Índice general

1. Introducción **1**
1.1. Lengua otomí 1
1.1.1. Origen 1
1.2. Problemática 2
1.3. Objetivo 3
1.4. Hipótesis 4

2. Avances en etiquetadores automáticos **5**
2.1. Marco teórico 5
2.1.1. Natural Language Processing (NLP) 5
2.1.2. Etiquetadores 5
2.1.3. Machine Learning (ML) 5
2.1.4. Modelos gráficos 5
2.1.5. Conditional Random Fields 6
2.2. Estado del arte 7
2.2.1. Trabajos sobre bajos recursos 8

3. Etiquetador morfológico para el otomí (Metodología) **9**
3.1. Corpus: otomí de Toluca 9
3.1.1. Tokens 12
3.1.2. Distribución de etiquetas 12
3.2. Arquitectura 12
3.3. *Pipeline* 14
3.3.1. Obtención y preprocesamiento del corpus 14
3.3.2. Feature functions 15
3.3.3. Hardware utilizado 15

4. Experimentación y Resultados **16**
4.1. Corpus de evaluación 16
4.2. Análisis de resultados 16

5. Conclusiones **17**

Etiquetador automático de la morfología del otomí usando predicción estructurada

Diego Alberto Barriga Martínez

17 de abril de 2020

Capítulo 1

Introducción

1.1. Lengua otomí

En esta sección se menciona los lugares donde se describe el idioma otomí de forma somera, se mencionan algunos lugares donde es hablado el otomí y características fundamentales de la lengua.

1.1.1. Origen

La palabra otomí es de origen náhuatl (singular: *otomitl*, plural: *otomî*). Por otra parte, los otomíes se nombran a sí mismos *ñähñu*[1], que significa "los que hablan otomí".

Los grupos indígenas que hablan el idioma otomí se encuentran en diversas partes del territorio mexicano como: Estado de México, Querétaro, Hidalgo, Puebla y Veracruz (Barrientos López, 2004). El otomí es una lengua indígena una gran variación dialectal que depende de su distribución geográfica.

En el Estado de México el pueblo *ñähñu* está disperso por varios municipios tales como: Toluca, Lerma, Chapa de Mota, Aculco, Amanalco, Atizapán de Zaragoza, por mencionar algunos. En otros municipios como Naucalpan, Ecatepec, Nezahualcóyotl y Tlalnepantla se pueden encontrar hablantes por efectos de la migración. Según Barrientos López (2004) la población total de hablantes otomíes en el Estado de México supera los cien mil, sin embargo,

[1]Existen organizaciones indígenas, como el Consejo de la Nacionalidad Otomí, que escriben la auto-denominación como hñätho hñähñu y también ñätho ñähño. Sin embargo, esta auto denominación puede variar.

datos actuales .

En concreto existen **nueve** variantes del otomí y cabe recalcar que dicha variación puede presentarse incluso dentro del mismo estado. Tan solo el Estado de México presenta tres variantes del otomí: El otomí de Tilapa, hablado en el municipio de Santiago Tianguistenco; el Otomí de Acazulco, del municipio de San Jerónimo Acazulco; y el Otomí de Toluca, de San Andrés Cuexcontitlán.

1.2. Problemática

El NLP es un área de la computación que permite reconocer, procesar e interpretar el lenguaje humano dentro de un sistema computacional. El objetivo de esta área es hacer que las computadoras realicen tareas que involucran el lenguaje humano. Algunas tareas generales consisten en permitir la comunicación humano-máquina, o simplemente hacer un exitoso procesamiento de texto o voz humanos (**?**). Ejemplos de aplicación actuales son los traductores automáticos, asistentes personales que reconocen voz, motores de búsquedas en Internet, análisis de sentimientos en textos, síntesis de voz, etiquetado de textos y muchas más aplicaciones.

Una de las tareas más populares de NLP es el etiquetado automático de textos. Este etiquetado puede realizarse a diferentes niveles lingüísticos, por ejemplo, morfosintáctico (*Part-Of-Speech tagging, POS*), sintáctico (*parsing*), morfológico, etc. El nivel morfológico tiene que ver con la estructura interna de las palabras (**?**); en particular, existe un tipo de etiquetado, de gran importancia para el análisis lingüístico, llamado glosado que asigna etiquetas a las unidades que conforman a una palabra.

Para lograr lo anterior, los enfoques actuales aplican técnicas de ML. El ML es un subcampo de la Inteligencia Artificial (IA), que constituye un enfoque de resolución de problemas caracterizado por estimar una solución a partir de la experiencia (**?**). La experiencia se refiere a datos etiquetados (ejemplos) que permiten inferir un modelo estadístico de aprendizaje. Entre los métodos de ML ampliamente utilizados se pueden mencionar las Support Vector Machines (SVMs), árboles de decisión, o bien los modelos gráficos, como las redes neuronales o los métodos generativos, solo por mencionar algunos. Para las tareas de etiquetado en NLP generalmente se utilizan modelos gráficos supervisados, por ejemplo, modelos ocultos de Markov (*Hidden Markov Models, HMM*).

No obstante, el lenguaje natural es complejo y dinámico, ya que tiene fenómenos que hacen que las tareas de reconocimiento, generación y procesamiento se vuelvan difíciles para las computadoras. Adicionalmente, existen escenarios donde estos métodos no son efectivos como es el caso de las lenguas de bajos recursos, que son lenguas que tienen pocos recursos digitales con los que trabajar. Por ejemplo, si se tienen pocos datos iniciales para el entrenamiento del modelo de aprendizaje las predicciones serán poco precisas o equivocadas. Los bajos recursos son un escenario común en México donde, a pesar de que existe una rica diversidad lingüística, gran parte de las lenguas originarias no poseen contenido web ni publicaciones digitales y por tanto carecen también de tecnologías del lenguaje. El escenario mencionado anteriormente supone un reto para los métodos de aprendizaje convencionales, que requieren de grandes cantidades de datos de entrenamiento para funcionar correctamente. Por lo tanto, es un importante reto de investigación desarrollar aproximaciones que funcionen con lenguas de escasos recursos. En particular, en este trabajo nos enfocamos en el glosado automático del otomí, una lengua con gran riqueza morfológica y con escasez de recursos digitales.

El glosado puede ser un primer paso para el desarrollo de más tecnologías del lenguaje; no solo para el otomí, que presenta un grado de extinción acelerada

Diego: ¿Esta cita donde la puedo encontrar?

(CDI), sino para las 68 agrupaciones lingüísticas que se hablan en México.

1.3. Objetivo

Diseñar e implementar un etiquetador morfológico para el otomí basado en técnicas de Procesamiento del Lenguaje Natural (*Natural Language Processing, NLP*) con Aprendizaje de Máquina (*Machine Learning, ML*). En particular, se hará énfasis en métodos de aprendizaje estructurado débilmente supervisados. Específicamente, se aplicará *Conditional Random Fields (CRF)* para etiquetado morfológico (glosado) del otomí, una lengua de bajos recursos.

1.4. Hipótesis

Se espera obtener un modelo que produzca glosa para el otomí, generada automáticamente, con base en el entrenamiento con pocos ejemplos previamente etiquetados. Al obtener una buena exactitud en la predicción automática de glosa se apoyaría a los anotadores humanos a reducir trabajo repetitivo y exhaustivo. Además, se espera obtener avances de una metodología adaptable a un mayor número de lenguas mexicanas. Sería deseable que esta metodología experimental pueda ser replícale en otras lenguas habladas en México.

Capítulo 2

Avances en etiquetadores automáticos

2.1. Marco teórico

2.1.1. Natural Language Processing (NLP)

2.1.2. Etiquetadores

2.1.3. Machine Learning (ML)

2.1.4. Modelos gráficos

Los límites de los modelos gráficos en para bajos recursos

En este capítulo se explicará qué ventajas tienen los *Conditional Random Fields (CRF)* sobre otros modelos de aprendizaje, se mencionan formalmente los elementos fundamentales que describen los *CRF's*.

En lingüística computacional una tarea de interés es el procesamiento estadístico del lenguaje natural, en particular, el etiquetado y segmentación de secuencias de datos. En ese sentido, es habitual la utilización de **modelos generativos**, cómo los *Hidden Markov Models (HMMs)*, o **modelos condicionales**, como los *Maximum Entropy Markov Models (MEMMs)*.

Por una parte, los modelos generativos intentan modelar una probabilidad conjunta $P(x, y)$ sobre observaciones y etiquetas. Para definir esta probabilidad conjunta se necesita enumerar todas las observaciones posibles. Las limitantes de este enfoque son de diversas índoles como las grandes dimen-

sionalidades en el vector de entrada X, la dificultad de representar múltiples características que interactúan unas con otras y dependencias complejas que hacen la construcción de la distribución de probabilidad un problema intratable con un enfoque computacional.

Por otro lado, una solución a las limitantes de los modelos generativos es un modelo condicional. Estos modelos no son tan estrictos como los primeros al momento de asumir independencias en las observaciones. Los modelos condicionales especifican la probabilidad de posibles etiquetas dada una secuencia de observación.

Consecuencia de lo anterior, no se gasta esfuerzo en modelar las observaciones, dado que en al momento de realizar pruebas estas observaciones son fijas. Segundo, la probabilidad condicional puede depender de características arbitrarias y no dependientes de la secuencia de observación sin forzar al modelo a tomar en cuenta la distribución de estas características, permitiendo que el modelo sea tratable (Lafferty et al., 2001).

Un ejemplo de estas ventajas con los *MEMMs* que son modelos secuenciales de probabilidad condicional. Sin embargo, estos modelos y otros que son no generativos, de estados finitos y que son clasificadores basados en el estado siguiente comparten una debilidad llamada *label bias problem*. Lafferty et al. (2001) define que existe el *label bias problem* cuando "las transiciones que dejan un estado compiten solo entre sí, en lugar de entre todas las demás transiciones en el modelo".

Dado que las transiciones son las probabilidades condicionales de los siguientes posibles estados una observación puede afectar cuál será el estado siguiente sin tomar en cuenta que tan adecuado será este. Por tanto, se tendrá un sesgo en los estados con menos transiciones de salida.

2.1.5. Conditional Random Fields

Como menciona Sutton et al. (2012) modelar las dependencias entre las entrada puede conducir a modelos intratables, pero ignorar estas dependencias puede reducir el rendimiento.

Dado el que problema abordado en este trabajo, dónde se requiere del etiquetado de secuencias y es en contexto de bajos recursos lingüísticos, se hace necesario utilizar un enfoque más conveniente.

Los *Conditional Random Fields (CRFs)* son un framework para la creación de modelos probabilístico utilizado en técnicas de aprendizaje estructurado. Tienen las ventajas de los *MEMMs* y, en principio, solucionan el *label*

bias problem. El framework tiene un solo modelo exponencial para la probabilidad conjunta de todas las secuencias de las etiquetas de salida dada la secuencia de observación. En contraste los *MEMMs* usan modelos exponenciales para cada probabilidad condicional de los estados siguientes dado el estado actual.

Formalmente Lafferty et al. (2001) definen los *CRFs* como a continuacion se enuncia:

Definición 1. Sea $G = (V, E)$ una gráfica tal que $\mathbf{Y} = (\mathbf{Y}_v)_{v \in V}$, entonces esa $\mathbf{Y}$ es indexada por los vertices de G. Entonces $(\mathbf{X}, \mathbf{Y})$ es un conditional random field en caso de que las variables aleatorias $\mathbf{Y}$ se condicionen por $\mathbf{X}$, la variable aleatoria $\mathbf{Y}_v$ cumple la *propiedad de Markov* con respecto a la gráfica: $p(\mathbf{Y}_v|\mathbf{X}, \mathbf{Y}_w, w \neq v) = p(\mathbf{Y}_v|\mathbf{X}, \mathbf{Y}_w, w \sim v)$, dónde $w \sim v$ significa que w y v son vecinos en G.

En esta tesis, para el modelado de secuencias, se utiliza la forma más sencilla de la gráfica G dónde es una cadena simple o línea. Esto quiere decir que $G = (V = \{1, 2, ...m\}, E = \{(i, i+1)\})$. A este tipo de *CRFs* se les conoce como *linear-chain CRFs*. Como menciona Lafferty et al. (2001) "si la gráfica $G = (V, E)$ de $\mathbf{Y}$ es un árbol (del cual una cadena es el ejemplo más sencillo), los *cliques* son los límites y vertices. Entonces, por el teorema de los *random fields* (Hammersley y Clifford, 1971), la distribución conjunta sobre las etiquetas de secuencias $\mathbf{Y}$ y $\mathbf{X}$ tiene la forma:

$$\mathrm{p}_\theta(y|x) \propto \exp\left(\sum_{e \in \mathbf{E}, k} \lambda_k f_k(e, \mathbf{y}|_e, \mathbf{x}) + \sum_{v \in \mathbf{V}, k} \mu_k g_k(v, \mathbf{y}|_v, \mathbf{x})\right) \tag{2.1}$$

De la ecuación se destacan f_k y g_k que representan las *feature functions*. Estas están definidas y son fijas. Las *feature functions* de está tesis serán descritas más adelante.

2.2. Estado del arte

Los *CRFs* han sido utilizados para la clasificación de regiones en una imagen, estimar el puntaje en un juego de Go, segmentar genes en una hebra

de ADN y análisis sintáctico de lenguaje natural en un texto por mencionar algunas (Sutton et al., 2012).

2.2.1. Trabajos sobre bajos recursos

Con base en el trabajo previo de (Moeller y Hulden, 2018) donde se utilizan técnicas de NLP y ML para tratar para el idioma Lezgi se plantea como hipótesis que dado el tamaño del corpus y la glosa que contiene

se obtendrá texto correctamente glosado con una precisión de al menos 80 %.

Capítulo 3

Etiquetador morfológico para el otomí (Metodología)

En este capitulo se describe el corpus utilizado en este trabajo, se mostrará la arquitectura propuesta para la generación automática de glosa para el idioma otomí. Adicionalmente, se explicará el diseño e implementación del *pipeline* que incluye, entre otras cosas, la determinación de las *feature functions*.

Los *CRFs* muestran claras ventajas sobre otros métodos de aprendizaje basados en gráficas. Su habilidad de tomar las virtudes de los modelos generativos y de los modelos condicionales presentan a este *framework* como una opción para el contexto de los bajos recursos digitales que nos impone, como se describirá más adelante, el tamaño del corpus. En ese sentido, los *CRFs* se utilizaron para predecir secuencias de etiquetas (aprendizaje estructurado) que describen las unidades morfológicas dentro de una palabra, de una variante en particular, del otomí.

3.1. Corpus: otomí de Toluca

La clasificación lingüística introduce al otomí dentro de las lenguas otomianas, las cuales a su vez pertenecen a la rama otopame de la familia otomangue (Barrientos López, 2004). Cada variante muestra particularidades fonológicas, morfológicas, sintácticas y léxicas. En el tratamiento de textos por medio de técnicas de *NLP* se requiere que estos estén normalizados y homogéneos. Lo anterior propicia la obtención del mejor desempeño posible

Categoría	**Cuenta**
Tokens (POS)	8578
Tipos (POS)	44
Tokens (Glosa)	14477
Tipos (Glosa)	112
Oraciones etiquetadas	1786

Tabla 3.1: Tamaño del corpus

Textos	**Número**
Narrativos	32
Dialogados	4
Total de textos	36

Tabla 3.2: Textos del corpus

en los diversos métodos de aprendizaje automático.

Se utilizó un corpus en otomí que, además, cumple la característica de estar glosado. Se trabaja con la variante del otomí de Toluca de la región de San Andrés Cuexcontitlan.

Esta tesis recogió un corpus basado en el trabajo de Lastra y de Suárez (1992) titulado *El otomí de Toluca* y que a su vez fue etiquetado y glosado manualmente por el lingüista Víctor Germán Mijangos de la Cruz[1]. Este corpus es un subconjunto del corpus paralelo español-otomi que se encuentra en la plataforma web Tsunkua[2] TODO: ¿Como se cita?. El subconjunto del corpus utilizado en la sección experimental está descrito en la tabla 3.1, donde se encuentran las etiquetas POS del corpus, y en la tabla 3.2, donde se puede ver la glosa que está presente en el corpus.

Los textos que componen el corpus fueron construidos a partir de las aportaciones de diez hablantes distintos de entre diez y setenta y tres años, de los cuales, siete son de sexo femenino y tres masculino (Lastra y de Suárez, 1992).

Dentro de las etiquetas POS encontramos etiquetas en si mismas como buena.vista, nada.más, zapata que no están incluidas en la tabla pero que se presentan en el copus ya que ...

Diego

Duda aqui

[1]TODO: Liga del repo

[2]https://tsunkua.elotl.mx/

v	obl	det	cnj	dem
unkwn	n	neg	p.loc	prt
conj.adv	dim	gen	cond	it
lim	aff	loc	dec	conj
cord	san	cnj.adv	regular/v	adv
adj	d	d	d	d

Tabla 3.3: POS

stem	det	3.cpl	psd	lim
prag	3.icp	lig	det.pl	1.icp
3.pot	ctrf	1.pot	pl.exc	1.cpl
dem	1.pss	dim	pl	1.obj
ila	2.icp	1.prf	3.cnt	3.obj
loc	mod	1.cnt	3.pls	prt
it	dual.exc	3.prf	3.icp.irr	3.pss
2.pss	1.enf	med	dual	p.loc
2.cnt	2	3.imp	int	neg
1.icp.irr	1.cpl.irr	2.obj	aum	1.pls
2.cpl	2.prf	gen	com	2.pot
adj	cond	3.cpl.irr	1.sg	encl
3.sg	3.pss.pl	spt	1.irr	2.enf
conj.adv	caus	con	chico	eh
comp	prf	dist mov	3.irr	det.dem
dcl	nom	2.icp.irr		

Tabla 3.4: Glosa

Vic: Explicar qué significa cada etiqueta

Etiqueta	Tokens
v	2596
obl	2447
det	975
cnj	837
dem	543
unkwn	419
n	273
neg	178
p.loc	81
prt	49

Tabla 3.5: POS (Primeros 10)

3.1.1. Tokens

3.1.2. Distribución de etiquetas

3.2. Arquitectura

Para esta tesis proponemos una arquitectura de aprendizaje estructurado débilmente supervisado utilizando el método gráfico *CRF* que permitirá la predicción de secuencias que describen las unidades morfológicas (glosa) dentro de una palabra en otomí. Ya que el resultado esperado es la generación de etiquetas que, en principio, dependen unas de otras, un método basado en gráficas como los *CRF* es el adecuado.

Los *CRFs* predicen las secuencias de glosa, que será la salida Y dadas las observaciones X (texto previamente glosado). Puntualmente, se utiliza el modelo **1st-order Markov CRF with dyad features**. Adicionalmente, es utilizado el algoritmo de aprendizaje de **Limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS)** como se mencionó en la sección 2.1.

El objetivo de esta arquitectura es realizar un correcto etiquetado automático de glosa para la lengua otomí, en particular la variante del otomí de Toluca, utilizando técnicas de aprendizaje estructurado débilmente supervisado. Se definieron de aprender un conjunto de *feature functions* que describen TODO el contexto y brindan información útil para la fase de entrenamiento.

Ximena

Tener bien claro cuál es nuestra hipótesis, separarnos un poco del trabajo de lezgi

Etiqueta	**Tokens**
stem	7527
det	733
3.cpl	450
psd	418
lim	374
prag	362
3.icp	346
lig	289
det.pl	271
1.icp	270

Tabla 3.6: Glosa (Primeros 10)

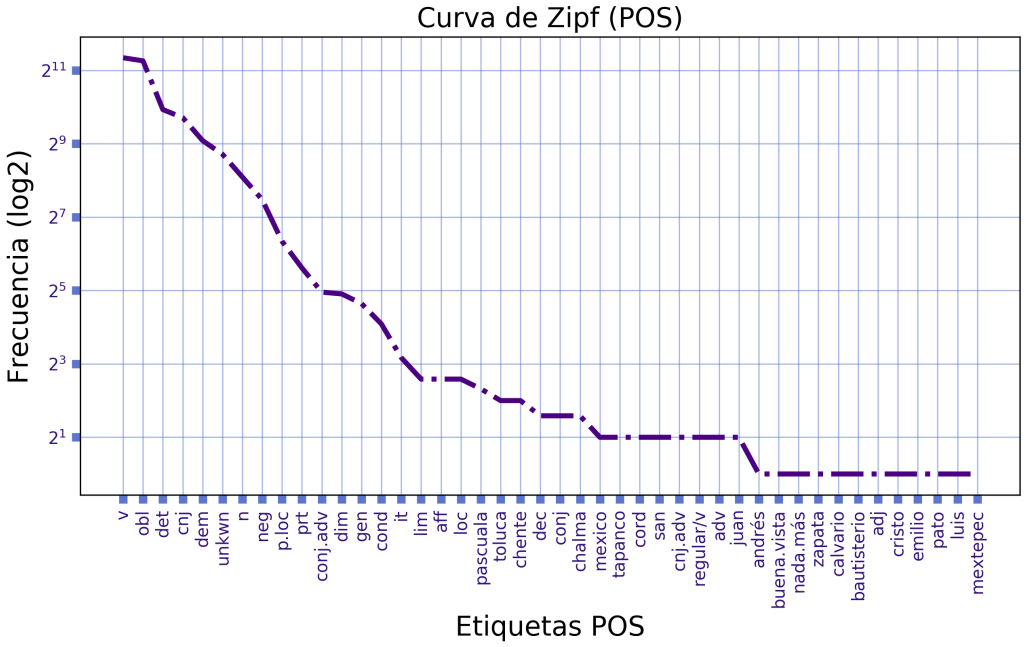

Figura 3.1: Distribución de etiquetas POS

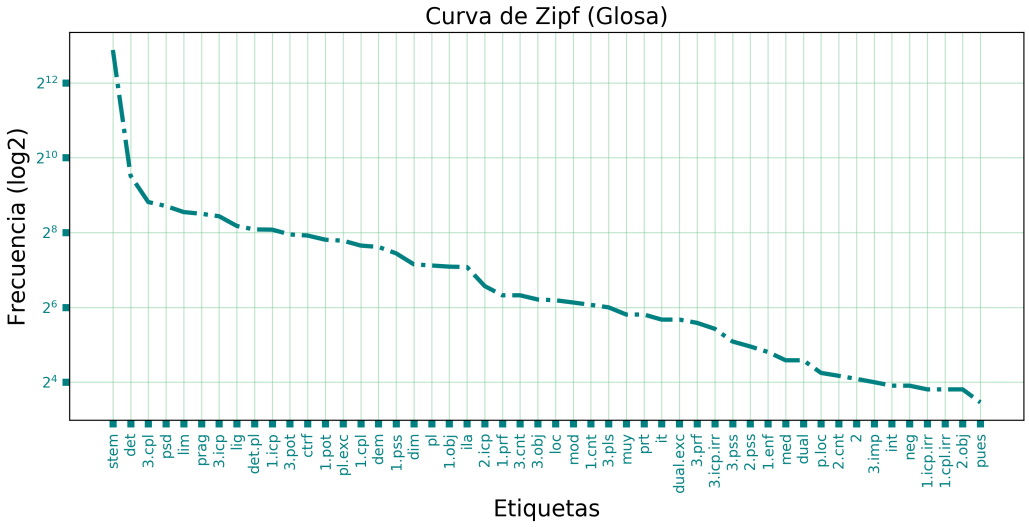

Figura 3.2: Distribución de glosa (primeras 50)

El *pipeline* de aprendizaje semi-supervisado para la generación de glosa para

el otomí de Toluca se describe en las secciones siguientes.

3.3. *Pipeline*

3.3.1. Obtención y preprocesamiento del corpus

Se obtiene el corpus de un archivo de texto plano. Cada renglón de este archivo es una oración en otomí con glosa. Además, se tiene una etiqueta *POS* por cada palabra. Las frases están estructuradas en forma de listas que contienen otras listas validas para el lenguaje `Python`. La glosa esta presente por cada fragmento de las posibles palabras en otomí. Ejemplificando, la frase **"piximähtratá gí"** se representa en el corpus como se muestra en el ejemplo 3.3.1;

Ejemplo 3.3.1. `[[["pi",ït"], ["xi","3.prf"], ["mähtratá","stem"], ["gí","1.obj"], "v"]]`

Entonces, la estructura de las listas, por renglón, tiene la estructura `[[[letras, glosa], [letras, glosa], ..., POS],...]`.

- Obtener el corpus en otomí previamente glosado y obtenerlo en un formato que especifique la información de las oraciones a nivel de letra especificando su Bio Label.
- Los CRF toman como entrada los datos X que corresponden al corpus en otomí introducido en las feature functions asociados de forma biyectiva con la etiqueta Bio Label que le corresponde. Con base en esto se entrenará un modelo que busque maximizar el logaritmo de verosimilitud con el método de aprendizaje L-BFGS
- Posterior se obtendrá un modelo entrenado con el que se generarán etiquetas de glosa para el otomí. Por lo tanto, el modelo recibirá párrafos de texto en otomí y retornará el texto glosado.
- Se considera exitosa la predicción si se logra maximizar la correcta clasificación de las secuencias de salida. Para determinar si la predicción fue exitosa se utilizaron técnicas típicas de ML como K-folds que consiste en tomar K fragmentos de los datos de entrada para utilizarlos para probar el modelo y así obtener una precisión, recall y F-score.

3.3.2. Feature functions

3.3.3. Hardware utilizado

En la fase de experimentación fue utilizado el paquete python-crfsuite que es un *binding* de la implementación de TODO:CITA para CRFs para lenguaje de programación *Python* en su versión *3.7*. La fase de experimentación corrió en una maquina con un procesador Intel i7-7700HQ @ 3.800GHz con 16 GB de memoria principal. En promedio una corrida de entrenamiento y evaluación *K-folds* con *k=10* tomó 52 minutos.

Capítulo 4

Experimentación y Resultados

4.1. Corpus de evaluación

Aquí se habla del K fold y de como se introdujo el corpus retador.

4.2. Análisis de resultados

Capítulo 5

Conclusiones

Bibliografía

Guadalupe Barrientos López. 2004. *Otomíes del Estado de México*.

John M Hammersley y Peter Clifford. 1971. Markov fields on finite graphs and lattices. *Unpublished manuscript* 46.

John Lafferty, Andrew McCallum, y Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data .

Yolanda Lastra y Yolanda Lastra de Suárez. 1992. *El otomí de Toluca*. Instituto de Investigaciones Antropológicas, UNAM.

Sarah Moeller y Mans Hulden. 2018. Automatic glossing in a low-resource setting for language documentation. In *Proceedings of the Workshop on Computational Modeling of Polysynthetic Languages*. pages 84–93.

Anil Kumar Singh y Samar Husain. 2005. Comparison, selection and use of sentence alignment algorithms for new language pairs. In *Proceedings of the ACL Workshop on Building and using Parallel texts*. Association for Computational Linguistics, pages 99–106.

Charles Sutton, Andrew McCallum, et al. 2012. An introduction to conditional random fields. *Foundations and Trends® in Machine Learning* 4(4):267–373.