

Proyecto final de Procesamiento de Lenguaje Natural 2017

Entrenamiento y validación de *Fasttext* para el Castellano

Juan M. Scavuzzo

Septiembre de 2017

- Librería de Facebook AI Research

- Librería de Facebook AI Research
- Open Source

- Librería de Facebook AI Research
- Open Source
- Diseñada para representación y clasificación de texto

- Librería de Facebook AI Research
- Open Source
- Diseñada para representación y clasificación de texto
 - Word Embeddings (skipgram, cbow)

- Librería de Facebook AI Research
- Open Source
- Diseñada para representación y clasificación de texto
 - Word Embeddings (skipgram, cbow)
 - Clasificación supervisada

- Librería de Facebook AI Research
- Open Source
- Diseñada para representación y clasificación de texto
 - Word Embeddings (skipgram, cbow)
 - Clasificación supervisada
- Ya hay entrenados modelos de embeddings para muchos idiomas

- Evaluar el performance de un modelo entrenado con un GRAN corpus

Problema

- Evaluar el performance de un modelo entrenado con un GRAN corpus
- Param tuning básico

Problema

- Evaluar el performance de un modelo entrenado con un GRAN corpus
- Param tuning básico
- Generar embeddings orientados a sintaxis

Modelo: Entrenamiento - Corpus

Spanish Billion Word Corpus (10GB)

- Se usó el corpus que recopiló Cristian

Modelo: Entrenamiento - Corpus

Spanish Billion Word Corpus (10GB)

- Se usó el corpus que recopiló Cristian
- Se dejó de lado: php, ubuntu, openoffice3 y kde

Modelo: Entrenamiento - Preprocesamiento

- Se quitaron todos los simbolos no alfanuméricos

Modelo: Entrenamiento - Preprocesamiento

- Se quitaron todos los simbolos no alfanuméricos
- Se generaron tokens de:

Modelo: Entrenamiento - Preprocesamiento

- Se quitaron todos los simbolos no alfanuméricos
- Se generaron tokens de:
 - Números correspondientes a fechas

Modelo: Entrenamiento - Preprocesamiento

- Se quitaron todos los simbolos no alfanuméricos
- Se generaron tokens de:
 - Números correspondientes a fechas
 - Flotantes

Modelo: Entrenamiento - Preprocesamiento

- Se quitaron todos los simbolos no alfanuméricos
- Se generaron tokens de:
 - Números correspondientes a fechas
 - Flotantes
 - Enteros

Modelo: Entrenamiento - Preprocesamiento

- Se quitaron todos los simbolos no alfanuméricos
- Se generaron tokens de:
 - Números correspondientes a fechas
 - Flotantes
 - Enteros
 - Abreviaciones (muy mejorable)

Modelo: Entrenamiento - Preprocesamiento

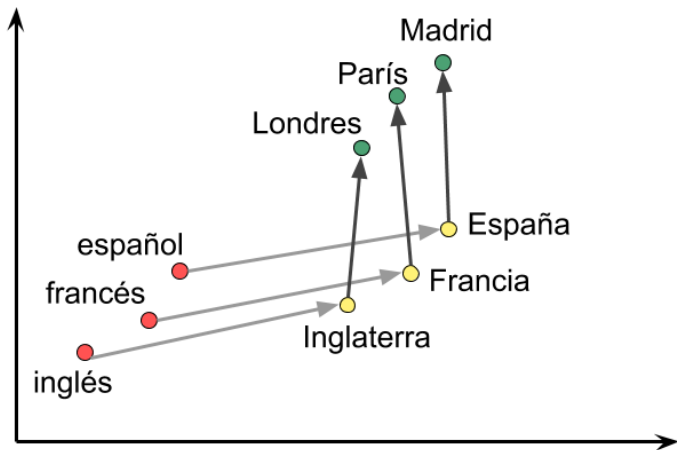
- Se quitaron todos los simbolos no alfanuméricos
- Se generaron tokens de:
 - Números correspondientes a fechas
 - Flotantes
 - Enteros
 - Abreviaciones (muy mejorable)
- No se tocaron las stop-words

Modelo: Entrenamiento - Param Tunning

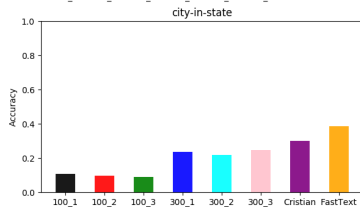
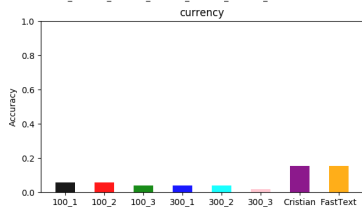
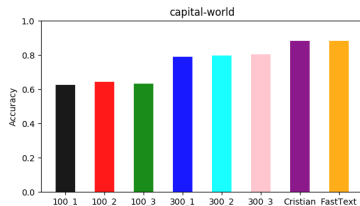
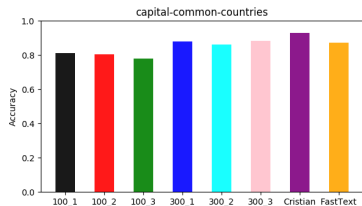
- Dimensión de los vectores (100 - 300)
- Tamaño del n-gram (1 - 3)

A mano

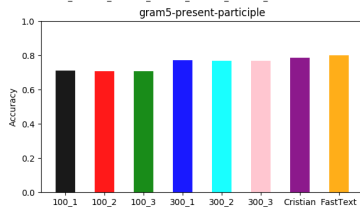
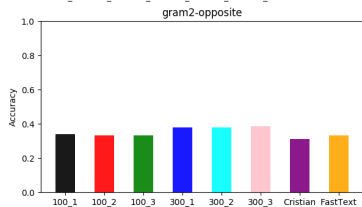
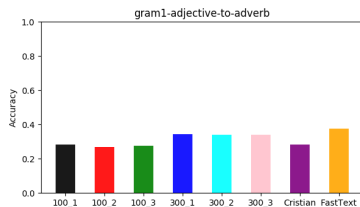
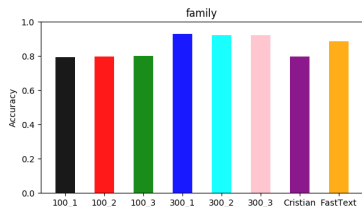
Modelo: Validación - Word2Vec



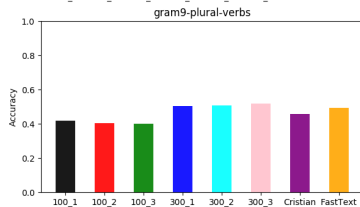
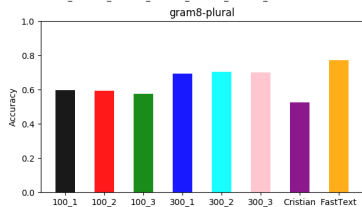
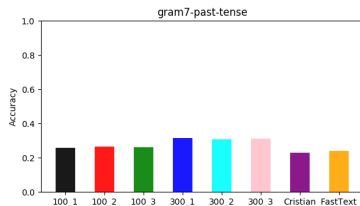
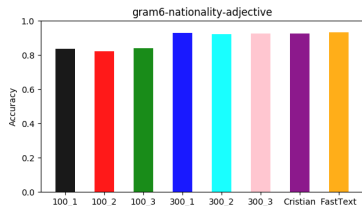
Modelo: Validación - Word2Vec accuracy



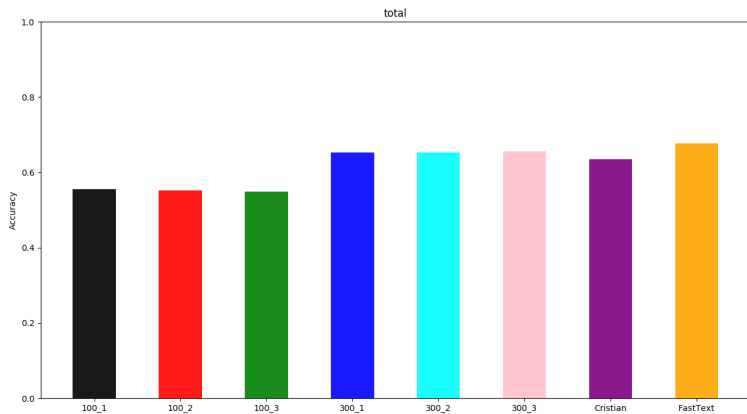
Modelo: Validación - Word2Vec accuracy



Modelo: Validación - Word2Vec accuracy



Modelo: Validación - Word2Vec accuracy



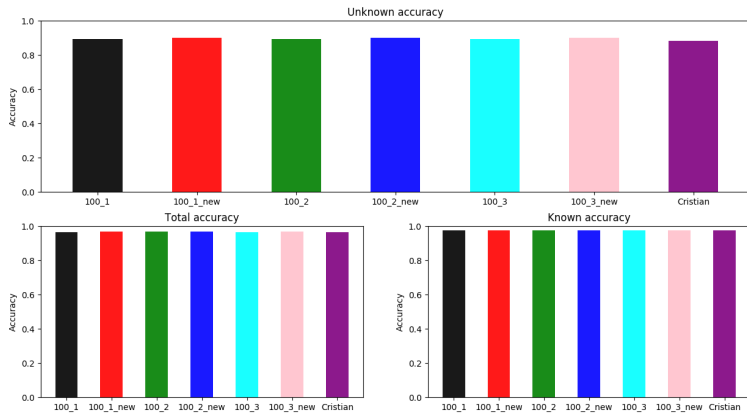
Modelo: Validación - Tagger

- Tagger implementado en práctico

Modelo: Validación - Tagger

- Tagger implementado en práctico
- Agregamos la feature de palabras out-of-vocabulary

Modelo: Validación - Tagger



Fin

Gracias!
Preguntas?

Referencias

- <https://github.com/juansca/WordVectors>
- <https://github.com/facebookresearch/fastText>
- <http://crscardellino.me/SBWCE/>
- <https://radimrehurek.com/gensim/models/word2vec.html>
- <https://www.quora.com/Is-it-compulsory-to-remove-stop-words-with-word2vec>
- <https://stackoverflow.com/questions/34721984/stopword-removing-when-using-the-word2vec>