: GEP	38 days?	Mon 22/02/1	Wed 13/04/
Entrega 1: Alcance y Contexto	8 days	Mon 22/02/16	Wed 02/03/16
Entrega 2: Planificación Temporal	4 days	Thu 03/03/16	Tue 08/03/16
Entrega 3: Gestión económica y sostenibilidad	5 days	Wed 09/03/16	Tue 15/03/16
Entrega 4 : Presentación preliminar	4 days	Wed 16/03/16	Sun 20/03/16
Entrega 5 : Revisión de competencias de el TFG	2 days	Mon 21/03/16	Tue 22/03/16
Entrega 6 : Documento Final	8 days	Wed 23/03/16	Fri 01/04/16
Presentación	9 days	Sat 02/04/16	Wed 13/04/

Figura 2.4: *Gantt parte2*

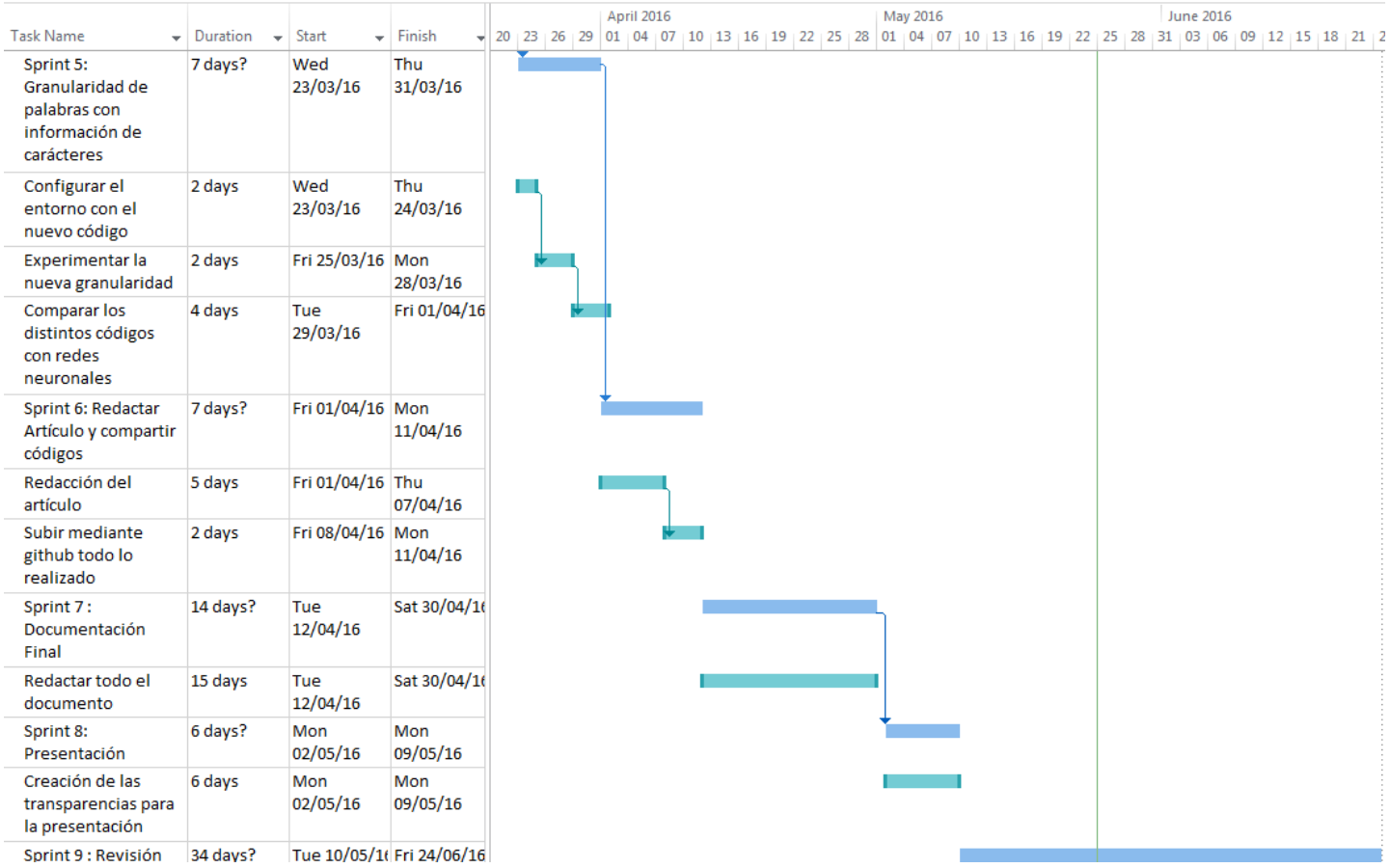


Figura 2.5: *Gantt parte3*

Capítulo 3

Gestión económica y Sostenibilidad

3.1. Gestión económica

3.1.1. Costes directos: recursos humanos

Este recurso ya se explicó en el capítulo 2. El cual decía que este proyecto estará desarrollador por una persona, la persona que lleva a cabo este proyecto. Se ha marcado el coste de la actividad de acuerdo al precio estimado que tendría en el mercado por realizar esta tarea, se muestran los resultados en la tabla 3.1.

3.1.2. Costes directos materiales

Ahora vamos a los costes materiales, donde se diferenciará en hardware y software. Seguidamente se mostrará su coste en la tabla 3.2.

Tarea	Unidades	Precio mercado	Coste mercado
Sprint 1	30h	20€/h	600€
Sprint 2	25h	20€/h	500€
Sprint 3	52h	20€/h	1040€
Sprint 4.1	140h	20€/h	2800€
Sprint 4.2	68,30h	20€/h	1370€
Sprint 5	25h	20€/h	500€
Sprint 6	28h	20€/h	560€
Sprint 7	35h	20€/h	700€
Sprint 8	20h	20€/h	400€
Sprint 9	15h	20€/h	300€
TOTAL	438,30h	20€/h	8.770€

Tabla 3.1: *Costes directos: recursos humanos.*

Producto	Unidades	Precio unitario	Porcent. Dedicación	Coste estimado
Intel inside core i7	5 meses	600€/4años	80 %	50€

Tabla 3.2: *Costes directos materiales.*

Hardware

Tenemos un único coste de hardware, el ordenador con el que se realiza el desarrollo del proyecto. Supondremos que tiene una vida útil de 4 años y que el 80 % de su uso irá dedicado al proyecto que en los 5 meses que dura.

Software

Durante el desarrollo del proyecto, el coste del software es 0. En un futuro, post proyecto, este podría tener costes asociados a contratar licencias necesarias para el uso profesional de algunos softwares, pero queda fuera del alcance de este proyecto. La tabla 3.2 muestra dichos costes.

$Cost.Estimado = \frac{PrecioUnitario}{N^{o}meses} * Unidades * Porcent.Dedicacion$

3.1.3. Presupuesto

El presupuesto realizado para este proyecto se muestra en la tabla 3.3

3.1.4. Control de gestión

Para el control de software y hardware no podemos hacer nada más que anotar si ha cumplido alguno de los imprevistos mencionados. Por otro lado, para el control de los recursos humanos, se mantendrá un registro de horas trabajadas en cada una de las tareas con la finalidad de ver posibles desviaciones temporales que pueda sufrir el proyecto a final de cada sprint y de

Concepto	Coste
Costes directos recursos humanos	8770€
Costes directo materiales	50€
Costes indirectos	156€
Contingencia	1072€
Imprevistos	30€
TOTAL	10078€

Tabla 3.3: *Presupuesto.*

aquí poder calcular la desviación en el presupuesto. En los costes indirectos se podrá calcular la desviación mediante los folios impresos con los previstos. Finalmente se agruparan todos estos costes en una tabla y se comprobará si se cubren o no los imprevistos o bién se habido alguna causa no mencionada anteriormente. También debe tenerse en cuenta si el inicio de contingencia lo cumple.

3.2. Sostenibilidad

En primer lugar mostraremos la tabla de sostenibilidad realizada para este proyecto en 3.1, donde seguidamente haremos un pequeño apartado para cada uno de los valores.

3.2.1. Sostenibilidad económica

Se ha hecho un estudio de los costes, tanto de los recursos humanos como los materiales, además se ha calculado los posibles imprevisto que podrían surgir.

Sostenibilidad	Económica	Social	Ambiental	TOTAL
Panificación	Viabilidad económica	Mejora en la calidad de vida	Análisis de recursos	
Valoración	7	9	6	22

Figura 3.1: *Tabla sostenibilidad.*

Debido a que el proyecto se realiza mediante código abierto para posibles mejoras en el campo y en campos relacionados, no se cuenta con obtener beneficios económicos, más bien sociales.

Respecto al coste y tiempo de proyecto sería bastante difícil de mejorar, ya que el tiempo está bastante limitado. Se ha tenido en cuenta que que tengan más horas de dedicación los sprints más importante como vendría a ser la integración de imágenes.

3.2.2. Sostenibilidad social

Los objetivos del proyecto son dos principalmente. Por un lado, aportar una gran cantidad de información de contraste entre diferente granularidades de segmentación a la hora de realizar una traducción de chino a español. Por otro lado, aportar una idea innovadora que permita obtener mejores resultados, el cual a posteriori se pueda investigar en ella en profundidad debido a los resultados mostrados en este proyecto. Todo esto sería útil en el sector de la traducción o bién campos relacionados con lectura de texto VS lectura de imagen, con la finalidad de mejorar estos sectores. La mejora del sistema de traducción es bastante necesaria hoy en día (concretamente chino-español) debido a que mejoraría mucho la comunicaciones entre los continentes Asiático y América del Sud los cuales están desatendidos.

3.2.3. Sostenibilidad ambiental

Los recursos del hardware consumen recursos en su construcción y gastan electricidad cuando se utilizan. Los recursos de software no consumen directamente, pero al utilizarlos es necesario utilizarlos a través de los recursos de hardware. La única contaminación que generan es la indirecta, debido a la electricidad que es necesaria. Por otro lado, si este proyecto se hubiese realizado sin TFG, se habrían ahorrado la impresión de los folios prevista.

Como comentario final diremos que este proyecto no aumenta ni disminuye la huella ecológica, ya que no se necesitan materias primas.

Capítulo 4

Descripción teórica

Debido a que este trabajo final de grado utiliza un código libre y abierto, es decir, se realiza una modificación de un código ya existente. Es completamente necesario aprender y explicar cómo funciona, para que de este modo sepamos qué estamos modificando y como influye al sistema. (Se concretará en las partes que son necesarias entender para la modificación, todas aquellas partes que no se vean alteradas ni requieran de un conocimiento profundo para el proceso realizado en este estudio se pasarán más por encima)

En primer lugar se explicará la descripción del sistema de traducción que se utiliza, que está compuesto por:

- El corpus que se utiliza, con sus características.
- Cómo se generan los diccionarios que utiliza la red neuronal
- Cómo funciona el sistema neuronal.

En segundo lugar hablaremos de la segmentación e inclusión de bitmaps que se ha realizado al sistema, es decir:

- Separación de palabras en caracteres

- Tratamiento de bitmaps para el sistema neuronal
- Inserción de bitmaps en el sistema neuronal

Antes de empezar con cada uno de los apartados dire que los códigos con los que se trabaja en este proyecto se encuentran en la en la referencia de [Cho2015c] [1].

4.1. Descripción del sistema de traducción basado en redes neuronales

Tal y como ya se ha comentado, se explicará cómo se realiza la traducción actual para la generación de los entrenamientos de palabras. Primeramente se mostrará el esquema previo en la imagen 4.1, el cual se realiza en este proyecto, de esta manera podemos hacernos a la idea de lo que se habla, y seguidamente explicaremos cada paso detalladamente.

Una vez realizada la descripción de la red neuronal, se obtiene el modelo de entrenamiento, donde seguidamente se realiza la traducción de los resultados obtenidos de la red neuronal, pero no se han mostrado en el esquema ya que esta parte no se ve alterada, tan solo se modifican los *path* del código para utilizar el corpus deseado. De esta manera se realiza el procedimiento completo.

4.1.1. Corpus

En este trabajo se utiliza el corpus paralelo Chino-Español, United Nation Corpus (UN). Donde las estadísticas de nuestro corpus las podemos encontrar en la tabla 5.1

[1] https://github.com/nyu-dl/dl4mt-tutorial

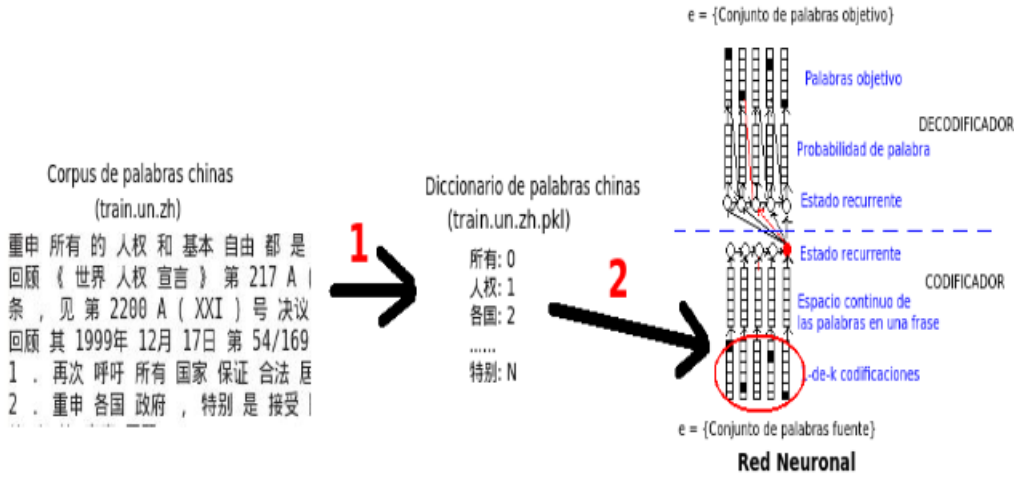


Figura 4.1: *Descripción del sistema de traducción basado en redes neuronales.*

Existen varios motivos por los que se ha elegido trabajar con este corpus y no con otro:

- Este corpus es totalmente libre, a diferencia de otros que no son totalmente de uso libre.
- Las dimensiones del corpus son muy importantes a la hora de trabajar, ya que como se ha mencionado la cantidad de datos que entra en una red neuronal es relevante. Podríamos haber trabajado con un corpus más grande, eso implica que el tiempo de entrenamiento crecería mucho, es decir, si ahora la ejecución del entrenamiento nos tarda una semana.

Una vez tomada esta decisión y teniendo claro cuales son las características de nuestra base podemos avanzar al siguiente paso del entrenamiento de la red neuronal.

L	Set	S	W	V
ES	Train	58.6K	2.3M	22.5K
	Dev	990	43.4K	5.4k
	Test	1K	44.2K	5.5K
ZH Palabras	Train	58.6K	1.6M	17.8K
	Dev	990	33K	3.7K
	Test	1K	33.7K	3.8K
ZH Caracteres	Train	58.6K	2.8M	3.8K
	Dev	990	53.9K	1.7K
	Test	1K	55.1K	1.7K

Tabla 4.1: *Detalles del Corpus. Número de frases (S),palabras (W), vocabulario (V)*

4.1.2. Generación de diccionarios

Este segundo paso, es el que se encarga actualmente de preparar la entrada al sistema, es decir, el programa que genera los diccionarios toma como entrada el corpus detallado anteriormente, seguidamente realiza un algoritmo donde crea un diccionario ordenado por frecuencias de palabras, de mayor a menor. De esta manera el sistema dará más importancia a las palabras con más apariciones para una mejor traducción. Para hacernos una idea la estructura de datos generada de este proceso se puede ver el la figura 4.2

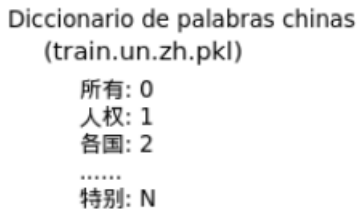


Figura 4.2: *Diccionario de Palabras.*

Donde la palabra con frecuencia más alta siempre se encuentra en la posición 0 y la menos frecuente en la posición N.

4.1.3. Sistema de referencia

El sistema neural se encarga de realizar el entrenamiento de manera completa. Nosotros nos centraremos en la parte inicial, ya que las partes posteriores de la red neuronal y consiguientemente su traducción no se ven afectadas por toda la ampliación que se explicará.

En primer lugar la inicialización de parámetros es la que se encarga de dar las características del sistema, donde principalmente a nosotros lo que nos interesa son:

- Las dimensiones de los vectores de bits aleatorios que genera la red neuronal
- El tamaño de nuestro vocabulario objetivo, es decir, la suma de las palabras del chino más las del español.
- El tamaño de nuestro vocabulario origen, es decir, la suma de las palabras del chino más las del español.

En segundo lugar, si entramos de lleno en el código, se debe localizar en qué punto se realizan las inicializaciones de los vectores de bits aleatorios, los llamados 1-de-k codificaciones, los cuales son de interés para nosotros, ya que posteriormente, en la inclusión de información de bitmaps en el sistema de traducción se deberá modificar.

Seguidamente se realiza todo el tratamiento neuronal, el cual tiene un algoritmo a muy bajo nivel en el que no se detalla, debido a que no se han realizado ningún tipo de modificación en esta parte, donde dicho sistema obtiene como resultado un modelo de entrenamiento. Para más información

sobre el funcionamiento de la red neuronal utilizada se puede ver la documentación en las referencias: [Cho2015a], [Cho2015b] y [Cho2015c].

El sistema realizado es como el que se muestra en la figura 4.3, donde hay que tener en cuenta que nuestro sistema tiene un sistema de atención añadido.

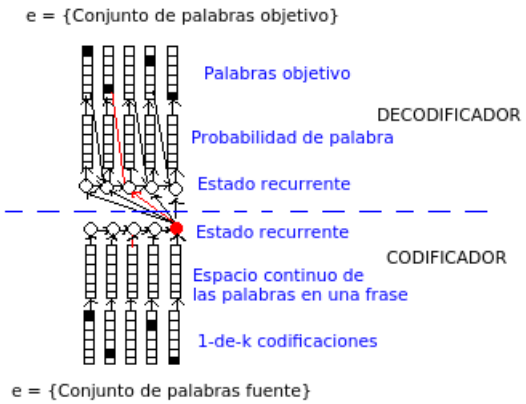


Figura 4.3: *Decodificador-Codificador.*

4.2. Estrategias de segmentación e inclusión de información de bitmaps en el sistema de traducción basado en redes neuronales

En este punto de la memoria, se explicará detalladamente cual son los cambios que se han realizado para la traducción basada en redes neuronales explicado anteriormente, además de razonar dichos cambios.

En la figura 4.4 se muestran las estrategias realizadas en este proyecto y explicaremos cada uno de los pasos en los apartados siguientes:

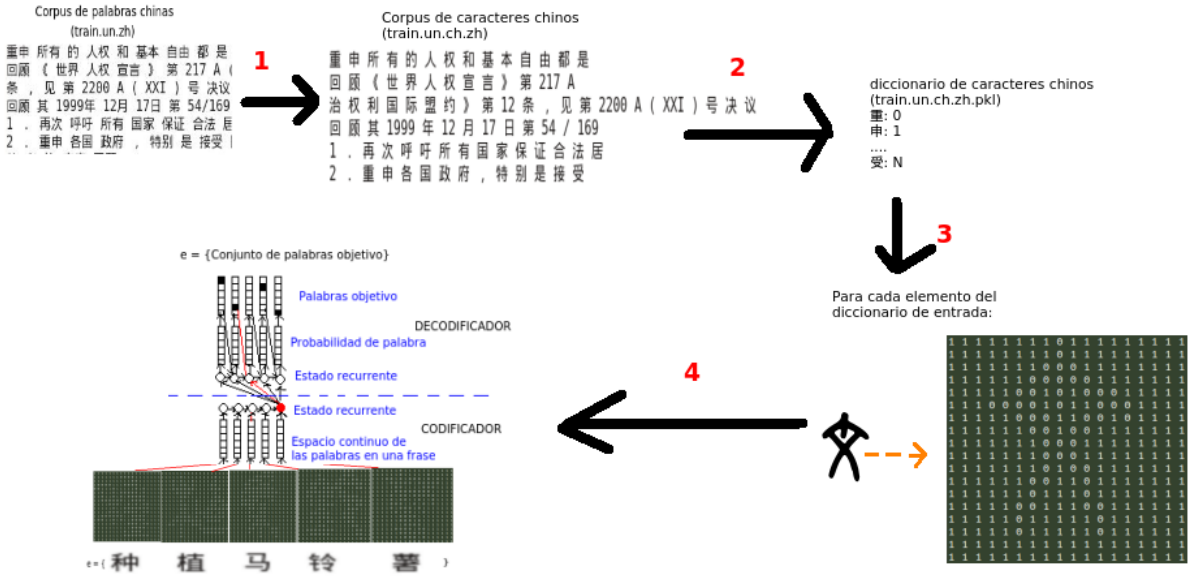


Figura 4.4: *Estrategias de segmentación e inclusión de bitmaps.*

4.2.1. Separación de palabras en caracteres

Como es obvio el corpus de partida no ha cambiado, por los que sus características tampoco.

En primer lugar, utilizando dicho corpus hemos realizado un *script* que separe todo el corpus en caracteres, de esta manera se adquiere un vocabulario mucho menor y por contra aumentamos las palabras. La minificación del vocabulario es muy relevante, ya que el hecho de tener un vocabulario muy grande implica que el número de palabras desconocidas en nuestro sistema es mucho mayor, lo cual afecta muy negativamente en el sistema. Por lo que haber reducido el vocabulario implica haber reducido el número de unidades mínimas, caracteres.

A todo esto, hay que mencionar que el hecho de separar en caracteres quiere decir que nuestro sistema trata una entrada de longitud uno, mientras que la salida tiene longitudes mayores ya que son las palabras traducidas al español. Esto quiere decir que estamos haciendo más grande la diferencia entre la longitud de entrada y la de salida, y esto es negativo para el sistema.

Seguidamente este nuevo corpus creado de manera artificial, se procesa de la misma manera que se hace en el generador de diccionarios, sin ninguna modificación, ya que no le importa lo que entre (palabras o caracteres), simplemente trata cada carácter como si fuese una palabra y los ordena por frecuencias de mayor a menor.

Finalmente se genera como salida una estructura de datos mucho menor a la generada cuando se procesan palabras (diccionario).

4.2.2. Tratamiento de datos del sistema neuronal

En este punto tenemos generado el diccionario de caracteres ordenados por frecuencia, donde a diferencia de la descripción teórica de la traducción, ya explicada en apartados anteriores, en el que pasaban directamente al sistema neuronal. Esta vez pasamos el diccionario por un tratamiento de la estructura de datos del diccionario, el cual es previo a dicho sistema neuronal.

El objetivo de este tratamiento de datos es la transformación de cada carácter del diccionario en imagen donde seguidamente se convierten en bitmaps, para ello hemos tenido que tener en cuenta el algoritmo que realiza el sistema neuronal, ya que debemos generar la salida adecuadamente para que el sistema neuronal pueda integrar correctamente.

En primer lugar, hemos transformado el diccionario en una lista ordenada por frecuencia de mayor a menor, para poder iterar sobre ella, donde cada elemento de la lista contiene el carácter adecuado y su posición de la lista indica que es más frecuente que la siguiente.

Una vez obtenida la lista, se procesa cada unidad del diccionario, y se transforma cada elemento a imagen. Para transformar cada elemento a imagen es necesario tener instalado el *package* que muestra la figura 4.5.

Figura 4.5: *Fuente para trabajar con caracteres chinos.*

El cual mediante la librería de python cairo es posible realizar dicha conversión. En la figura 4.6 se mostrará un pqueño ejemplo.

```
import cairo

# adapted from
# http://heuristically.wordpress.com/2011/01/31/pycairo-hello-world/

# setup a place to draw
surface = cairo.ImageSurface(cairo.FORMAT_ARGB32, 100, 100)
ctx = cairo.Context (surface)

# paint background
ctx.set_source_rgb(1, 1, 1)
ctx.rectangle(0, 0, 100, 100)
ctx.fill()

# draw text
ctx.select_font_face('金文大篆体')
ctx.set_font_size(80)
ctx.move_to(12,80)
ctx.set_source_rgb(0, 0, 0)
ctx.show_text('我')

# finish up
ctx.stroke() # commit to surface
surface.write_to_png('我.gif')
```

Figura 4.6: *Código para generar bitmaps.*

Una vez ejecutado este código en python, obtenemos imagenes como las de la figura 4.7

Como podemos ver en el código se genera un rectángulo de $10X10$ es decir una matriz de 10,000 elementos. Dichos elementos son valores entre 0 y 255 por lo que se tienen que escalar a 0 y 1, en nuestro caso es bastante sencillo, ya que cada valor superior a uno tendra el valor 1 y los de valor 0 no cambiarán.

Figura 4.7: *Letra China.*

Debido a que el tamaño es muy grande, ya que el sistema neuronal está pensado para tamaños de 512 bits (1-de-K codificaciones) , se ha reducido el tamaño de dicha matriz a $23x23$, obteniendo una resolución de 529 bits, se ha intentado de ajustar al máximo, ya que hay estudios que la longitud experimental ideal ronda sobre esta cifra, eso implica que se ha asumido que no es necesario experimentar demasiado en ese punto de proyecto.

Se realiza este proceso para cada elemento de la lista, pero se ha encontrado que no todos los caracteres funcionan correctamente, concretamente había problemas en algunos signos de puntuación a la hora de generar imágenes, por lo que se ha realizado un híbrido, es decir, por híbrido entendemos generar vectores de bits a partir de la imagen, y por otro lado vectores aleatorios, para aquellos que no son posible de generar mediante imágenes, los cuales no son demasiados.

Finalmente se obtiene una matriz en la que cada posición hay un vector de bits inteligente (con información del carácter), donde en la posición 0 está el vector del elemento más frecuente y en la N la menos frecuente.

4.2.3. Integración de bitmaps

Ahora volvemos a encontrarnos que principalmente hay que modificar la parametrización y seguidamente el proceso del sistema neuronal en sí.

En primer lugar, en la parametrización, hemos de indicarle que nuestros vectores de 1-de-k codificaciones tendrán como tamaño de entrada el tamaño

que se haya elegido en el pre procesamiento de datos. Además, descartamos todas aquellas palabras cuya frecuencia sea 1 por lo que nuestro sistema se focaliza más en resolver el resto. Por otro lado, hay que tener en cuenta los *paths*, ya que ahora tomaremos el que hemos generado con nuestro *script* para obtener un corpus de caracteres.

En cuanto al propio sistema, hemos de quitar la inicialización aleatoria de los vectores 1-de-k codificaciones que se realiza, ya que ahora nosotros tenemos la matriz inicializada con los valores obtenidos de las imágenes, por lo que la estructura neuronal queda reducida a la imagen 4.8

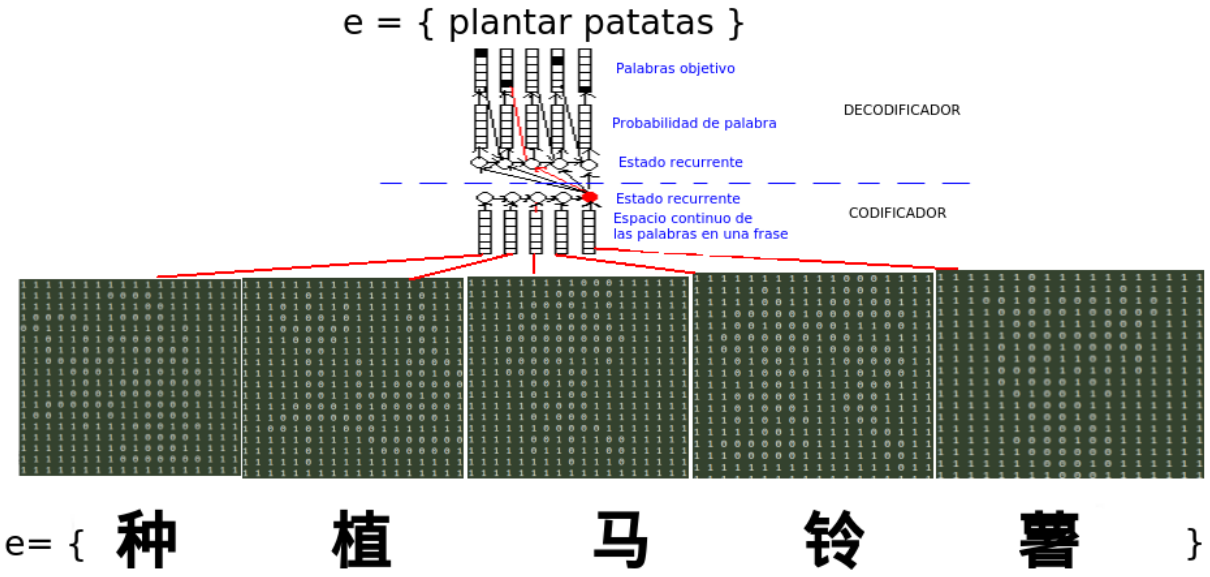


Figura 4.8: *Nuevo sistema de traducción automática neuronal.*

Con estas nuevas incorporaciones menciones es posible ejecutar todo este nuevo proceso, el cual permite obtener como resultado la traducción final. Cabe remarcar que este proceso está realizado de manera generalizada, es decir, es posible partir desde el corpus de palabra sin pasar por el *script* que crea el corpus de caracteres, generando directamente los diccionarios o bien

crear el corpus de caracteres, de este modo es posible generar imágenes tanto para caracteres o bien para palabras. Lo único que es necesario modificar es el tamaño de las imágenes y el tamaño de texto en la imagen para que se adapte correspondientemente.

Todo esto se encuentra totalmente disponible en github[2].

[2]https://github.com/aldomin/NMTbitMaps

Capítulo 5

Experimentación

En este capítulo hablaremos sobre todos los experimentos que se han realizado a lo largo de este proyecto con el objetivo de ver qué vías son mejores de cara a proyectos futuros.

Principalmente este capítulo se divide en cuatro grandes puntos, donde los tres primeros puntos intentan encontrar que tipo de vía es mejor para obtener buenos resultados, y el último punto intenta magnificar esos resultados. (El tiempo de realización del experimento aumenta correspondientemente a estos cuatro puntos).

1. En primer lugar hablaremos de los experimentos que se han realizado para los sistemas que no tienen atención añadida, el cual era bastante obvio que sus resultados no serían buenos, pero debido a que trabajan mucho más rápido que los sistemas con atención, nos era útil para poder ver la correlación de los resultados y ver cómo impactarían en un sistema con atención añadida.

2. En segundo lugar se mostrarán los resultados obtenidos en los sistemas con atención añadida, los cuales tratan de experimentos más largos, debido al incremento exponencial de cálculos añadidos (a causa de la

integración de la red bidireccional añadida al sistema), los cuales mostrarán resultados mejores a los existentes. Hay que remarcar que este proyecto está centrado sobre este experimento, ya que era la idea principal, ya que es imposible abarcar todos los experimentos en un tiempo tan reducido.

3. En tercer lugar se intenta experimentar con un sistema ya construido (adaptación del sistema inicial que hemos utilizado), el cual intenta dar a las palabras información de caracteres.

4. Finalmente, como apartado extra de este proyecto se ha intentado empezar a experimentar con un corpus más grande, es decir, con un vocabulario de palabras y caracteres mucho mayor. Este experimento ha requerido de mucho tiempo, donde se ha realizado mediante granularidad de palabras.

   En los primeros apartados explicaremos en detalle el proceso de entrenamiento detalladamente, despueś reduciremos muchas anotaciones ya que todo sigue la misma metodología.

   Antes de adentrarnos en cada uno de los apartados, mostraremos una pequeña tabla que recoge todos los resultados 5.1, de manera que de cara al lector le sea más fácil compactar los resultados y dirigirse a la sección deseada.

Experimentos	Tipo	BLEU	1PC	2PC	3PC	4PC
Sin Atención	Sist. de ref. caract.	1.54	7.3	2.2	0.9	0.4
	Información de bitmaps de caract.	2.29	11.1	3.4	1.3	0.6
Con Atención	Sist. de ref. de palab.	5.55	36.0	9.1	3.3	1.7
	Sist. de ref. de caract.	5.52	35.2	8.8	3.5	1.9
	Inf. de bitmaps de caract. $40x40$	4.08	29.8	6.1	2.4	1.3
	Inf. de bitmaps de caract. $23x23$	5.72	32.5	8.5	3.5	1.8
	Inf. de bitmaps de palab.	8.49	38.1	11.3	4.9	2.7
Inf. Caract.)	Sist. de ref. caract.	5.01	35.5	8.6	3.5	1.8
	Información de bitmaps de palab.	4.61	29.4	6.5	2.4	1.2
Con Atención (Corp. Grande)	Sist. de ref. caract.	21.71	57.1	30.2	18.9	12.6
	Información de bitmaps de palab.	23.55	52.4	28.3	17.8	11.7

Tabla 5.1: *Detalles de los Resultados. Tipo de experimento (T), Porcentaje de traducción correcta (BLEU) 1 palabra bien traducida consecutiva (1PC), 2 palabras bien traducidas consecutivas (2PC), 3 palabras bien traducidas consecutivas (3PC), 4 palabras bien traducidas consecutivas (4PC)*

5.1. Sistema sin atención añadida

Para estos experimentos se ha utilizado el código que encontramos en la referencia de [Cho2015c] [1] de la Sesión 1, el cual usa un sistema sin atención.

Es cierto que en el sistema de referencia se obtienen malos resultados, pero debido al tiempo del que se dispone, y qué realizar estos experimentos no requiere de mucho tiempo, un dia y poco para cada uno. Por lo que hemos considerado que era de gran utilidad realizarlos por varios motivos:

1. En primer lugar nos dan una idea de cómo impactan estos resultados en sistemas más complejos, los sistemas con atención añadida.

[1] https://github.com/nyu-dl/dl4mt-tutorial

2. Nos permite ver la correlación que hay entre los resultados de referencia y la integración de bitmaps. Para así evitar perder mucho tiempo en los sistemas más complejos, ya que estos sistemas complejos requieren más de una semana como mínimo.

Lo que quiero transmitir es que utilizar estos sistemas a medida que se construye el tratamiento de datos es mucho mejor. De este modo nos hacemos a la idea de que podrá tener impacto en un corpus con vocabulario mucho más grandes, evitando pérdidas de tiempo en dichos sistemas.

5.1.1. Sistema de referencia

En este sistema de referencia hemos puesto los *paths* correctos, es decir, apuntando a nuestro corpus, y además hemos realizado la segmentación de caracteres. Ya que de esta manera estamos trabajando todo el proceso completo y probar su funcionamiento correctamente, del proceso construido.

Respecto al resultado de traducción obtenido podemos ver que la traducción es bastante pésima con el sistema de referencia que se utiliza, sin atención. Algo que ya era bastante evidente, debido a la falta de la red bidireccional añadida, la cual se encarga de informar mucho más a nuestro sistema.

5.1.2. Sistema de referencia usando información de bitmaps

En segundo lugar, hemos realizado como estrategia de segmentación la inclusión de información de bitmaps en el sistema de traducción basado en redes neuronales (sin atención), de esta manera vemos que tan buena puede ser la técnica de cara a los siguiente experimentos (mediante caracteres).

Como podemos ver la traducción en sí sigue siendo bastante desastrosa, pero por contra vemos que el porcentaje de traducción ha mejorado casi en 1, lo cual hace creer que en el momento que apliquemos esto a un sistema con atención, estos resultados se podrían magnificar.

5.1.3. Ejemplos comparativos sin atención

Seguidamente mostraremos en la tabla 5.2 ejemplos de frases del sistema de referencia con el sistema de bitmaps, para ver en qué mejora (Aunque es difícil observar las mejoras debido a la baja calidad de solución)

	Type	Sentence
1	Src	55/14 . 大会议事规则第
	CH	UNK , **la Asamblea General** en el párrafo 5
	+Bitmap	reglamento , **de la Asamblea General** en el párrafo 5
	Ref	reglamento **de la Asamblea General**
2	Src	(e) 小口径弹药;
	CH	b) UNK el UNK
	+Bitmap	e) Los territorios no autónomos ;
	Ref	e) Municiones de pequeño calibre ;
3	Src	58/147 . 妇女的家庭暴力行为
	CH	56 / UNK
	+Bitmap	discriminación **contra** el fomento de discriminación contra el fomento
	Ref	violencia **contra** la mujer en el hogar
4	Src	(e) 法律和司法体制;
	CH	e) UNK ;
	+Bitmap	e) Los territorios no sea posible ;
	Ref	e) las instituciones jurídicas y judiciales ;

Tabla 5.2: *Frases de ejemplo. Origen (Src), sistema de referencia (CH), inclusión de Bitmap (+Bitmap), Objetivo (Ref)*

Como podemos ver, en el resultado con segmentación de caracteres aparecen palabras desconocidas, mientras si utilizamos la incrustación de bitmaps podemos ver como desaparecen los caracteres desconocidos (Elementos UNK, son palabras desconocidas).

La conclusión de este experimento nos hace creer que los resultados podrían aumentar con la utilización de un sistema con atención, por lo que la probabilidad de que suponga una pérdida de tiempo en nuestro proyecto es prácticamente nula.

5.2. Sistema con atención añadida

Este experimento es en el que más se ha trabajado en este proyecto, ya que era la esencia del proyecto. Su importancia es debido a que hay estudios previos que muestran que es mucho mejor trabajar con sistemas de atención añadidos que no sin, pero el tiempo computacional de cálculos es mucho más elevado, por eso hemos de medir con lupa las probabilidades de éxito. (Hay que remarcar que el artículo de investigación realizado para este proyecto está centrado en este punto, donde se puede ver detalladamente todo)

Este experimento, a diferencia del anterior, se han lanzado 5 ejecuciones en total, ya que se ha intentado analizar en detalle cualquier posible observación,error, mejora,etc . Antes de entrar en detalle en cada uno, comentaremos por encima de que trata cada uno de ellos:

- La primera ejecución fue el sistema de referencia, el cual trabaja con la granularidad de palabras, es decir, este valor nos muestra el resultado que hay hoy en dia en la traducción neuronal de chino a español, tomando como referencia el corpus que se utilizará. Ya que como es obvio este valor variará en función del corpus.

- La segunda ejecución trata sobre obtener el sistema de referencia, pero aplicando la segmentación de caracteres, tal y como ya se explicó su realización en apartados anteriores.

- En esta prueba se hizo los mismo que el párrafo anterior, pero además

se realizó la integración del tratamiento de bitmaps, tomando como resolución de la matriz de bitmaps un tamaño de $40x40$.

- Esta prueba sigue exactamente el mismo procedimiento que el anterior, pero se varía el tamaño de la resolución de bitmaps, reduciendo la matriz a $23x23$.

- Finalmente, la última prueba evita de hacer la segmentación de caracteres, pero mantiene la integración de bitmaps, es decir, codifica 1-de-K codificaciones como bitmaps de palabras.

Para detallar las 5 pruebas de este experimento realizado, hemos decidido dividirlo en dos secciones que detallan sobre los resultados, proceso y motivos. El primer apartado incluye las dos primeras pruebas, y el segundo apartado los tres últimos. Posteriormente hay apartado que muestra ejemplos de los resultados.

5.2.1. Sistema de referencia con segmentación de caracteres o palabras

En este apartado, tal y como ya hemos comentado, trataremos las 2 primeras pruebas.

En primer lugar tenemos el sistema de referencia, el cual trabaja con palabras. En esta prueba se ha ajustado el vocabulario (la suma de el vocabulario del español y el chino) para que nuestro sistema de referencia sea correcto, y a sí pueda generar todas las palabras posibles.

Como resultados tenemos:

BLEU = 5,55, 36,0/9,1/3,3/1,7

Esta prueba es la que se debe fijar como base, ya que cualquier prueba que sea igual o inferior querrá decir que no estamos aportando ninguna mejora al sistema.

Una vez visto este punto importante, como segunda prueba se realizará la segmentación de caracteres, tal y como se vio en los sistemas sin atención, deberíamos esperar una mejora debido a que los sistemas neuronales muestran buenos resultados a más cantidad de información, pero no demasiada.

En esta segunda prueba como parámetros hemos vuelto a fijar el vocabulario pertinente (suma del chino más el español). Además como punto importante hemos marcado que se centre en la palabras que aparecen más de una vez en el texto, ya que hay una gran cantidad de palabras que solo aparecen una vez y por lo que son menos importantes que las que se repiten muchas veces. En otras, palabras, hemos intentado reducir el vocabulario de origen, obviando el vocabulario de aquellos caracteres que no tienen relevancia en el sistema.

Como resultados de esta prueba tenemos:

BLEU = 5,52, 35,2/8,8/3,5/1,9

Como podemos ver los resultados son mínimamente peores que los de referencia, pero no los debemos descartar aún, ya que tan solo se ha aplicado una técnica de segmentación y hemos conseguido mantener los resultados, por lo que en el momento de realizar una inclusión de bitmaps deberíamos poder obtener una mejora en cuanto al sistema de referencia.

5.2.2. Integración de información de bitmaps con segmentación de caracteres o palabras

Este apartado es el que muestra las tres últimas pruebas mencionadas, donde tratan sobre la inclusión de los bitmaps en el sistema neuronal.

Las das primeras pruebas que se han realizado son prácticamente similares, ya que lo que intentan es ver cual es el valor óptimo de la imagen para las redes neuronales. Se intenta valorar tanto la pérdida de resolución de la imagen como puede afectar. O bién la longitud de los bitmaps como puede

impactar a los vectores de 1-de-k codificaciones.

En primer lugar, se realiza una segmentación de caracteres, ademaś transformar los caracteres a bitmaps. Respecto a los parámetros volvemos a marcar el tamaño del vocabulario y las palabras en que nos queremos centrar, y finalmente eliminamos la parte del sistema neuronal que crea los 1-de-k codificaciones aleatorias y integramos los bitmaps. Respecto al tamaño de los bitmaps óptimo (por óptimo entenderemos que la imagen tiene una visualización perfecta) deberíamos hacer bitmaps de 10.000 bits, pero debido a que el sistema neuronal no soporta longitudes tan largas, debido a la memoria que necesita para realizar operaciones, se fue realizando pruebas hasta encontrar el valor máximo permitido por el sistema, el cual son 1.600 bits, como es de esperar se pierde mucha resolución.

Como resultados a este experimento se obtiene:

BLEU = 4,08, 29,8/6,1/2,4/1,3

Como volvemos a observar hemos vuelto a perder traducción, pero esto es debido a que el sistema utilizado, se ha experimentado en las longitudes de 1-de-k codificaciones y se vio que los resultados óptimos se encuentran en longitudes de 512 bits, por lo que es nuestro segundo experimento lanzado.

En este nuevo experimento, volvemos a realizar el mismo procedimiento que acabamos de explicar pero se decide reducir el tamaño de los vectores de bitmaps a un tamaño de 529 bits (se intenta aproximar al máximo a 512, debido a las proporciones que genera la imagen).

En este experimento se tiene como resultados:

BLEU = 5,72, 32,5/8,5/3,5/1,8

En este punto obtenemos la primera mejora respecto a nuestro sistema de referencia que se encontraba en 5.55, a pesar que los resultados son mínimos es importante que no se ha perdido traducción a la hora de hacer el cambio por bitmaps.

Finalmente, como último experimento que relaciona este apartado se realizó un experimento que mantenía la parte de creación de bitmaps (tratamiento de datos), pero suprime la parte de segmentación de caracteres, es decir, aplicar la misma idea de bitmaps a palabras, en vez de caracteres.

Como resultados a este experimento tenemos:

BLEU = 8,49, 38,1/11,3/4,9/2,7

Nuevamente volvemos a tener una mejora, y sorprendentemente 3 por ciento por encima del sistema de referencia de palabras, lo cual es una gran mejora para la traducción.

En primer lugar hay que comentar que fue sorprendente que mejorase más con palabras que no con caracteres, ya que los caracteres aportan mucha más información, pero esto tiene explicación y es que en la traducción neuronal es muy importante los tamaños de entrada de las palabras con el de salida, en cuanto más se distancian estos tamaños la traducción empeora, y cuanto más similares son mejora. El hecho de que los resultados fuesen igual de buenos o algo mejor es debido que lo que se perdía en distancia de longitudes de entrada y salida se ganaba con los bitmaps.

Además el hecho de hacer una traducción neuronal por caracteres es necesario primero contextualizar cada carácter en la palabra correspondiente y seguidamente en la frase.

5.2.3. Ejemplos comparativos con atención

Seguidamente volveremos a mostrar los mismos ejemplos que hemos realizado para los experimentos que no tenían atención, de manera que puedan ser comparativos con los que sí que tienen. El ejemplo que mostraremos estará centrado en el caso de éxito, es decir, en el uso de palabras como bitmaps $23x23$, ya que una vez obtenido una mejora en este campo es importante fijarlo como base para futuros experimentos. En la tabla 5.3 se muestran

dichos ejemplos.

	Type	Sentence
1	Src	55/14 . 大会议事规则第
	Words	Convenio sobre la emprendidas tratado
	+Bitmap	Reforma en **la Asamblea General**
	Ref	reglamento de **la Asamblea General**
2	Src	（e）小口径弹药;
	Words	e) el Medio comunicación Mundial ;
	+Bitmap	e) La información **pequeño calibre ;**
	Ref	e) Municiones de **pequeño calibre ;**
3	Src	58/147 . 妇女的家庭暴力行为
	Words	ampliamente de los derechos humanos
	+Bitmap	discriminación **contra la mujer**
	Ref	violencia **contra la mujer** en el hogar
4	Src	（e）法律和司法体制;
	Words	e) acceso a la tecnología y la gestión de la información;
	+Bitmap	e) la **jurídicas** o **judiciales ;**
	Ref	e) las instituciones **jurídicas** y **judiciales ;**

Tabla 5.3: *Frases de ejemplo. Origen (Src), Palabras del sistema de referencia (Words), Fuentes de Bitmap (+Bitmap), Objetivo (Ref)*

Como se puede apreciar en la tabla, los resultados son mucho mejores que en los resultados del sistema que no usan atención. No solo eso, sino que la traducciones son mínimamente mejores.

5.3. Sistema con información de caracteres

Este experimento es algo diferente a los anteriores, ya que es una adaptación del código que hemos utilizado hasta el momento. El cual se basa en la utilización de granularidad de palabras, pero a diferencia de los anteriores recibe información de caracteres de manera adicional.

EL motivo por el que se decide probar esta vía es porque existe una investigación en la que se utiliza este código con caracteres alemanes, en el que se obtuvieron mejores resultados.

Para nuestro experimento la parte que utilizamos en formato bitmap son las palabras, es decir, las palabras que actualmente se forman con vectores de bits aleatorios ahora se forman con vectores de bitmaps, mientras que los caracteres se mantienen en el mismo formato.

En este experimento hemos lanzado dos pruebas con el mismo corpus utilizado hasta el momento:

- La primera ejecución fue el sistema de referencia, el cual trabaja con la granularidad de palabras con información de caracteres.
- La segunda ejecución es el sistema de referencia, pero aplicando la segmentación de palabras con bitmap, y mantenido la granularidad de caracteres del mismo modo.

Los resultados tal y como ya hemos mostrado en la tabla inicial del capítulo.

BLEU = 5,01, 35,5/8,6/3,5/1,8 (sistema de referencia) BLEU = 4,61, 29,4/6,5/2,4/1,2 (sistema de referencia con bitmaps)

Principalmente hay que decir que no se siguió investigando en este punto, ya que los resultados como sistema de referencia era peor que sistemas de referencias con palabras y caracteres. Por esta misma razón no se muestran traducciones como ejemplo ya que no hay ninguna mejora respecto ha experimentos anteriores.

Se debe mencionar que no se seguirá investigando por esta vía, ya que aparentemente no muestra ninguna mejora.

En este punto de la experimentación hemos realizado las posibles alternativas de las que disponemos para ver un buen abanico de experimentos, y así comenzar a ver qué vía es más prometedora.

5.4. Sistema con atención añadida con un corpus más grande

Este experimento (el cual se ha realizado de manera extra de cara a este proyecto), trata de mostrar la magnificación que puede suponer el uso de un corpus u otro. Ya que como todo sistema de inteligencia artificial, a más información tratada mejores son sus resultados. Evidentemente el hecho de tratar más información supone encontrarse obstáculos mayores respecto el tiempo y la memoria.

Si nos centramos un poco más en el experimento, esta prueba es exactamente igual que la que se realiza en el segundo experimento *Sistema con atención añadida*, donde la única diferencia es el cambio de corpus (sus características), es decir, a nivel de código deberemos cambiar los *paths* del sistema con atención añadida, donde ahora apunten al nuevo corpus que utilizamos. Además las longitudes de los diccionarios de origen y destino crecerán ha 90.000. El tamaño real necesario de los diccionarios es mucho más grande, pero la GPU no soporta tamaños superiores.

En este experimento se han lanzado dos ejecuciones:

- La primera ejecución fue el sistema de referencia, el cual trabaja con la granularidad de palabras.
- La segunda ejecución es el sistema de referencia, pero aplicando la segmentación de palabras, tal y como ya se explicó su realización en apartados anteriores.

No es necesario realizar un apartado descriptivo para cada ejecución, ya que tal y como hemos dicho no hay ninguna diferencia entre el segundo experimento y este. La única diferencia es que el tiempo de ejecución del entrenamiento del corpus crece a 17 días por ejecución.

Seguidamente mostraremos las características del corpus utilizado en la tabla 5.4.

L	Set	S	W	V
ES	Train	3.02M	51.7M	184.8K
	Dev	990	43.4K	5.02K
	Test	1K	44.2K	5.5K
ZH Palabras	Train	3.02M	43.9M	373.5K
	Dev	990	33.4K	3.7K
	Test	1K	33.7K	3.8K

Tabla 5.4: *Detalles del Corpus. Número de frases (S),palabras (W), vocabulario (V)*

Respeto a los resultado, volveremos a mostrar la misma tabla que en los experimentos anteriores, es decir, mostraremos el mismo conjunto de frases para poder apreciar las mejoras obtenidas, tal y como muestra la tabla 5.5.

Como hemos observado los resultados son mucho mejores que en el segundo experimento, por lo que podríamos decir que este corpus es bastante mejor para trabajar a pesar del coste de tiempo que supone.

Sería muy interesante lanzar dos pruebas con este corpus con la granularidad de caracteres, ya que aunque nuestros resultados fueron escasamente mejor respecto al sistema de referencia podría magnificar con un corpus más grande y incluso tener mejores resultados que este experimento. Pero la realización de este experimento supondría cerca de un més, ya que la experimentación de caracteres es mucho más lenta que la de palabras.

	Type	Sentence
1	Src	55/14 . 大会议事规则第
	Words	UNK . artículo 1 del reglamento de la asamblea general
	+Bitmap	UNK . enmiendas del artículo 1 de la asamblea general
	Ref	reglamento **de la Asamblea General**
2	Src	（e）小口径弹药;
	Words	e) municiones de las calibre ;
	+Bitmap	e) municiones de pequeño calibre ;
	Ref	e) Municiones de pequeño calibre ;
3	Src	58/147 . 妇女的家庭暴力行为
	Words	UNK . eliminación de la violencia contra la mujer
	+Bitmap	UNK . eliminación de la violencia contra la mujer en la familia
	Ref	violencia **contra** la mujer en el hogar
4	Src	（e）法律和司法体制;
	Words	e) fortalecimiento de las instituciones jurídicas y judiciales ;
	+Bitmap	e) aumentar las instituciones jurídicas y judiciales ;
	Ref	e) las instituciones jurídicas y judiciales ;

Tabla 5.5: *Frases de ejemplo. Origen (Src), sistema de referencia (Words), inclusión de Bitmap (+Bitmap), Objetivo (Ref)*

Capítulo 6

Conclusiones

En este capítulo se hablará de las conclusiones de todo este estudio, donde las catalogaremos en tres secciones diferenciadas:

- En primer lugar hablaremos de que conclusiones técnicas y personales podemos extraer de este proyecto, es decir, a que se deben estos resultado.
- En segundo lugar, hablaremos como se podría continuar este proyecto, el cual podría aumentar los resultados.
- Finalmente hablaremos del lugar donde se ha realizado una publicación de un artículo de investigación sobre este proyecto.

6.1. Conclusiones técnicas y personales

Como conclusión global de este proyecto se ha visto que la utilización de bitmaps es mejor que la aleatoria. Se han visto mejoras en segmentación de caracteres y palabras con representaciones de vectores de bitmap. Lo

cual podríamos decir que cualquier aportación sobre la representación o el tratamiento de datos previo podría ser una mejora a las redes neuronales.

Si concretamos un poco más en porque ha funcionado mejor la granularidad de palabras contra la de caracteres podríamos decir dos cosas:

- Una posibilidad de que la granularidad de palabras funcione mejor podría ser por la longitud de las palabras de origen y las palabras destino, ya que es importante que esta desviación no sea demasiado grande para tener buenas traducciones debido a aproximaciones de longitud que hace internamente el sistema para reconocer a qué palabra corresponde su traducción.

- La otra posibilidad que los caracteres no hayan sido mejores que las palabras es que se trataba de un corpus pequeño, por lo que sería muy interesante realizar el último experimento que realizamos (granularidad de palabras con un corpus mayor), con granularidad de caracteres, de este modo podríamos saber con certeza que es mejor, pero esto queda fuera del alcance del proyecto debido a que el tiempo de ejecución desborda el tiempo del que se dispone para su realización.

6.2. Trabajo futuro

Este proyecto está realizado con un código libre y abierto, tal y como ya hemos dicho. Por lo que es un proyecto que está completamente abierto a ser mejorado, ya que está en un fase totalmente experimental y es muy sencillo conseguir mejoras mediante pequeñas modificaciones.

A todo esto me gustaría ofrecer una propuesta, como proyecto futuro, ya que estoy completamente seguro que puede ser una propuesta interesante. Seguidamente realizaré la propuesta.

Tal y como ya hemos mencionado, en las redes neuronales se juega un gran papel en la información que reciben, que es básicamente en lo que se ha centrado este proyecto.

Hasta el momento, este proyecto ha consistido en generar una matriz más informativa lo cual ha permitido tener mejores resultados, de los conseguidos hasta el momento. Pero esta información (matriz informativa) solo la toma como matriz inicial, después en cada iteración se actualizan los valores, y en la siguiente iteración itera sobre la nueva matriz, perdiendo así los valores iniciales.

Ahí va la propuesta: Dicho esto, una idea para continuar este proyecto sería crear una copia de la matriz inicial, donde cada iteración en vez de leer de la matriz actualizada leyese de la inicial, guardando los valores en la matriz original. De esta manera el sistema siempre trataría sobre la matriz de bitmaps obtenida.

Otra propuesta algo más simple, sería realizar la ejecución con granularidad de caracteres con un corpus más amplio (tal y como ya hicimos en el último experimento). De esta manera podremos saber realmente cual es mejor. Y seguidamente podremos introducir la misma idea que la propuesta realizada en el párrafo anterior.

6.3. Publicación

Para acabar, se debe mencionar que una parte de este proyecto de investigación se ha publicado en Hytra-5, un congreso internacional, el cual realiza su quinto Workshop junto EAMT (European Association for Machine Translation) sobre enfoques híbridos de traducción.

Hytra reúne a investigadores que trabajan en diversos aspectos de la traducción automática híbrida para comparar y contrastar las diversas formas

de integrar paradigmas traducción automática. Donde se experimenta con temas de actualidad, incluidos los enfoques estadísticos que integren información morfológica, sintáctica, semántica y basado en reglas.

El artículo publicado se encuentra en el apéndice del proyecto, redactado totalmente en inglés ya que al tratarse de un artículo internacional es completamente necesario.

Bibliografía

[Abaitua2006] Joseba Abaitua. 2006. Traducción automática: Introducción en 10 horas.

[Banchs et al.2006] Rafael Banchs, Josep Maria Crego, Patrik Lambert, and José B. Mariño. 2006. A feasibility study for chinese-spanish statistical machine translation. In *in Procedings of the 5th International Symposium on Chinese Spoken Language Processing (ISCSLP)*.

[Cho2015a] Kyunghyun Cho. 2015a. Introduction to neural machine translation with gpus (part 1). In *Parallel Forall*.

[Cho2015b] Kyunghyun Cho. 2015b. Introduction to neural machine translation with gpus (part 2). In *Parallel Forall*.

[Cho2015c] Kyunghyun Cho. 2015c. Introduction to neural machine translation with gpus (part 3). In *Parallel Forall*.

[Costa-jussà2015] Marta R. Costa-jussà. 2015. Traducción automática estadística entre chino y castellano. *Komputer Sapiens*, 1:16–36.

[Figueroa et al.2011] Figueroa, David Ferrer, and Isabel Cuadrado Gutiérrez. 2011. Traducción Automática Técnicas Y Aplicaciones."Traducción Automática.

[Pangea2010] Pangea. 2010. Q2 – ¿por mt estadística y no la regla basada en mt? ¿cuáles son las ventajas y desventajas?

[Riaño et al.2015] Riaño, María Paula, and Germán Camilo. 2015. Los Intereses De China En América Latina."Los Intereses De China En América Latinaa.

[San2016] 2016. China: Política Y Economía."Política Y Economía China.

[Wikipedia2016] Wikipedia. 2016. Anexo:Idiomas Por El Total De Hablantes. In *La Enciclopedia Libre*.

[Ángel José Riesgo2016] Ángel José Riesgo. 2016. Apuntes de chino.

Apéndice A

Publicación

Baltic J. Modern Computing, Vol. 0 (2016), No. 0, pp. 00–00

Neural Machine Translation using Bitmap Fonts

David Aldón Mínguez, Marta R. Costa-jussà, José A. R. Fonollosa

Universitat Politècnica de Catalunya, Barcelona

david.aldon@est.fib.upc.edu, {marta.ruiz,jose.fonollosa}@upc.edu

Abstract. Recently, translation systems based on neural networks are starting to compete with systems based on phrases. The systems which are based on neural networks use vectorial representations of words. However, one of the biggest challenges that machine translation still faces, is dealing with large vocabularies and morphologically rich languages. This work aims to adapt a neural machine translation system to translate from Chinese to Spanish, using as input different types of granularity: words, characters, bitmap fonts of Chinese characters or words. The fact of performing the interpretation of every character or word as a bitmap font allows for obtaining more informed vectorial representations. Best results are obtained when using the information of the word bitmap font.

1 Introduction

Deep learning (or neural networks) allows to solve problems that require the processing of big amounts of data[1]. Neural networks try to simulate a common feature of human beings, the accumulated experience. In brief, neural networks are not more than a simplified and artificial model that try to emulate the features of a human brain, which mostly consist of:

1. Processing units that exchange data or information.
2. Use them to recognize patterns, including bitmap fonts, manuscript and time sequences.
3. Have the ability of learning and improving its operation mode.

[1] http://blog.cit.upc.edu/?p=986

Machine translation (MT) can be defined a set of algorithms that aims at transforming a source language to a target language. Since the decade of the 90s, statistical translation systems, among which phrase-based systems, have prevailed over others. These statistical translation systems maximize the probability of target phrases given source phrases.

Recently, systems based on neural networks, which use vector representations of words, have began to have a great relevance [Kalchbrenner and Blunsom2013, Cho et al.2014, Sutskever et al.2014]. The big obstacle that these systems arise is the inability to deal with large vocabularies. This problem originates because of the architecture of these systems, not to mention their computational cost. In this work, we are introducing the bitmap font information of the input unit (either word or character) in order to provide the neural MT system with an informed initialization instead of random [Bahdanau et al.2015].

The rest of the paper is structured as follows. Section 2 reviews the related work, both in the specific task of Chinese-Spanish and in neural MT. Section 3 explains the baseline neural MT system, and our contribution of integrating translation unit (words or characters) bitmap fonts. Section 4 describes the experimental framework and the results obtained. Finally, Section 5 concludes.

2 Related Work

This section does a brief overview of works that approach Chinese-Spanish translation and previous works in neural MT.

2.1 Chinese to Spanish

Surprisingly, there are not many publications in the field of automatic translation between the pair Chinese-Spanish despite being two of the most spoken languages in the world, occupying the first position and the third respectively [Costa-jussà2015].

A work that was done was the creation of a pseudo corpus which is intended to translate English to Chinese or to Spanish and create an artificial corpus for the association of Spanish-Chinese [Banchs et al.2006]. The problem was tried to resolve Chinese-Spanish with Spanish-English and English-Chinese corpora. As reference system, they use a system based on n-grams that differs from the phrases mainly in the translation model.

In 2008, a IWSLT[2] (International Workshop on Spoken Language Translation) evaluation between these two languages was performed. There were two tasks for Spanish-Chinese. The first task was based on a direct translation for Spanish-Chinese. The second task was motivated by the fact that there is little corpus between Spanish-Chinese, but many among the Chinese-English and English-Spanish, so the task proposed consisted in translating from Chinese-Spanish by pivoting through English. As a final result, the second task, the pivot technique, performed better than direct translation because of the larger corpus provided. [Costa-jussà et al.2012] show a comparison between two types of standard pivots (pseudo corpus and cascade) using English and the direct system. These results show that the pivot and direct techniques do not differ much in their results, but that the technical pivot cascade is slightly better than the pseudo corpus.

Differently, [Costa-jussà and Centelles2016] presents the first rule-based MT system for Chinese to Spanish. Authors describe a hybrid method for constructing this system taking advantage of available resources such as parallel corpora that are used to extract dictionaries and lexical and structural transfer rules.

Additionally, to all this research, there is the Chispa Android application and web service[3], that can be useful to tourists or traveling between countries, which use these languages [Centelles et al.2014].

2.2 Neural Machine Translation

Text translation via deep learning relies on an autoencoder structure [Bahdanau et al.2015] to translate from a source to a target language. The autoencoder is trained using translated texts. Source words are mapped to a small space. The new representation of words is encoded in a summary vector (a representation of source sentence) using a recurrent neural network. Then, the summary vector is decoded into the target language. From 2013, there were different groups proposing competitive architectures that have progressed towards this new approach of neural MT [Kalchbrenner and Blunsom2013, Sutskever et al.2014]. And in 2015, [Bahdanau et al.2015] proposed to use gated recursive units (i.e. attention-based mechanism) that allows a better performance on long sentences. This same attention-based mechanism is also used to describe the content of bitmap fonts [Xu et al.2015]. [Lamb and Xie2015] propose a general Convo-

[2] http://iwslt2010.fbk.eu

[3] http://www.chispa.me

lutional Neural Network (CNN) encoder model for MT that fits within in the framework of encoder-decoder models.

3 Theoretical description of the system

In this section, we introduce the neural MT baseline system together with the main technique that we propose to enhance it.

3.1 Baseline System

As a baseline system, see Figure 1, we use the neural network system proposed by [Bahdanau et al.2015] which is mainly based on a encoder-decoder with the attention-based mechanism. For simplification, the Figure does not show the attention-based mechanism.

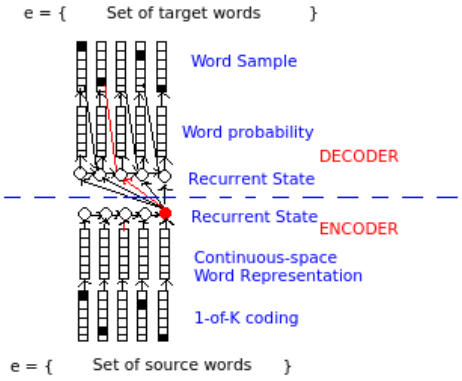


Fig. 1. Encoder/decoder neural MT scheme.

The encoder takes a source sentence, and encodes each word in 1-of-K coding vectors, then trains word embeddings which will be codified into a summary vector through the recurrent network with attention. Then, the decoder applies the reverse process obtaining the destination sentence.

3.2 Adding Word Bitmap fonts

In this study, the previous system will be enhanced to be able to use word bitmap fonts to add further information to the neural system. Given that the baseline representations of vectors are random bits 0/1 level (with the 1-of-K coding), we propose to create a more informative representation. We represent Chinese words by means of 2-dimensional bitmap which reflects the shape of the written word characters. See Figure 2 for illustration.

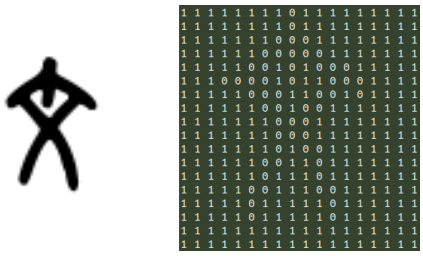


Fig. 2. Chinese Word Representation

Like this, we are converting Chinese words to bitmap fonts. Then we can get the vector of bits representing the bitmap fonts obtained from each word. In this way it is not only providing more information to the system, but it is contributing with much smarter information than a set of random values, due to the characteristic that offer the neural networks learning patterns. This new bitmap font vector becomes the initialization of word embeddings used in the encoder. See the integration of this new encoding in the system in Figure 3.

4 Experiments and Results

In this work, we use the Chinese-Spanish parallel corpus, United Nations Corpus (UN) [Rafalovitch and Dale2009]. Corpus statistics are shown in Table 1. Statistics for Chinese are shown both with word and character segmentations. In the case of word segmentation, the size of the vocabulary is similar to the target vocabulary, while in the case of using Chinese characters, we have a much lower vocabulary.

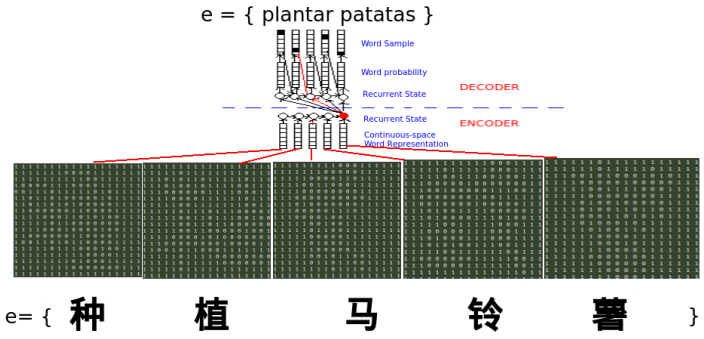


Fig. 3. New Encoder-Decoder

L	Set	S	W	V
ES	Train	58.6K	2.3M	22.5K
	Dev	990	43.4K	5.4k
	Test	1K	44.2K	5.5K
ZH Words	Train	58.6K	1.6M	17.8K
	Dev	990	33K	3.7K
	Test	1K	33.7K	3.8K
ZH Characters	Train	58.6K	2.8M	3.8K
	Dev	990	53.9K	1.7K
	Test	1K	55.1K	1.7K

Table 1. Corpus details. Number of sentences (S),words (W), vocabulary (V). M stands for millions and K stands for thousands.

The neural-based system was built using the software available in github [4]. We generally used settings from previous work: networks have an embedding of 529 and a dimension of 1024. We used a vocabulary size of 20000 in Spanish, 3500 for Chinese when using characters and 15000 for Chinese when using words.

Given that we are experimenting with either Chinese words or characters, we also tried with both word/character initialization. In the case of words, the bitmap fonts had less resolution than characters because the size is the same. Table 2 shows the results in terms of BLEU.

[4] http://github.com/nyu-dl/dl4mt-tutorial/

System	BLEU
Characters	5.52
Characters +Bitmap	5.72
Words	5.55
Words +Bitmap	**8.49**

Table 2. BLEU results. In bold, best results.

Results show that using bitmap fonts as initialization is much better than using a random initialization, since much more information is provided to the neural system. When using character bitmap fonts the improvement is of 0.2 BLEU points, while using word bitmap fonts the improvement is of almost of 3 BLEU points. In any case, it is observed that it is better to use words than characters as translation units.

	Type	Sentence
1	Src	55/14 . 大会议事规则第
	Words	Convenio sobre la emprendidas tratado
	+Bitmap	Reforma en **la Asamblea General**
	Ref	reglamento de **la Asamblea General**
2	Src	（e）小口径弹药;
	Words	e) el Medio comunicación Mundial ;
	+Bitmap	e) La información **pequeño calibre ;**
	Ref	e) Municiones de **pequeño calibre ;**
3	Src	58/147 . 妇女的家庭暴力行为
	Words	ampliamente de los derechos humanos
	+Bitmap	discriminación **contra la mujer**
	Ref	violencia **contra la mujer** en el hogar
4	Src	（e）法律和司法体制;
	Words	e) acceso a la tecnología y la gestión de la información;
	+Bitmap	e) la **jurídicas** o **judiciales ;**
	Ref	e) las instituciones **jurídicas** y **judiciales ;**

Table 3. Example Sentences. Source (Src), Baseline (Words), Bitmap fonts (+Bitmap), Reference (Ref)

Table 3 shows some examples of the kind of improvements that the neural MT system with the new initialization is capable of. Examples show how it improves the adequacy and fluency of the translations in general.

5 Conclusions

This work has presented an alternative to the representation of 1-of-K coding using bitmap fonts instead. We have tried to take advantage of the representation of patterns.

Neural performance in this task is far from state-of-the-art results [Costa-jussà et al.2012]. The fact of not achieving comparable performance to the standard phrase-based system may be due to the fact that we are using a small dataset. However, this study shows a significant improvement when using a smarter initialization of the neural word vectors from standard neural MT system. The bitmap font initialization definitively provides more information to the neural systems than the random 1-of-K vectors. When comparing Chinese words or characters, the performance is similar, but it makes a big difference introducing bitmap fonts of words instead of bitmap fonts of characters. However, experiments on larger corpus would be required and are left for further work.

Software for experiments reported in this paper is freely available in github[5].

Acknowledgements

This work is supported by the 7th Framework Program of the European Commission through the International Outgoing Fellowship Marie Curie Action (IMTraP-2011-29951) and also by the Spanish Ministerio de Economía y Competitividad and the European Regional Development Fund, contract TEC2015-69266-P (MINECO/FEDER, UE).

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *CoRR*.

Rafael Banchs, Josep Maria Crego, Patrik Lambert, and José B. Mariño. 2006. A feasibility study for chinese-spanish statistical machine translation. In *in Procedings of the 5th International Symposium on Chinese Spoken Language Processing (ISCSLP)*.

Jordi Centelles, Marta R. Costa-jussà, and Rafael E. Banchs. 2014. Chispa on the go: A mobile chinese-spanish translation service for travellers in trouble. In *Proceedings of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 33–36.

[5] https://github.com/aldomin/NMTbitMaps

Kyunghyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder–decoder approaches. In *CoRR*.

Marta R. Costa-jussà and Jordi Centelles. 2016. Description of the chinese-to-spanish rule-based machine translation system developed using a hybrid combination of human annotation and statistical techniques. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 15.

Marta R. Costa-jussà, Carlos A. Henríquez Q., and Rafael E. Banchs. 2012. Evaluating indirect strategies for chinese-spanish statistical machine translation. *Journal Of Artificial Intelligence Research*, 45:762–780.

Marta R. Costa-jussà. 2015. Traducción automática estadística entre chino y castellano. *Komputer Sapiens*, 1:16–36.

Nal Kalchbrenner and Phil Blunsom. 2013. Recurrent convolutional neural networks for discourse compositionality. In *CoRR*.

Andrew Lamb and Michael Xie. 2015. Convolutional encoders for neural machine translation. In *CoRR*.

Alexandre Rafalovitch and Robert Dale. 2009. United Nations General Assembly Resolutions: A Six-Language Parallel Corpus. In *Proc. of the MT Summit XII*, pages 292–299, Ottawa.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *CoRR*.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *CoRR*.