



Universidad de Buenos Aires  
Facultad de Ingeniería  
Departamento de Computación

## Propuesta de Trabajo Profesional para Ingeniería en Informática

### Aplicación de Redes Neuronales para Detección de Mentiras

#### **Alumnos**

Riesgo, Daniela Paula 95557

Esteban, Federico Martín 94557

#### **Tutor**

Lic. Luis Argerich

# Índice

<b>Introducción</b>	<b>2</b>
<b>Descripción del Problema</b>	<b>2</b>
<b>Estado del Arte</b>	<b>3</b>
<b>Objetivos</b>	<b>4</b>
<b>Características del Trabajo</b>	<b>4</b>
Módulo de detección de mentiras	4
Interfaz Web	4
Framework open-source	4
<b>Tecnologías</b>	<b>5</b>
Framework Principal	5
Herramientas y otras tecnologías	5
<b>Alcance</b>	<b>5</b>
<b>Plan de Trabajo</b>	<b>6</b>
Equipo de Trabajo	6
Metodología	6
Estimación	6
Cronograma de entregables	8
<b>Bibliografía</b>	<b>9</b>

# Introducción

El siguiente documento presenta la propuesta de Trabajo Profesional de Ingeniería en Informática de los estudiantes Federico Martín Esteban (padrón 94557) y Daniela Paula Riesgo (padrón 95557).

El detectar mentiras es una cualidad muy difícil de adquirir para las personas, pero que posee un valor muy grande en un gran número de situaciones. Ya sea en la corte o en la policía, en un interrogatorio de aeropuerto o hasta en la vida personal de cualquiera cuando le pregunta algo a sus hijos, encontrar la verdad es siempre el fin que se quiere lograr para actuar en consecuencia. Hay muchos crímenes asociados a la mentira, como el perjurio o el fraude. Encontrar las mentiras lo antes posible ahorra tiempo y dinero, y puede prevenir consecuencias triviales o serias como la seguridad.

El objetivo final del proyecto es desarrollar un sistema que pueda detectar cuando una declaración de una persona es verídica o no a partir de un video de dicha declaración.

## Descripción del Problema

El trabajo a realizar presenta una serie de problemas no triviales a resolver:

- Se necesita conseguir una base de datos muy amplia de videos para el entrenamiento de las redes neuronales a utilizar. Ante la baja probabilidad de conseguir una, anticipamos que será necesario construir una.
- Se necesita determinar el mejor método para procesar los datos y conseguir el output esperado. Se plantean las siguientes maneras de resolver el problema:
  - Opción 1: Hacer un procesamiento de los videos crudos utilizando una red neuronal, ajustando los parámetros para que la misma pueda aprender utilizando únicamente los videos y su etiqueta como entrada.
  - Opción 2: Hacer un análisis de sentimiento de los videos mediante algún software (como Affectiva[1]) y utilizar ese output junto con los videos como input para una red neuronal.
  - Opción 3: Hacer un análisis de sentimiento tanto del video como de lo dicho por la persona y utilizar estos outputs como input para una red neuronal. Vale aclarar que el análisis de lo dicho sería principalmente sobre su contenido y para esto se usaría software de traducción de audio a texto y de análisis de sentimiento de un texto. Para esto último existen servicios como Indico[2].

## Estado del Arte

Por las razones mencionadas y muchas otras, este tema se ha convertido en área de estudio desde el 1900[3]. Comenzó con varios estudios que medían signos biológicos, como la relación inspiración-espирación o la presión sanguínea y buscaban sus relaciones con el mentir. Luego, los estudios fueron evolucionando a análisis más visuales, con métodos menos invasivos.

Una de las investigaciones más importantes del área de detección de mentiras (o Deception Detection en inglés), fue iniciada por el psicólogo Paul Ekman[4] a fines de los años 50. Actualmente, la línea de investigación está siendo llevada a cabo por el denominado Paul Ekman Group[5]. La teoría detrás utiliza el concepto de FACS (Facial Action Coding System) para identificar sentimientos. Junto con esto, utiliza también el concepto de de micro expresiones que revelan los verdaderos sentimientos de la persona y que son imposibles de ocultar. Éstas están presentes en todas las personas independientemente de su origen; y con entrenamiento pueden ser detectadas por seres humanos. El grupo, provee herramientas pagas para realizar estos entrenamientos, pero no provee ningún tipo de software que permita hacer esta detección automáticamente y reemplazar el trabajo de los seres humanos.

El primer software que implementa el concepto de FACS es el denominado CERT (The Computer Expression Recognition Toolbox)[6]. El software permite detectar sonrisas, la pose, y realizar un reconocimiento básico de emociones. El mismo se publicó junto con el paper de manera gratuita pero no se pudo encontrar ninguna versión del mismo para descargar.

Actualmente existe un software que se base en el concepto de FACS para detectar emociones en video en tiempo real, con el nombre de Afectiva[1]. Actualmente provee por un lado un SDK para reconocimiento de emociones a través de video y por otro lado una API de servicio para realizar el procesamiento de manera remota (en la nube). También posee un SDK para realizar el reconocimiento de emociones a través de audio.

Por otro lado, hay una gran cantidad de programas en Internet que aclaman detectar mentiras mediante el análisis de la voz. Tal es el caso de X13-VSA[7], TRUSTER[8], eX-Sense[9]. La eficacia de estos productos no está confirmada por ninguna fuente oficial.

## Objetivos

El Trabajo Profesional consta de 3 objetivos:

- Desarrollar el módulo para detección de mentiras en videos.

- Desarrollar una aplicación web que sirva como interfaz con el usuario para que pueda hacer uso del módulo creado.
- Desarrollar el framework open-source para manejar el módulo de detección de mentiras.

## Características del Trabajo

### Módulo de detección de mentiras

El módulo de detección de mentiras es un módulo de clasificación que estará basado en la utilización de una red neuronal de tipo convolucional (Convolutional Neural Network)[10]. Éste es un tipo de red neuronal que utiliza el concepto de back propagation para el aprendizaje y actualmente son la configuración de preferencia (por sus resultados) a utilizar frente a un problema de procesamiento de imágenes o video.

Para realizar este módulo, se utilizará la herramienta TensorFlow[11], librería que facilita el desarrollo de algoritmos de Machine Learning.

### Interfaz Web

La interfaz web debe permitirle al usuario proveer un video de una persona hablando para que el módulo de detección de mentiras pueda detectar en éste si la persona se encuentra mintiendo. Deberá mostrarle al usuario esta información siguiendo estándares de diseño web.

### Framework open-source

El framework deberá tener una interfaz útil y amigable para que otros programadores puedan utilizarla en sus proyectos, siguiendo heurísticas de usabilidad y principios de diseño. Este framework permitirá a otros desarrolladores utilizar este módulo de detección de mentiras dentro de sus aplicaciones de forma fácil.

Además, incluirá la base de datos construida para que pueda usar por otros desarrolladores para proyectos de distintos propósitos.

# Tecnologías

## Framework Principal

El framework principal de trabajo será TensorFlow[11]. El mismo es una librería open source utilizada para cálculos numéricos que funciona mediante el uso de grafos de flujos de datos. Está disponible para algunos lenguajes de programación específicos.

## Herramientas y otras tecnologías

Categoría	Herramientas
Lenguajes de Programación	<ul style="list-style-type: none"> <li>• Módulo de detección: Python</li> <li>• Interfaz Web: Javascript</li> </ul>
Entorno de Desarrollo	<ul style="list-style-type: none"> <li>• Atom[12]</li> <li>• Sublime Text[13]</li> </ul>
Control de Versiones	<ul style="list-style-type: none"> <li>• Git[14]</li> </ul>
Administración y Control de Proyecto	<ul style="list-style-type: none"> <li>• Google Drive[15]</li> <li>• Trello[16]</li> </ul>

## Alcance

El alcance del presente trabajo incluye:

- Desarrollo del módulo de detección de mentiras en videos
  - Obtención de datos para entrenar la red neuronal
  - Evaluación de la Opción 1 y la Opción 2
    - Diseño de la red neuronal
    - Entrenamiento de la red y pruebas
- Interfaz web para la utilización del módulo
  - Diseño de la interfaz
  - Desarrollo de la interfaz
- Herramientas open source
  - Organización y publicación de la base de datos conseguida
  - Organización y publicación del módulo de detección de mentiras
  - Organización y publicación de la aplicación para el usuario final

# Plan de Trabajo

## Equipo de Trabajo

Tutor: Lic. Luis Argerich.

Desarrolladores: Federico Martín Esteban y Daniela Paula Riesgo.

## Metodología

Para la implementación del proyecto se llevará a cabo una metodología basada en Scrum, para la cual se definirán una serie de iteraciones con entregables a la finalización de cada una de ellas, pautando para ello fechas de reunión y avance.

Al comienzo de cada iteración se analizarán las prioridades de los distintos requerimientos en conjunto con el tutor, luego se establecerán fechas para los entregables parciales (de ser considerados necesarios), y al finalizar la iteración se realizará una reunión para la presentación del entregable de la iteración.

## Estimación

A continuación se muestra un listado de tareas necesarias para alcanzar los objetivos del proyecto, con los esfuerzos estimados de los mismos. Los esfuerzos, en todos los casos, se encuentran expresados en horas.

# Iteración	Descripción	Esfuerzo	Recursos	Total
-	Propuesta	15	2	30
1	Configuración de Entorno Desarrollo sistema para la recolección de datos (diseño de preguntas y grabación y etiquetado de videos)	15	1	15
		15	2	30
2	Grabación de set inicial de datos de prueba	10	2	20
	[Opción 1] Investigación y prueba de TensorFlow	30	2	60
	[Opción 2] Prueba de los datos con sistemas existentes de detección de	15	1	15

	emociones			
3	Expansión del set de datos	5	2	10
	[Opción 1] Investigación y diseño inicial de la red neuronal	40	2	80
	[Opción 2] Investigación y diseño inicial de la red neuronal	40	2	80
4	Expansión del set de datos	5	2	10
	[Opción 1] Pruebas y rediseño	35	2	70
	[Opción 2] Procesamiento de los nuevos datos en detección de emociones	5	1	5
	[Opción 2] Pruebas y rediseño	35	2	70
5	Expansión del set de datos	5	1	5
	Pruebas y refinamiento de la red	20	2	40
6	Expansión del set de datos	5	1	5
	Pruebas y refinamiento de la red	10	2	20
	Desarrollo de la interfaz web	10	2	20
7	Definición final de la base de datos	5	1	5
	Desarrollo del framework	10	2	20
	Integración de los sistemas	25	2	50
-	Reuniones	15	2	30
-	Presentación	15	2	30

Además, se tiene en cuenta el tiempo dedicado a la administración del proyecto, el cual estimaremos alrededor del 10% del tiempo de desarrollo. El resultado final de la estimación es:

Descripción	Esfuerzo
Iteración 1	45
Iteración 2	95
Iteración 3	170
Iteración 4	155
Iteración 5	45
Iteración 6	45



Iteración 7	75
Otros	90
Administración	72
<b>Esfuerzo Total</b>	<b>792</b>

## Cronograma de entregables

A continuación se hace un cronograma tentativo de entregables al finalizar cada una de las iteraciones. Las mismas pueden ser modificadas en caso de verse necesario, por el tutor o por los desarrolladores del proyecto. La fecha de cada una de las entregas serán determinadas en conjunto con el tutor.

# Iteración	Entregables	Hitos
1	Sistema para recolección de datos	Configuración del entorno de desarrollo terminado
2	Primer versión de la base de datos	Tutorial de Tensorflow terminado
3	Implementación básica de la red neuronal Análisis inicial de evaluación de alternativas	Análisis y diseño iniciales terminados
4	Implementación de la segunda versión de la red neuronal Segundo análisis de evaluación de alternativas	Elección de la alternativa a utilizar Nuevos datos conseguidos Modificaciones terminadas
5	Implementación de la tercera versión de la red neuronal Informe de pruebas	Nuevos datos conseguidos Modificaciones terminadas
6	Primera versión de la aplicación web	Nuevos datos conseguidos Modificaciones terminadas
7	Base de datos final Aplicación web final Framework open source	Red neuronal terminada Sistemas satélites terminados

# Bibliografía

[1] Afectiva: Software de detección de emociones.

<https://www.affectiva.com>

[2] Indico: Sentiment Analysis SaaS.

<https://indico.io/>

[3] Artículo sobre la historia de la detección de mentiras.

<http://scholarlycommons.law.northwestern.edu/cgi/viewcontent.cgi?article=2844&context=jclc>

[4] Biografía de Paul Ekman, The Paul Ekman Group.

<https://www.paulekman.com/paul-ekman/>

[5] The Paul Ekman Group.

<https://www.paulekman.com/about-paul-ekman-group-llc/>

[6] Computer Expression Recognition Toolbox, Marian Bartlett et al.

[http://mplab.ucsd.edu/~marni/pubs/Bartlett\\_demo\\_FG2008.pdf](http://mplab.ucsd.edu/~marni/pubs/Bartlett_demo_FG2008.pdf) (6/12/2017)

[7] X13-VSA: Software de Detección de Mentiras por Audio.

<http://lie-detection.com/>

[8] Truster: Software de Detección de Mentiras por Audio.

<http://www.pimall.com/nais/truster.html>

[9] eX-Sense: Software de Detección de Mentiras por Audio.

<http://exsense.trusters.com/>

[10] Backpropagation Applied to Handwritten Zip Code Recognition, Y. LeCun et al.

<http://yann.lecun.com/exdb/publis/pdf/lecun-89e.pdf>

[11] TensorFlow: Librería de software para Machine Intelligence.

<https://www.tensorflow.org/>

[12] Atom: Editor de texto.

<https://atom.io>

[13] SublimeText: Editor de texto.

<https://www.sublimetext.com>

[14] Git: Software para control de versiones.

<https://git-scm.com>

[15] GoogleDrive: Almacenamiento en la nube

<https://www.google.com/drive/>

[16] Trello: Software para Gestión y Seguimiento de Proyectos.

<https://trello.com/>