

Análisis de sentimientos de noticias escritas usando un modelo basado en la red neuronal *long short-term memory* para determinar si las noticias positivas mejoran el estado de ánimo de las personas

Gustavo Adolfo Reyes-Paredes
kivada@icloud.com / Universidad de Lima, Perú

Recepción: 17-6-2019 / Aceptación: 8-8-2019

RESUMEN. Es un hecho que el paradigma de distribuir noticias negativas a la población es el más aceptado mundialmente. Una gran cantidad de investigaciones se han enfocado en establecer los efectos de este paradigma en la población y, en todos los casos, se ha demostrado que es dañino para la salud y el comportamiento de las personas. Por ello, se ha decidido demostrar que el paradigma opuesto, la distribución de noticias positivas, genera una mejora en la salud, en el comportamiento y en el estado de ánimo de la población. Para lograr este propósito, se desarrolló un modelo basado en la red neuronal *long short-term memory* para realizar el análisis de los sentimientos relacionados con las noticias escritas en español. El experimento consistió en determinar el estado de ánimo de las personas luego de haber leído noticias positivas.

PALABRAS CLAVE: aprendizaje de máquinas, análisis de sentimientos, red neuronal recurrente, *long short-term memory*, bienestar psicológico y social

Sentiment Analysis of Written News Using a Model Based on the Long Short-Term Memory Neural Network to Determine if Positive News Improve People's Mood

ABSTRACT. It is a fact that the paradigm of distributing negative news to the population is the most accepted worldwide. A large amount of research has been done to determine the effects of this paradigm on the population and, in all cases, it has been shown to be harmful to the health and behavior of people. Therefore, this paper aims to demonstrate that the opposite paradigm, the distribution of positive news, generates an improvement in the health, behavior and mood of the population. To achieve this, a model based on the long short-term memory neural network has been developed in order to analyze sentiments caused by news written in Spanish. Moreover, an experiment was conducted to determine people's mood after having read positive news.

KEYWORDS: machine learning, sentiment analysis, recurrent neural network, long short-term memory, psychological and social well-being

1. INTRODUCCIÓN

En la actualidad, el paradigma de difundir noticias negativas es el más aceptado, por ello la gran mayoría de los medios de comunicación las muestran a la población. Se ha demostrado que las personas se sienten atraídas hacia los eventos negativos en un nivel genético (Hatemi *et al.*, 2009). Por ello, no es sorprendente que la prensa mundial siga el paradigma de difundir noticias negativas, ya que esto genera un mayor ingreso monetario.

Por otro lado, es de suma importancia mantener un balance emocional positivo; de no lograrse, ocasiona una gran cantidad de problemas de salud afectando la vida diaria de las personas. La incapacidad de poder alcanzar un balance emocional es un problema actual y creciente que repercute en todo el mundo (Fredrickson, 2001); agravándose aún más por la proliferación de noticias negativas en los medios de comunicación, debido a que estas afectan la salud mental de la población si se exponen constantemente a ellas (Johnston y Davey, 1997).

Para reducir estos problemas existe otro paradigma basado en la exposición a la población de noticias positivas que nos indica que el estado de ánimo de las personas mejora luego de estar en contacto con noticias positivas. Por lo tanto, para corroborar estas investigaciones se propone realizar un modelo basado en la red neuronal recurrente *long short-term memory* (LSTM) que tiene como objetivo clasificar noticias escritas en positivas o negativas y luego realizar un experimento para determinar si el estado de ánimo de las personas realmente mejora.

Para ello se realizará una serie de tareas, las cuales resumidas son las siguientes:

- a) Recolectar una base de datos de noticias escritas en español.
- b) Clasificar estas noticias utilizando el servicio de Google AutoML, esto sirve para el entrenamiento.
- c) Transformar las noticias en vectores para que sirvan de *input* para el modelo.
- d) Formular e implementar el modelo de redes neuronales *long short-term memory*.
- e) Realizar la validación del modelo.
- f) Realizar el experimento para determinar el estado de ánimo de las personas luego de leer las noticias.

2. METODOLOGÍA

2.1 Recolección y preparación de noticias

Se utilizó una librería del lenguaje de programación Python llamada BeautifulSoup para poder realizar un web *scraping* de diferentes páginas web de noticias del Perú (*RPP, El Comercio, La República y Exitosa*)

Se escogió el título y el cuerpo de la noticia como *input* para el modelo, y se seleccionaron noticias entre el 12 de agosto del 2018 al 11 de septiembre del 2018 (Easton y McColl, 2007). Luego fueron almacenadas en un formato separado por comas (csv).

Para clasificar las noticias en negativas o positivas, primero fueron traducidas al inglés y luego se usó el servicio AutoML de Google para clasificarlas. Las noticias fueron traducidas al inglés debido a que AutoML de Google no puede determinar el sentimiento de textos en español.

En el siguiente paso, se empezó a realizar un proceso de muestreo aleatorio para balancear la proporción de noticias en 1:1, esto se realizó para que no exista un bias a la hora de entrenar el modelo. Al concluir este proceso se obtuvo una base de datos de noticias, de 20 000 noticias (10 000 noticias negativas y 10 000 noticias positivas) (Trochim, 2007).

Tabla 1
Distribución de las noticias

Base de datos de noticias		
Fuente	Positivas	Negativas
<i>El Comercio</i>	2563	1709
<i>La República</i>	2896	2108
<i>RPP</i>	2563	3156
<i>Exitosa</i>	1978	3027
	10 000	10 000

Elaboración propia

Luego, se realizó un proceso de *encoding* y tokenización (Famili, Shen, Weber y Simoudis, 1997). De esta forma se transformarán las noticias en vectores de números. También se creó automáticamente un diccionario de palabras en el cual se identifican las palabras con un valor numérico, generado después del *encoding*.

En la tarea de tokenización se realizó la eliminación de *stop-words*, la eliminación de caracteres especiales y signos de puntuación (Klevecka y Lelis, 2008).

2.2 Desarrollo del modelo

Para determinar la polaridad de las noticias se decidió usar una granularidad a nivel del documento, esto se debe a que una noticia posee un texto extenso, entonces es necesario poder obtener la polaridad de este en forma conjunta, pues a lo largo del escrito puede haber diferentes puntos de vista, tomar la granularidad a nivel de oración podría haber sido perjudicial para la tarea de clasificación.

El modelo de aprendizaje de máquinas seleccionado fue de redes neuronales recurrentes, en específico el tipo LSTM (Hochreiter y Schmidhuber, 1997), este tipo se escogió debido a la capacidad de poder “recordar” los elementos de un texto, esto es fundamental en el análisis de textos largos porque de esta manera se puede capturar su contexto completo.

Para normalizar los datos de entrada (*inputs*) del modelo se usó la técnica de *mean pooling* por dos motivos: 1) para controlar la cantidad de *features* que van a ser recibidas por la capa de regresión logística; 2) para extraer la información promedio de cada texto, tomando en cuenta toda la información del texto, esto quiere decir que todos los valores son usados para realizar un mapeado de *features*.

Para determinar la clasificación de las noticias, en la capa final de la red neuronal, se usó una regresión logística, debido a que la salida de la red LSTM normalizada por el *average pooling* brinda valores entre 0 y 1, lo cual demuestra una probabilidad de ser positivos o negativos, pero para ser aún más exactos, se decidió usar la regresión logística ya que sólo se cuenta con dos posibles clasificaciones, positivas o negativas.

Se decidió recolectar un *dataset* propio de noticias de medios de comunicación locales, debido a que no existe uno. Igualmente, se recurrió al uso de técnicas de limpiado de datos como la eliminación de signos de puntuación y de *stopwords*, ya que ambos no agregan valor para la tarea a realizar. Luego se realizó un muestreo aleatorio en la base de datos para tenerla en una relación de 1:1 con respecto a noticias positivas y negativas, esto con el fin de que no exista un bias a la hora de entrenar el modelo.

El modelo propuesto es una variación de la red neuronal LSTM. En esta variación, la activación que ocurre en la puerta de salida de la capa LSTM no depende de su estado C_t , esto permite realizar parte de las operaciones necesarias en paralelo, volviendo el modelo mucho más eficiente computacionalmente, menos tiempo en entrenar el modelo sin algún impacto notable en la efectividad del modelo (Gers, Schmidhuber y Cummins, 2000).

Las ecuaciones siguientes describen como se actualiza la capa LSTM en cada unidad de tiempo t , según el modelo propuesto.

- 1) Es el *input* (datos de entrada) de la capa LSTM en un tiempo t .
- 2) $W_f, W_c, W_o, U_i, U_f, U_c, U_o$ y V_o son matrices de *weights* (pesos del modelo).
- 3) b_i, b_f, b_c y b_o son vectores “bias”.

Primero, se calculan los valores para la puerta de entrada y el valor candidato para los estados de la capa LSTM en un tiempo t .

$$i_t = (W_i x_t + U_i h_{t-1} + b) \quad (1)$$

Donde i_t es el valor de la puerta de entrada en un tiempo t y σ es la función de activación.

$$\tilde{C}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c) \quad (2)$$

Donde \tilde{C}_t es el valor candidato para los estados de la red LSTM en un tiempo t .

Luego se calcula el valor de la función de activación f_t de las puertas de olvido en un tiempo t .

$$f_t = (W_f x_t + U_f h_{t-1} + b_f) \quad (3)$$

Luego de obtener el valor de la función de activación de la puerta de entrada, el valor de la función de activación de la puerta de salida y el valor candidato para los estados de la capa LSTM, se pasa a calcular el nuevo estado C_t de la capa LSTM en un tiempo t .

$$C_t = i_t * \tilde{C}_t + f_t * C_{t-1} \quad (4)$$

Donde \tilde{C}_t es el valor candidato para los estados de la red LSTM en un tiempo t .

Una vez calculado el nuevo estado C_t de la capa LSTM, se puede obtener el valor de sus puertas de salida y, como consecuencia, la salida final h_t de la capa LSTM.

$$o_t = (W_o x_t + U_o h_{t-1} + b_o) \quad (5)$$

Donde o_t es el valor de la puerta de salida de la red LSTM y σ es la función de activación.

$$h_t = o_t * \tanh(C_t) \quad (6)$$

La estructura final del modelo a usar consiste en una sola capa de redes neuronales LSTM, luego se implementará una capa de normalización (*mean pooling*), esto va a disminuir la variancia entre los valores, debido a que se va a tomar un promedio de la cantidad de datos para realizar un mapeo final de *features*. Después se implementará una capa de regresión logística para obtener una mayor eficacia a la hora de clasificar las noticias.

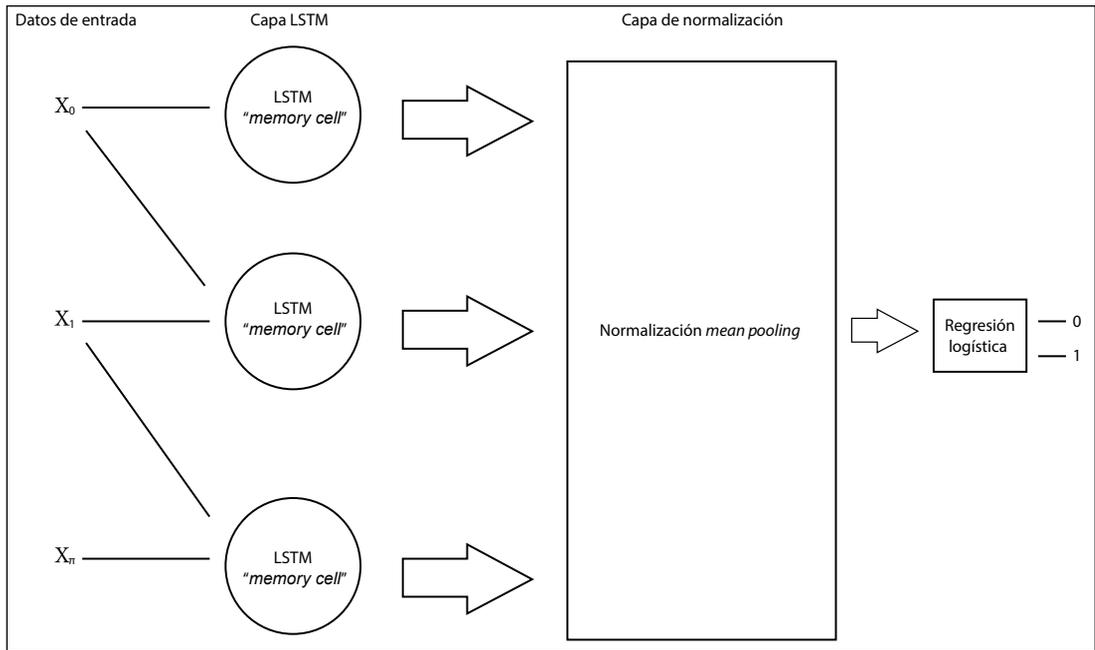


Figura 1. Estructura del modelo propuesto
Elaboración propia

2.3 Algoritmo de optimización

Para incrementar la eficiencia de la red neuronal propuesta, se ha decidido implementar un algoritmo llamado Adam, el cual sirve para escoger el mejor *learning rate* para que el entrenamiento de la red neuronal sea el más rápido y efectivo posible (Kingma *et al.*, 2014).

Adam es un algoritmo de *learning rate* adaptativo, lo que significa que calcula *learning rates* individuales para diferentes parámetros. Su nombre se deriva de la estimación del momento adaptativo, y la razón por la que se llama así es porque Adam usa estimaciones del primer y segundo momento del gradiente para adaptar la velocidad de aprendizaje para cada *weight* de la red neuronal.

Adam utiliza los gradientes elevados al cuadrado para escalar la velocidad de aprendizaje como el algoritmo RMSprop (Tieleman y Hinton, 2012) y aprovecha el impulso al usar la media móvil de los gradientes en lugar de solo la gradiente como el algoritmo *stochastic gradient descent*.

Para estimar los momentos, Adam utiliza promedios móviles exponenciales, calculados en el gradiente evaluado en un minilote actual:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \quad (7)$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \quad (8)$$

Donde m_t y v_t son los promedios móviles, g es la gradiente del minilote actual y los β son los hiperparámetros del algoritmo. Ambos tienen un valor *default* de 0,9 y 0,999, respectivamente

La fórmula para obtener el promedio móvil m_t también puede ser representada por la siguiente ecuación:

$$m_t = (1 - \beta_1) \sum_{i=0}^t \beta_1^{t-i} g_i \quad (9)$$

Después de esto se necesita realizar un paso llamado “*bias correction*”, esto quiere decir que se necesita corregir el valor inicial de m_t y v_t , para ello se realizan las siguientes transformaciones:

$$\hat{m}_t = 1 - \beta_1^t \quad (10)$$

$$\hat{v}_t = 1 - \beta_2^t \quad (11)$$

Para finalizar, se utilizan estos promedios móviles \hat{m}_t y \hat{v}_t para escalar la *learning rate* individualmente para cada parámetro. La forma en que se hace en Adam es simple, para realizar una actualización de *weights* hacemos lo siguiente:

$$W_t = W_{t-1} - \eta \left(\sqrt{\hat{v}_t} + \varepsilon \right) \quad (12)$$

Donde W_t es una matriz *weights* y la η es el tamaño del *step* (Zhang, Ma, Li y Wu, 2017).

2.4 Impacto de las noticias positivas

Para poder validar la hipótesis se decidió usar el test PANAS (Positive and Negative Affect Schedule).

Fue desarrollado en 1988 por Watson, Clark y Tellegen con el objetivo de medir de una forma más pura el efecto negativo o positivo de algún evento en específico (Watson, Clark y Tellegen, 1988). En un inicio el test PANAS contaba con 60 ítems (sentimientos) para determinar el sentimiento negativo o positivo, pero luego se realizó una reducción de la lista, basada en el coeficiente de importancia, y al final se obtuvieron 20 ítems, 10 para el sentimiento negativo y 10 para el sentimiento positivo (Watson, Clark y Tellegen, 1988).

Los resultados de este cuestionario fueron alentadores, porque fueron acertados, presentando los siguientes porcentajes: para la escala de sentimiento positivo, el coeficiente alfa de Cronbach (Cronbach, 1951) fue de 0,86 a 0,90; para la escala de sentimiento negativo, 0,84 a 0,87 (Watson, Clark y Tellegen, 1988).

3. RESULTADOS

3.1 Validación del modelo LSTM

Luego de haber implementado el modelo, se probó su eficacia utilizando el método de validación cruzada *k-folds* con un $k=10$ (Kohavi, 1995). Los resultados fueron los siguientes:

Tabla 2
Resultados de la validación cruzada k-folds

Validación cruzada <i>k-folds</i> (k=10)	
Folds (k)	Accuracy
1	86,56 %
2	85,36 %
3	86,85 %
4	89,37 %
5	88,29 %
6	85,67 %
7	89,95 %
8	91,51 %
9	88,30 %
10	87,92 %
Promedio: 87,98 %	

Elaboración propia

Para comparar la efectividad del modelo se implementaron otros dos métodos que también son usados en la actualidad:

- 1) Naive Bayes
- 2) Red neuronal recurrente (RNN)

El método usado para la validación, por consistencia, fue el de validación cruzada *k-folds* ($k=10$).

1) *Naive Bayes*

Tabla 3
Resultados de la validación cruzada k-folds

Validación cruzada <i>k-folds</i> (k=10)	
Folds (k)	Accuracy
1	79,89 %
2	83,25 %
3	79,28 %
4	81,59 %
5	83,17 %
6	79,03 %
7	79,98 %
8	80,55 %
9	81,71 %
10	82,99 %
Promedio: 81,14 %	

Elaboración propia

2) *Red neuronal recurrente RNN*

Tabla 4
Resultados de la validación cruzada k-folds

Validación cruzada <i>k-folds</i> (k=10)	
Folds (k)	Accuracy
1	82,79 %
2	81,90 %
3	83,58 %
4	81,27 %
5	82,90 %
6	83,51 %
7	84,95 %
8	80,11 %
9	82,76 %
10	80,56 %
Promedio: 82,43 %	

Elaboración propia

Luego de ver estos resultados, se puede afirmar que el modelo propuesto, una variación de la red neuronal recurrente LSTM tiene un mejor *accuracy* que el método de Naive Bayes y la red neuronal recurrente.

3.2 Procedimiento experimental

Test PANAS

Para validar la hipótesis se realizó un experimento basado en encuestas, con la participación de 520 personas de edades diferentes (entre 20-45 años) y de diferentes géneros (320 mujeres, 200 hombres). La encuesta se realizó antes y después de haber leído las noticias. La muestra se dividió de la siguiente forma:

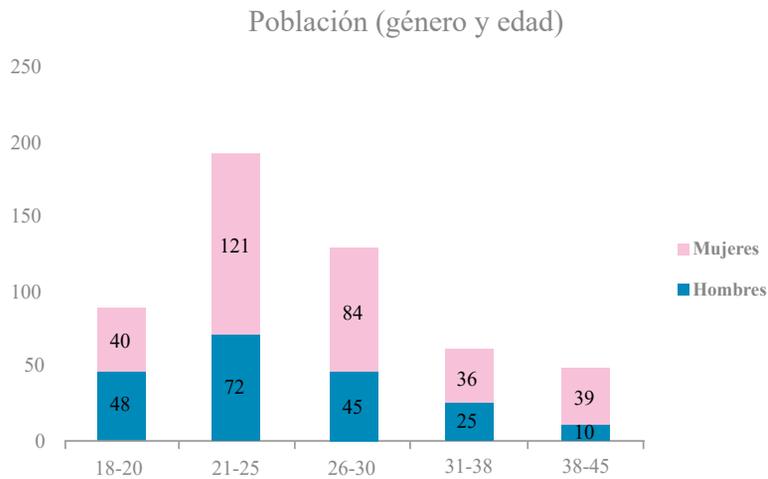


Figura 2. Perfil demográfico (género y edad)
Elaboración propia

Después de haber realizado las encuestas, los resultados fueron los siguientes:

Tabla 5
Resultados del test PANAS, antes y después de haber leído las noticias positivas

		Resultados Test PANAS			
		Positivo		Negativo	
		Antes	Después	Antes	Después
Hombres	18-20	25	39	30	16
	21-25	19	40	38	18
	26-30	15	38	42	20
	31-38	12	38	40	16
	38-45	14	40	41	17
Mujeres	18-20	27	42	25	12
	21-25	20	45	31	14
	26-30	12	37	38	17
	31-38	15	38	40	10
	38-45	17	42	38	19

Elaboración propia

4. CONCLUSIONES

Luego de haber realizado este reporte, implementando tanto el modelo de LSTM, para poder clasificar las noticias escritas en español en positivas o negativas, como el test de PANAS, con una población variada para poder determinar el efecto de las noticias positivas en ellas, se pueden concluir dos grandes puntos:

- a) El modelo propuesto basado en las redes neuronales LSTM cuenta con un *accuracy* del 87,98 %, esto es bueno comparado con el método de Naive Bayes y la red neuronal recurrente.
- b) Los resultados del test de PANAS nos demuestran que todas las personas que han leído las noticias positivas han mejorado su estado de ánimo.

5. TRABAJOS FUTUROS

Con respecto a la primera parte de este artículo, existen otras técnicas que pueden ser utilizadas a cambio de las propuestas en este artículo, como *max-pooling* o una versión del algoritmo Adam (NDAAdam) que resuelve alguno de sus errores (Zhang *et al.*, 2017).

También existen otros métodos, en especial el enfoque basado en la granularidad de aspecto, en contraste con la granularidad de texto, usada en este artículo, tiene un gran campo de investigación abierto, ya que usando este enfoque se pueden reconocer diferentes puntos de vista que existen en un texto, así como también el sarcasmo.

Por otro lado, las técnicas basadas en aprendizaje de máquinas no supervisadas podrían representar el futuro en este campo de investigación, esto se debe a la inmensa cantidad de datos que se generan diariamente y que en su mayoría no se encuentran clasificados.

REFERENCIAS

- Bastien, F., Lamblin, P., Pascanu, R., Bergstra, J., Goodfellow, I., Bergeron, A., Bouchard, N., Warde-Farley, D., y Bengio, Y. (2012). Theano: new features and speed improvements. *Deep Learning Workshop, NIPS 2012*.
- Bergstra, J., Breuleux, O., Bastien, F., Lamblin, P., Pascanu, R., Desjardins, G., Turian, J. P., Warde-Farley, D., y Bengio, Y. (2010). Theano: a CPU and GPU math expression compiler in Python. *Proceedings of the Python for Scientific Computing Conference (SciPy 2010)*.
- Cronbach, L. J. (1951). Coefficient alpha and the internal structure of tests. *Psychometrika*, 16(3), 297-334. doi:10.1007/BF02310555.
- Easton, V. J., y McColl, J. H. (2007). Statistics glossary: Sampling. Recuperado de <http://www.stats.gla.ac.uk/steps/glossary/sampling.html>
- Famili, A., Shen, W.-M., Weber, R., y Simoudis, E. (1997). Data preprocessing and intelligent data analysis. *Intelligent Data Analysis*, 1(1), 3-23.
- Fredrickson, B. L. (2001). The role of positive emotions in positive psychology: The broaden-and-build theory of positive emotions. *American Psychologist*, 56(3), 218-226.
- Gers, F., Schraudolph, N. N., y Schmidhuber, J. (2002). Learning precise timing with LSTM recurrent networks. *Journal of Machine Learning Research*, 3(1), 115-143.
- Gers, F. A., Schmidhuber, J., y Cummins, F. A. (2000). Learning to forget: Continual prediction with LSTM. *Neural Computation*, 12(10), 2451-2471.
- Hatemi, P. K., Funk, C. L., Medland, S. E., Maes, H. M., Silberg, J. L., Martin, N. G., y Eaves, L. J. (2009). Genetic and Environmental Transmission of Political Attitudes Over a Life Time. *Journal of Politics*, 71(3), 1141-1156.
- Hochreiter, S., y Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735-1780.

- Kingma, D. P., y Ba, J. (2015). Adam: A method for stochastic optimization. CoRR, abs/1412.6980.
- Klevecka, I., y Lelis, J. (2008). Pre-processing of input data of neural networks: The case of forecasting telecommunication network traffic. *Teletronikk: Telecommunications Forecasting (Special issue in co-operation with International Institute of Forecasters)*, 104(3/4): 168-178.
- Johnston, W. M., y Davey, G. C. (1997). The psychological impact of negative TV news bulletins: The catastrophizing of personal worries. *British Journal of Psychology*, 88, 85-91. doi:10.1111/j.20448295.1997.tb02622.x
- Tieleman, T., y Hinton, G. (2012). Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. COURSERA: *Neural Networks for Machine Learning*, 4(2), 26-31,
- Trochim, W. K. (2007). Sampling. Research methods knowledge base. Recuperado de <http://www.socialresearchmethods.net/kb/sampling.php>
- Watson, D., Clark, L. A., y Tellegen, A. (1988). Development and validation of brief measures of positive and neative affect: The PANAS Scales. *Journal of Personality and Social Psychology*, 54(6), 1063-1070. doi:10.1037/0022-3514.54.6.1063
- Zhang, Z., Ma, L., Li, Z., y Wu, C. (2018). Normalized direction-preserving Adam. arXiv:1709.04546v2