

Creación eficiente de modelos estadísticos para detección automática y precisa de entidades nombradas

Horacio Miguel Gómez (L : 50825)

Juan Pablo Orsay (L : 49373)

Proyecto final de carrera

2019-11-20

Índice

Abstract	2
1 Introducción	2
2 Estado del arte	2
2.1 El «naive approach»	2
2.2 Redes neuronales y modelos estadísticos	2
2.3 La formula para «deep-learning»	2
2.4 statistical entity recognition model	3
2.5 Word vectors	3
3 Definición del problema	4
4 NERd (Implementación)	4
4.1 Vista lógica	5
4.2 Vista de proceso	17
4.3 Vista de desarrollo	17
4.4 Vista física	17
4.5 Escenarios	18
5 Resultados	18
6 Discusión	18
6.1 Tipos de entidades relevantes	18
6.2 Seed en los types	18
6.3 Mejora live vs offline	18
6.4 Utilidad de la herramienta	19
7 Conclusiones	19
7.1 Examples	19

Abstract

TODO: escribir abstract

1. Introducción

2. Estado del arte

2.1. El «naive approach»

tagueo con expresiones regulares...

2.2. Redes neuronales y modelos estadísticos

Redes neuronales convolucionales https://es.wikipedia.org/wiki/Redes_neuronales_convolucionales

2.3. La formula para «deep-learning»

articulo spacy

2.3.1. embed



Figura 1: TODO: embed

2.3.2. encode

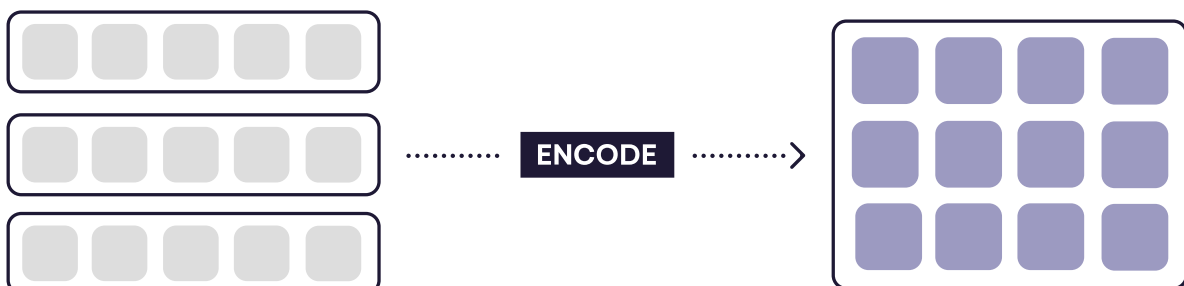


Figura 2: TODO: encode

2.3.3. attend

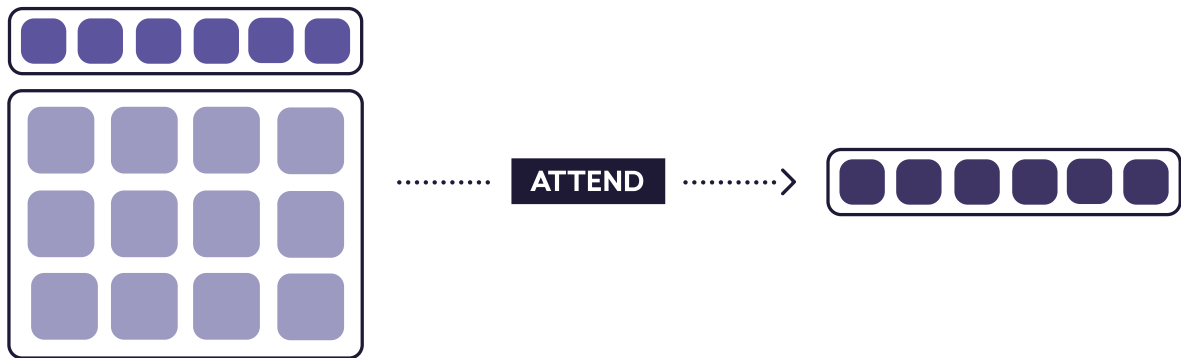


Figura 3: TODO: attend

2.3.4. predict



Figura 4: TODO: predict

Here is a review of existing methods.

2.4. statistical entity recognition model

2.5. Word vectors

$$\vec{king} - \vec{man} + \vec{woman} \approx \vec{queen}$$

(Ethayarajh, Duvenaud, & Hirst, 2019)

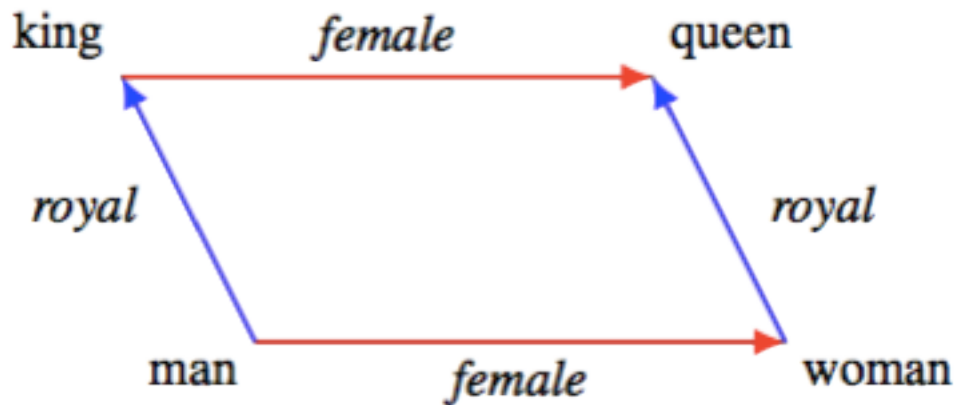


Figura 5: Parallelogram structure in the vector space (by definition)

<https://www.youtube.com/watch?v=sqDHBH9IjRU> SPACY'S ENTITY RECOGNITION MODEL: incremental parsing with Bloom embeddings & residual CNNs

https://github.com/explosion/talks/blob/master/2017-11-02_Practical-and-Effective-Neural-NER.pdf https://github.com/explosion/talks/blob/master/2018-04-12_Embed-Encode-Attend-Predict.pdf

3. Definición del problema

El bottleneck en AI es la data, no los algoritmos. quote de: https://github.com/explosion/talks/blob/master/2016-11-28_The-State-of-AI-2016.pdf

https://github.com/explosion/talks/blob/master/2018-04-12_Embed-Encode-Attend-Predict.pdf

4. NERd (Implementación)

Definido el problema, queda claro que la creación de un modelo entrenado es de vital importancia para cualquier problema de tagueo de entidades. Es por ello que en el presente proyecto final hemos creado una herramienta para el entrenamiento eficiente de modelos estadísticos así como también una interfaz y API para poder consultar entidades. El nombre de esta herramienta es **NERd**, sigla cuyo significado en inglés es **Named Entity Recognition Duh**¹!

Para organizar este capítulo vamos a realizar una descripción basada en el modelo de vistas de arquitectura 4+1.

¹Expresión de obviedad. *Used to express your belief that what was said was extremely obvious* ("Duh definition," 2019)

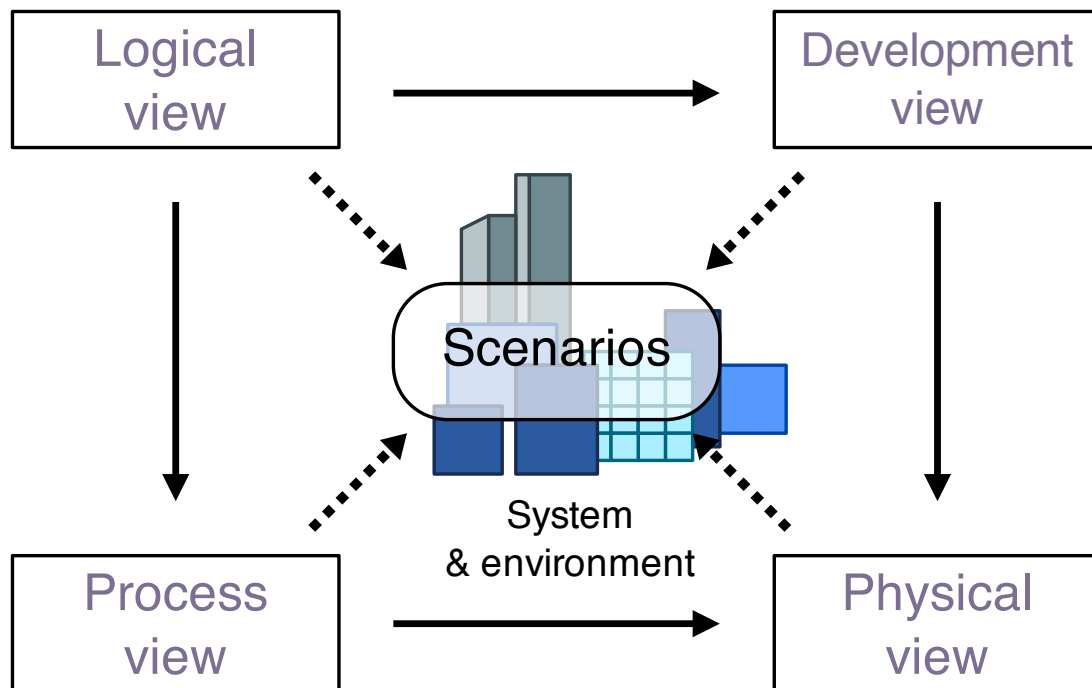


Figura 6: Ilustración de arquitectura 4+1

Este modelo nos permite describir la aplicación de una manera genérica y ordenada.

The «4+ 1» view model is rather «generic»; other notations and tools can be used, other design methods can be used, especially for the logical and process decompositions, but we have indicated the ones we have used with success.

— (Kruchten, 1995)

4.1. Vista lógica

La vista lógica se refiere a la funcionalidad que el sistema proporciona a los usuarios finales.

A continuación detallamos distintas partes del servicio NERd así como también de la interfaz de entrenamiento.

4.1.1. Servicio

El acceso al servicio se realiza mediante un API REST que se auto-documenta debido a implementar la especificación de OpenAPI. A continuación detallamos los endpoints.

4.1.1.1. Autenticación

Rutas del API dedicadas a la autenticación de usuarios.

- **POST** /api/auth/register

- Registrar un usuario nuevo.
- **POST** */api/auth/token*
 - Genera un nuevo token de acceso y refresco con credenciales.
 - Utilizado para la funcionalidad de *login*
- **POST** */api/auth/refresh*
 - Refresca el token de acceso.
 - Utilizado cuando un token de acceso caducó.

4.1.1.2. Usuarios

Conjunto de operaciones relacionadas con los usuarios del sistema.

- **GET** */api/users*
 - Lista de usuarios existentes
 - Separa los resultados en páginas
- **GET** */api/users/top5*
 - Lista de los 5 usuarios con más entrenamientos
- **GET** */api/users/me*
 - Retorna la información del usuario logueado
- **PATCH** */api/users/me*
 - Actualiza la información del usuario logueado
- **GET** */api/users/me/trainings*
 - Retorna los entrenamientos del usuario logueado
- **GET** */api/users/{user_id}*
 - Retorna la información del usuario especificado por *user_id*
- **PATCH** */api/users/{user_id}*
 - Actualiza la información del usuario especificado por *user_id*
- **DELETE** */api/users/{user_id}*
 - Borra al usuario especificado por *user_id*
- **GET** *_api/users/{user_id}/trainings*
 - Retorna los entrenamientos del usuario especificado

4.1.1.3. Roles

- **GET** */api/roles*
 - Retorna la lista de todos los roles asignables a usuarios del sistema

4.1.1.4. Corpus

Rutas dedicadas a operaciones con el corpus del sistema.

- **GET** */api/corpus/{text_id}*
 - Retorna los detalles del texto especificado por *text_id*
- **DELETE** */api/corpus/{text_id}*
 - Borra un texto especificado por *text_id* del corpus
- **GET** */api/corpus/{text_id}/trainings*
 - Retorna la lista de entrenamientos proporcionados por los usuarios sobre las entidades en el texto
- **PUT** */api/corpus/{text_id}/trainings*

- Agrega un entrenamiento para el texto con id *text_id*
- **POST** */api/corpus/upload*
 - Permite agregar textos de manera masiva al sistema
 - Acepta una lista de archivos .txt donde cada línea es un texto a agregar
 - Los archivos deben ser UTF-8
- **GET** */api/corpus*
 - Lista de textos cargados en el sistema para entrenamiento
- **POST** */api/corpus*
 - Agrega un texto al sistema para entrenamiento

4.1.1.5. Snapshots

Conjunto de operaciones relacionadas con los snapshots y workers.

- **GET** */api/snapshots*
 - Listado de los snapshots disponibles
- **GET** */api/snapshots/{snapshot_id}*
 - Retorna información (tipos de entidades, fecha de creación, fecha de entrenamiento, etc.) sobre un snapshot específico
- **DELETE** */api/snapshots/{snapshot_id}*
 - Borra un snapshot con el id especificado
- **POST** */api/snapshots/{snapshot_id}/force-train*
 - Envía la tarea de entrenamiento a los workers que tienen el snapshot *snapshot_id* cargado.
- **POST** */api/snapshots/{snapshot_id}/force-untrain*
 - Envía la tarea de desentrenar a los workers que tienen el snapshot *snapshot_id* cargado.
- **GET** */api/snapshots/current*
 - Retorna información sobre el snapshot actual
- **PUT** */api/snapshots/current*
 - Crea un nuevo snapshot con la información provista

4.1.1.6. Reconocimiento de Entidades Nombradas

Conjunto de operaciones relacionadas al *Reconocimiento de Entidades Nombradas*

- **GET** */api/ner/train*
 - Retorna un texto para que un usuario del sistema revise si está correctamente inferido
 - Únicamente retorna textos que el usuario logueado no haya corregido ya
- **GET** */api/ner/compare/{first_snapshot}/{second_snapshot}*
 - Compara el *Reconocimiento de Entidades Nombradas* entre dos snapshots distintos
- **POST** */api/ner/current/parse*
 - Retorna un documento Spacy para un texto dado utilizando el snapshot actual
- **POST** */api/ner/{snapshot_id}/parse*
 - Retorna un documento Spacy para un texto dado utilizando el snapshot especificado
- **POST** */api/ner/current/entities*
 - Retorna la lista de *Entidades Nombradas* para un texto dado utilizando el modelo actual
- **POST** */api/ner/{snapshot_id}/entities*
 - Retorna la lista de *Entidades Nombradas* para un texto dado utilizando el modelo especificado

4.1.1.7. Entrenamientos

- **DELETE** `/api/trainings/{training_id}`
 - Borra un entrenamiento

4.1.1.8. Workers

- **GET** `/api/workers/`
 - Lista de los workers disponibles
- **POST** `/api/workers/reassign`
 - Reasigna un trabajador de un a versión de snapshot a otra

4.1.2. Web

La página web de *NERd* está enfocada en las tareas de mantenimiento de los servicios ofrecidos por el *API* así como también ofrece de interfaces que permiten a usuarios del sistema corregir de manera eficiente el modelo de inferencia.

4.1.2.1. Inicio

Pantalla de inicio donde se encuentran accesos rápidos para entrenar el modelo o para poder buscar entidades en textos. También se encuentra aquí una lista de los 5 usuarios que más contribuyeron a entrenar el modelo. Detrás de esta funcionalidad se busca generar un espíritu competitivo entre los usuarios para que los mismos busquen contribuir más.

The screenshot shows the NERd web application interface. At the top is a dark blue navigation bar with the logo 'NERd' on the left and several menu items: 'PERFIL', 'USUARIOS', 'CORPUS', 'COMPARAR', 'ESTADO', 'ESPAÑOL' (with a dropdown arrow), and 'SALIR'. Below the navigation bar, the main content area is divided into three sections. The top left section, titled '¡Ayúdame a ser más inteligente!', contains the text 'Te mostraré un texto con anotaciones y podrás decidir si es correcto o no.' and a blue button labeled 'ENTRENAR'. The top right section, titled 'Revisa tu texto', contains the text '¿Querés ver que encuentro? ¡Vení por acá entonces!' and a blue button labeled 'BUSCAR ENTIDADES'. The bottom center section, titled 'Salón de la fama', contains a table listing the top 5 users by training count.

Nombre	Entrenados
Horacio Miguel Gomez	4001
Maria Orsay	200
Ariel Aizemberg	187
Agustin Giussani	107
jorge galloni	45

Figura 7: Pantalla de inicio con usuario logueado

Si la persona no cuenta con permisos de entrenador, se le sugiere que contacte a un administrador para que le otorgue el permiso.

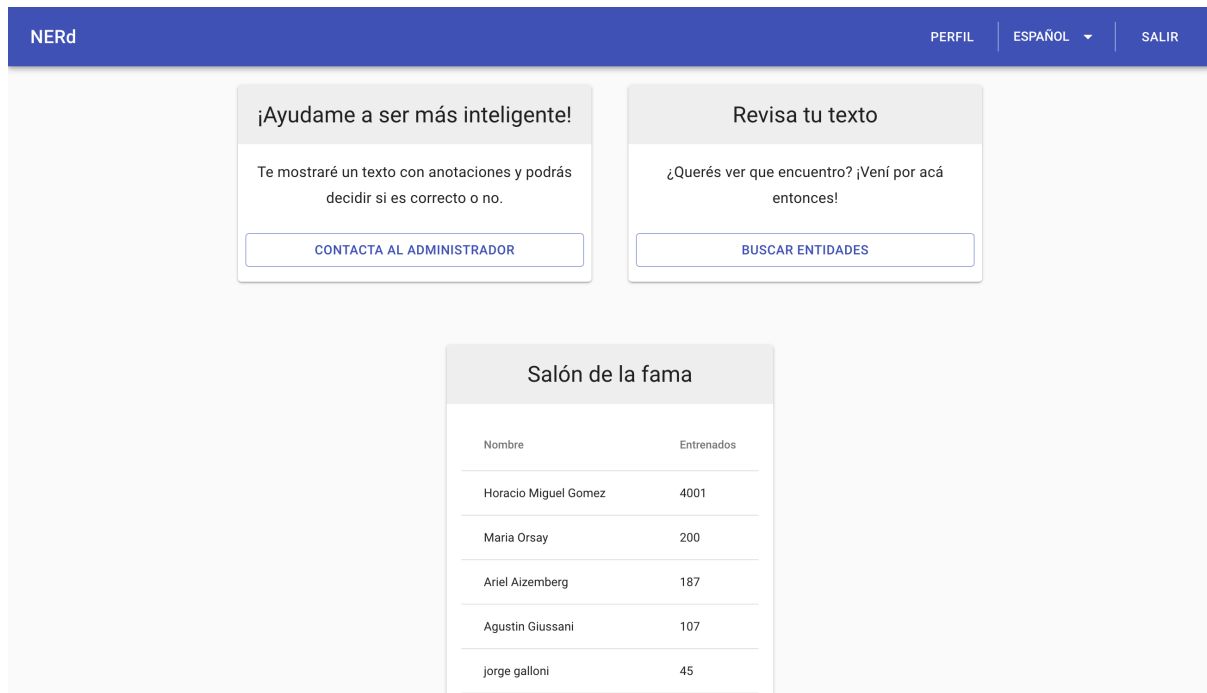


Figura 8: Pantalla de inicio sin rol de entrenador

Si la persona visitando la página no cuenta con una sesión activa, se le invita a ingresar con una cuenta pre-existente o a registrarse.

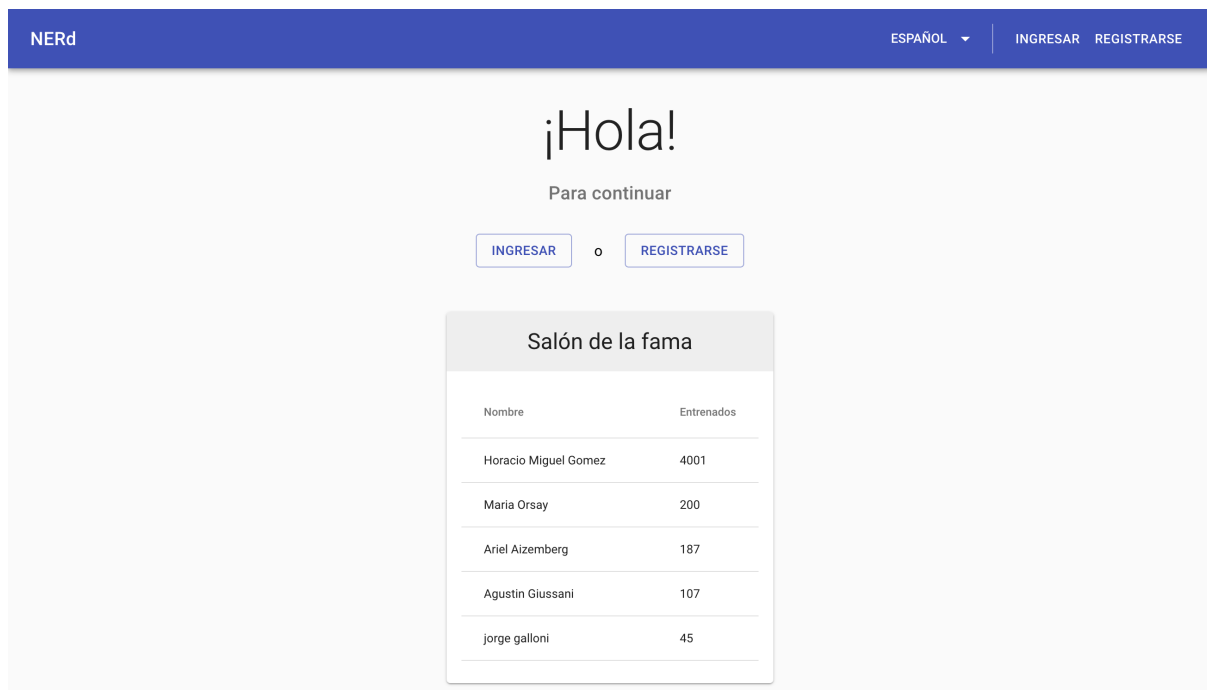


Figura 9: Pantalla de inicio sin sesión

4.1.2.2. Entrenamiento

La pantalla de *Entrenamiento* es el núcleo de la web en la cual es posible entrenar el modelo.

El usuario es presentado con un texto perteneciente al Corpus del servicio con las entidades inferidas por el modelo actual. Con un editor especial, le permitimos al usuario poder corregir las entidades inferidas y enviarle la corrección al servicio. Esa corrección será utilizada posteriormente a la hora de mejorar el modelo actual.



Figura 10: Pantalla de entrenamiento

4.1.2.2.1. Usabilidad

Tuvimos un foco fuerte en la usabilidad del widget ya que los entrenadores del servicio van a pasar prácticamente todo su tiempo en ésta pantalla, por lo que se tuvieron las siguientes consideraciones en la implementación.

Llamado a acción y ayuda

Dado que lo primero que ve el usuario es un texto con anotaciones, agregamos un título que invita al usuario a realizar acciones sobre el texto. De esta manera, le mostramos las dos acciones principales realizables desde el widget de entrenamiento: Click en alguna palabra o entidad y arrastrar un conjunto de palabras para crear una entidad nueva. Como refuerzo de este llamado a acción, agregamos un botón que al ser clickeado muestra un mensaje de ayuda con instrucciones más detalladas sobre el objetivo del entrenador y las acciones que deben de realizarse sobre el mismo.



Figura 11: Ayuda del entrenador

Creación y edición de entidades

Para la creación de entidades decidimos ofrecer dos maneras: La primera es arrastrando un conjunto de palabras de manera tal de unir las todas en una única entidad. La otra es hacer click en una palabra y ahí se ofrecen opciones dependiendo de la ubicación de la palabra dentro del texto:

- Si no existen entidades en el texto actual, se le asigna por defecto el tipo *MISC* y se muestran el resto de los tipos para permitir cambiarlo de ser necesario.
- Si existen entidades antes o después, se ofrece la opción de unir la palabra actual con la entidad más próxima para el lado elegido.



Figura 12: Edición de entidad

Para la edición de entidades decidimos permitir únicamente la modificación del tipo de una entidad inferida. Si el modelo inferió una entidad de manera incorrecta, ya sea por que sea una entidad inválida o agregó palabras de más a una entidad inválida, permitimos que el usuario remueva la entidad y que después vuelva a agregar la entidad correcta.

Optimización en tiempos de carga

Dado que es esperado que un usuario entrene más de un texto, al momento de pedir un texto para mostrar, se pide el siguiente. Mediante este mecanismo de pre-carga, podemos eliminar el tiempo de espera entre texto y texto ofreciendo al usuario una experiencia completamente fluida.

4.1.2.3. Administración de usuarios

La pantalla de *Administración de usuarios* permite a los usuarios con el rol de administrador poder modificar los roles de todos los usuarios del sistema, borrarlos o acceder a los detalles del usuario, tal como la lista de textos entrenados.

Administración de usuarios				
PERFIL USUARIOS CORPUS COMPARAR ESTADO ESPAÑOL ▾ SALIR				
Usuarios 🔍 Search...				
<input type="checkbox"/>	Nombre	Email	Roles	Entrenamientos
<input type="checkbox"/>	Admin	admin@example.com	user ▾	0
<input type="checkbox"/>	Juan Pablo Orsay	jorsay@itba.edu.ar	user trainer admin ▾	19
<input type="checkbox"/>	Horacio Miguel Gomez	hogomez@itba.edu.ar	user trainer admin ▾	4001
<input type="checkbox"/>	Martin	mcapparelli@itba.edu.ar	user trainer ▾	0
<input type="checkbox"/>	Pablo Alejandro Costesich	pablo.costesich@gmail.com	user trainer ▾	0
<input type="checkbox"/>	Ariel Aizemberg	aaizemberg@itba.edu.ar	user trainer admin ▾	187
<input type="checkbox"/>	Cris	cristian@elciudadano.cl	user ▾	0
<input type="checkbox"/>	Alejandro Vaisman	alejandro.vaisman@gmail.com	user ▾	0

Figura 13: Administración de usuarios

4.1.2.4. Detalles de usuario

La pantalla de detalle de usuario permite al usuario con sesión activa ver sus entrenamientos y cambiar su contraseña.

Los usuarios con rol administrador pueden realizar las acciones mencionadas previamente pero a otros usuarios.

Mi perfil

PERFIL | USUARIOS | CORPUS | COMPARAR | ESTADO | ESPAÑOL ▾ | SALIR

Cambiar contraseña

Contraseña * Confirmar contraseña *

ACTUALIZAR

Entrenamientos

☐ Entrenamientos

- ☐ **PER** Matías Martin habló de la desvinculación de **PER** Cabito de "Basta de Todo": "Es una decisión dolorosa , de la cual me hago ..."
- ☐ En **DATE** 2018 , el **MISC** déficit comercial con **LOC** Brasil se redujo a la mitad
- ☐ La reacción del **PER** Papa cuando una joven interrumpió su audiencia y se puso a jugar
- ☐ La ciencia empieza a entender el **MISC** alzhéimer
- ☐ Acusan a empleados de un súper de golpear y matar a un jubilado que había robado chocolates , un queso y una botella de aceite

Figura 14: Perfil de usuario

4.1.2.5. Corpus

La pantalla de *Corpus* permite a un usuario con el rol de administrador realizar tareas relacionadas con el corpus del sistema.

Corpus

PERFIL | USUARIOS | CORPUS | COMPARAR | ESTADO | ESPAÑOL ▾ | SALIR

Subir

[SUBIR ARCHIVO](#)

Textos

<input type="checkbox"/>	Texto	Añadido	Entrenamientos
<input type="checkbox"/>	[Elecciones 2019] Bomba electoral: en Córdoba, el PJ ganó en 28 comunas gobernadas por otros partidos	28 de agosto de 2019 22:47	0
<input type="checkbox"/>	¿Acto fallido?: "Nuestro candidato es María Eug... Macri"	28 de agosto de 2019 22:47	1
<input type="checkbox"/>	¿Cómo combatir las noticias falsas durante las elecciones en Argentina?	28 de agosto de 2019 22:47	1
<input type="checkbox"/>	¿Cómo combatir las noticias falsas?	28 de agosto de 2019 22:47	0
<input type="checkbox"/>	¿Cómo entender los resultados de las elecciones en Córdoba?	28 de agosto de 2019 22:47	1
<input type="checkbox"/>	¿Cómo votan los jóvenes? Pistas para entender las demandas de este electorado	28 de agosto de 2019 22:47	1
<input type="checkbox"/>	¿Cuál es la diferencia entre el voto en blanco, el voto nulo e impugnado?	28 de agosto de 2019 22:47	0
<input type="checkbox"/>	¿Cuáles son las dos opciones que tiene Sergio Massa en estas elecciones?	28 de agosto de 2019 22:47	1

Figura 15: Administración de corpus

Desde aquí es posible agregar textos al corpus utilizando la funcionalidad de subida de archivos. Los archivos deben ser archivos con extensión *.txt* y cada línea del archivo será agregada al corpus como un texto individual.

También es posible desde aquí ver todos los textos que forman parte del corpus así como también poder ver los entrenamientos para cada uno de los textos. Finalmente, es posible quitar textos del corpus así como también es posible eliminar correcciones a las inferencias de entidades cargados por usuarios.

4.1.2.6. Estado

La pantalla de *Estado* permite a un usuario con el rol de administrador visualizar el estado de entrenamiento del corpus así como también realizar diversas acciones sobre los *workers*.

The screenshot shows the 'Administración del Corpus' interface. On the left, a sidebar displays corpus statistics: 'Corpus' with a 'VER' button, 'Estado' showing '4589 de 43751 entrenados', 'Total entrenados' as '4590', and 'Sin entrenar' as '0'. The main area is titled 'Crear snapshot' and includes a 'Tipos:' section with filters for DATE, LOC, MISC, ORG, and PER, along with a 'NUEVO' button and a 'CREAR' button. Below this is a 'Snapshots' table with columns for checkboxes, Version, Creado el, Último entrenamiento, Estado, Trabajadores, and Acciones. The table lists four snapshots: 'Actual', '1', '2', and '3'. At the bottom, there is a 'Reasignar trabajador' section with a dropdown menu showing 'vCURRENT' and 'a v1', and an 'APLICAR' button.

<input type="checkbox"/>	Version	Creado el	Último entrenamiento	Estado	Trabajadores	Acciones
<input type="checkbox"/>	Actual	7 de octubre de 2019 17:02	hace 3 horas	Listo	1	
<input type="checkbox"/>	1	7 de octubre de 2019 7:59	hace un mes	Listo	1	
<input type="checkbox"/>	2	7 de octubre de 2019 8:57	Nunca	Listo	Ninguno	
<input type="checkbox"/>	3	7 de octubre de 2019 8:57	hace 5 días	Entrenando	Ninguno	

Figura 16: Información de corpus y manejo de workers

4.1.2.6.1. Secciones

Corpus

Es la columna la izquierda y aquí se puede ver rápidamente que porcentaje de el corpus contiene correcciones por usuarios así como también saber la cantidad total de correcciones del sistema (un texto puede tener más de una corrección por distintos usuarios) y también presenta un botón que permite al administrador ir a la pantalla de *Corpus*.

Crear snapshot

Es la sección en la cual será posible crear, borrar o modificar los tipos de entidades reconocidos por el snapshot actual. La acción de editar las entidades genera un snapshot nuevo.

Si el administrador así lo quisiera, puede utilizar esta sección para crear un snapshot nuevo sin editar entidades.

Snapshots

Sección en la cual podemos ver la lista completa de snapshots. Para cada Snapshot, se muestra cuando fue la última vez que se entrenó así como también cuantos trabajadores tiene asignados. Finalmente es posible desde aquí forzar a entrenar el modelo para ese snapshot en particular y también se presenta la opción para desentrenar, borrando el modelo guardado en el disco.

Reasignar trabajador

Sección que permite reasignar trabajadores para que sirvan un snapshot distinto. De esta manera se pueden servir distintas versiones del modelo de inferencia para poder realizar distintas pruebas sobre los mismos.

4.1.2.7. Sandbox

La pantalla de *Sandbox* permite a los usuarios hacer consultas al servicio NERd para poder obtener entidades nombradas a partir de textos arbitrarios. Adicionalmente, si el usuario tiene el rol de entrenador, podrá corregir las entidades inferidas y agregar el texto con sus correcciones al corpus.

The screenshot shows the 'Sandbox de NER' interface. At the top is a blue navigation bar with a home icon, the title 'Sandbox de NER', and several menu items: 'PERFIL', 'USUARIOS', 'CORPUS', 'COMPARAR', 'ESTADO', 'ESPAÑOL' (with a dropdown arrow), and 'SALIR'. Below the navigation bar is a large text input area. Inside this area, there is a placeholder text: 'El director técnico argentino Mauricio Pochettino fue despedido este martes por el Tottenham de Inglaterra, club al que dirigió durante cinco años y al que condujo a la final de la pasada edición de la Champions League.' To the right of the input area is a blue button labeled 'BUSCAR ENTIDADES'. Below the input area, the same sentence is displayed with entities highlighted in colored boxes: 'PER' for 'Mauricio Pochettino', 'ORG' for 'Tottenham', 'LOC' for 'Inglaterra', and 'MISC' for 'Champions League'. To the right of this displayed text is a blue button labeled 'GUARDAR'.

Figura 17: Inferencia de entidades en sandbox

4.1.2.8. Comparar

Sección accesible únicamente a administradores en la que es posible comparar las entidades inferidas por dos modelos distintos. A su vez, si el usuario logueado tiene el permiso de entrenador, es posible corregir de manera inline los errores en la inferencia del modelo actual.



Comparar snapshots

PERFIL | USUARIOS | CORPUS | COMPARAR | ESTADO | ESPAÑOL | SALIR

v1 y vCURRENT COMPARAR

Marcar distintos

v1 vCURRENT

[MISC Elecciones 2019] [PER Bomba] electoral : en [LOC Córdoba], el [ORG PJ] ganó en 28 comunas gobernadas por otros partidos

¿ Acto fallido ? : " Nuestro candidato es [PER María Eug ...]

¿ Cómo combatir las noticias falsas durante las elecciones en [LOC Argentina] ?

¿ Cómo combatir las noticias falsas ?

¿ Cómo entender los resultados de las elecciones en [MISC Córdoba] ?

¿ Cómo votan los jóvenes ? Pistas para entender las demandas de este electorado

[MISC ¿ Cuál] es la diferencia entre el voto en blanco , el

[MISC Elecciones 2019] Bomba electoral : en [LOC Córdoba], el [ORG PJ] ganó en 28 comunas gobernadas por otros partidos

¿ Acto fallido ? : " Nuestro candidato es María Eug ...

¿ Cómo combatir las noticias falsas durante las elecciones en [LOC Argentina] ?

¿ Cómo combatir las noticias falsas ?

¿ Cómo entender los resultados de las elecciones en [LOC Córdoba] ?

¿ Cómo votan los jóvenes ? Pistas para entender las demandas de este electorado

¿ Cuál es la diferencia entre el voto en blanco , el voto nulo

CORREGIR

Figura 18: Comparativa de modelos

4.2. Vista de proceso

La vista de proceso trata los aspectos dinámicos del sistema, explica los procesos del sistema y cómo se comunican, y se centra en el comportamiento del sistema en tiempo de ejecución. La vista de proceso aborda concurrencia, distribución, integradores, rendimiento y escalabilidad, etc.

4.3. Vista de desarrollo

La vista de desarrollo ilustra un sistema desde la perspectiva de un programador y se ocupa de la gestión de software. Esta vista también se conoce como la vista de implementación.

Python es el lenguaje más utilizado para resolver problemas de Machine Learning, en especial NLP ("The state of the octoverse," 2019)

Spacy es el framework mejor ranqueado para la tarea de NLP ("The state of the octoverse," 2019). Su implementación es robusta y orientada a la implementación de aplicaciones en producción, a diferencia de muchas otras librerías de NLP que sólo se utilizan con fines académicos.

4.4. Vista física

La vista física representa el sistema desde el punto de vista de un ingeniero de sistemas. Se refiere a la topología de los componentes de software en la capa física, así como

a las conexiones físicas entre estos componentes. Esta vista también se conoce como la vista de *deployment*.

4.5. Escenarios

La descripción de una arquitectura se ilustra utilizando un pequeño conjunto de casos de uso, o escenarios, que se convierten en una quinta vista. Los escenarios describen secuencias de interacciones entre objetos y entre procesos. Se utilizan para identificar elementos arquitectónicos y para ilustrar y validar el diseño de la arquitectura. También sirven como punto de partida para las pruebas de un prototipo de arquitectura. Esta vista también se conoce como vista de caso de uso.

5. Resultados

6. Discusión

6.1. Tipos de entidades relevantes

tener en cuenta (Brunstein, 2002)

Notas sobre mejora en tipos de entidades

Presidente -> Person Descriptor

NORP -> (Polical) Peronistas, Kirchneristas

Facility Name -> usually location. "Wall Street", "Muralla China"

Organization Name -> Government vs Corporation.

Product Name -> autos "Fiat Toro", celulares "Galaxy S10"

Events -> Superclásico. Superliga. Copa argentina. Elecciones 2019. Las Paso.

Disease ->

Game -> Football, Basket (para "titulos" no tan relevante)

6.2. Seed en los types

en especial para los nuevos.

6.3. Mejora live vs offline

Mejora «Uncertainty sampling» -> buscar entidades que tengan un score ~ 0.5

6.4. Utilidad de la herramienta

Para poder poner a prueba nuestra herramienta **NERd** en un entorno real participamos de la hackaton en MediaParty 2019.

(“**Hackaton,” 2019**) es un evento de tres días en Argentina, que reúne a 2500 emprendedores, periodistas, programadores de software y diseñadores de cinco continentes para trabajar juntos para el futuro de los medios de comunicación. Nacido de Hacks/Hackers Buenos Aires, el evento fusiona a grandes empresas como New York Times, The Guardian, Vox, ProPublica, Watchup, Neo4J o DocumentCloud y comunidades regionales de la mayor red de periodistas y desarrolladores del mundo.

Participamos en conjunto con otro proyecto final en el que van a utilizar nuestra API para hacer detección de entidades en documentos PDF.

La experiencia fue muy satisfactoria, recibimos buenas críticas sobre la Usabilidad de nuestra aplicación y la gran utilidad que presta a la comunidad.

Por tal motivo recibimos el primer premio de dicha hackaton (“Mención itba,” 2019)

7. Conclusiones

7.1. Examples

You can label chapter and section titles using `{#label}` after them, e.g., we can reference Chapter 1. If you do not manually label them, there will be automatic labels anyway, e.g., Chapter 2.

Figures and tables with captions will be placed in `figure` and `table` environments, respectively.

```
par(mar = c(4, 4, .1, .1))  
plot(pressure, type = 'b', pch = 19)
```

Reference a figure by its code chunk label with the `fig:` prefix, e.g., see Figure 19. Similarly, you can reference tables generated from `knitr::kable()`, e.g., see Table 1.

```
knitr::kable(  
  head(iris, 20), caption = 'Here is a nice table!',  
  booktabs = TRUE  
)
```

You can write citations, too. For example, we are using the **bookdown** package (Xie, 2019) in this sample book, which was built on top of R Markdown and **knitr** (Xie, 2015).

Brunstein, A. (2002). Annotation guidelines for answer types. Retrieved from <https://catalog.ldc.upenn.edu/docs/LDC2005T33/BBN-Types-Subtypes.html>

Duh definition. (2019). Retrieved October 14, 2019, from <https://dictionary.cambridge.org/es/diccionario/ingles/duh>

Ethayarajh, K., Duvenaud, D., & Hirst, G. (2019). Towards understanding linear word analogies. *Proceedings of the 57th annual meeting of the association for computational linguistics*, 3253–3262. <https://doi.org/10.18653/v1/P19-1315>

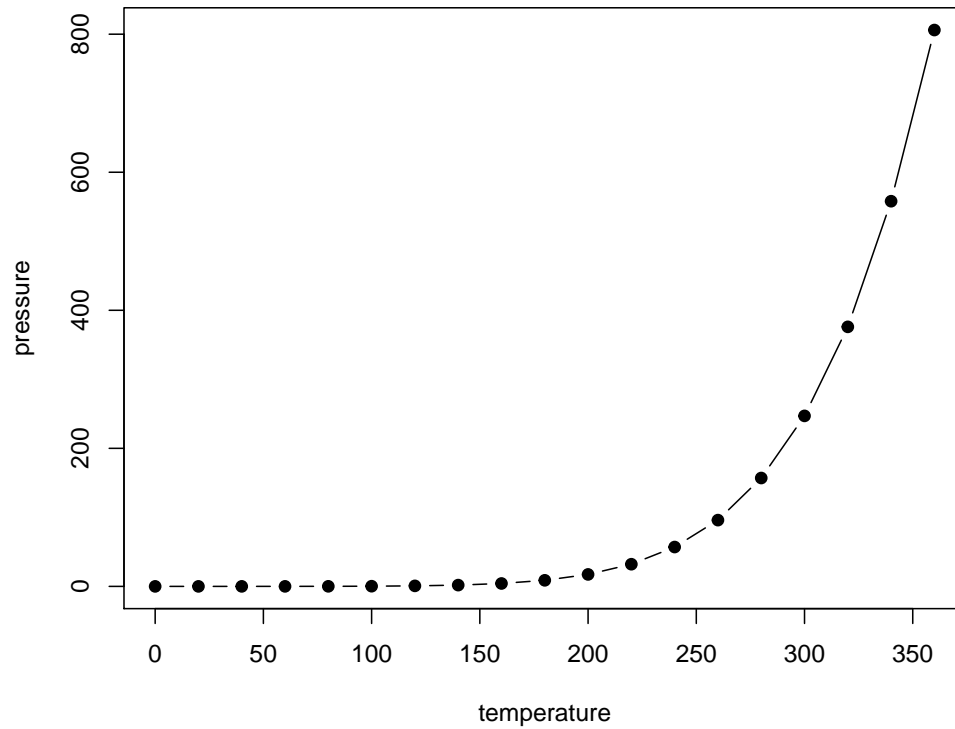


Figura 19: Here is a nice figure!

Cuadro 1: Here is a nice table!

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
5.0	3.4	1.5	0.2	setosa
4.4	2.9	1.4	0.2	setosa
4.9	3.1	1.5	0.1	setosa
5.4	3.7	1.5	0.2	setosa
4.8	3.4	1.6	0.2	setosa
4.8	3.0	1.4	0.1	setosa
4.3	3.0	1.1	0.1	setosa
5.8	4.0	1.2	0.2	setosa
5.7	4.4	1.5	0.4	setosa
5.4	3.9	1.3	0.4	setosa
5.1	3.5	1.4	0.3	setosa
5.7	3.8	1.7	0.3	setosa
5.1	3.8	1.5	0.3	setosa

Hackaton. (2019). Retrieved August 31, 2019, from <https://mediaparty.info/>

Kruchten, P. (1995). The 4+1 view model of architecture. *IEEE Softw.*, 12(6), 42–50. <https://doi.org/10.1109/52.469759>

Mención itba. (2019). Retrieved October 3, 2019, from <https://www.instagram.com/p/B3Koum2peD-/>

The state of the octoverse: Machine learning. (2019). Retrieved January 24, 2019, from <https://github.blog/2019-01-24-the-state-of-the-octoverse-machine-learning/>

Xie, Y. (2015). *Dynamic documents with R and knitr* (2nd ed.). Retrieved from <http://yihui.name/knitr/>

Xie, Y. (2019). *Bookdown: Authoring books and technical documents with r markdown*. Retrieved from <https://github.com/rstudio/bookdown>