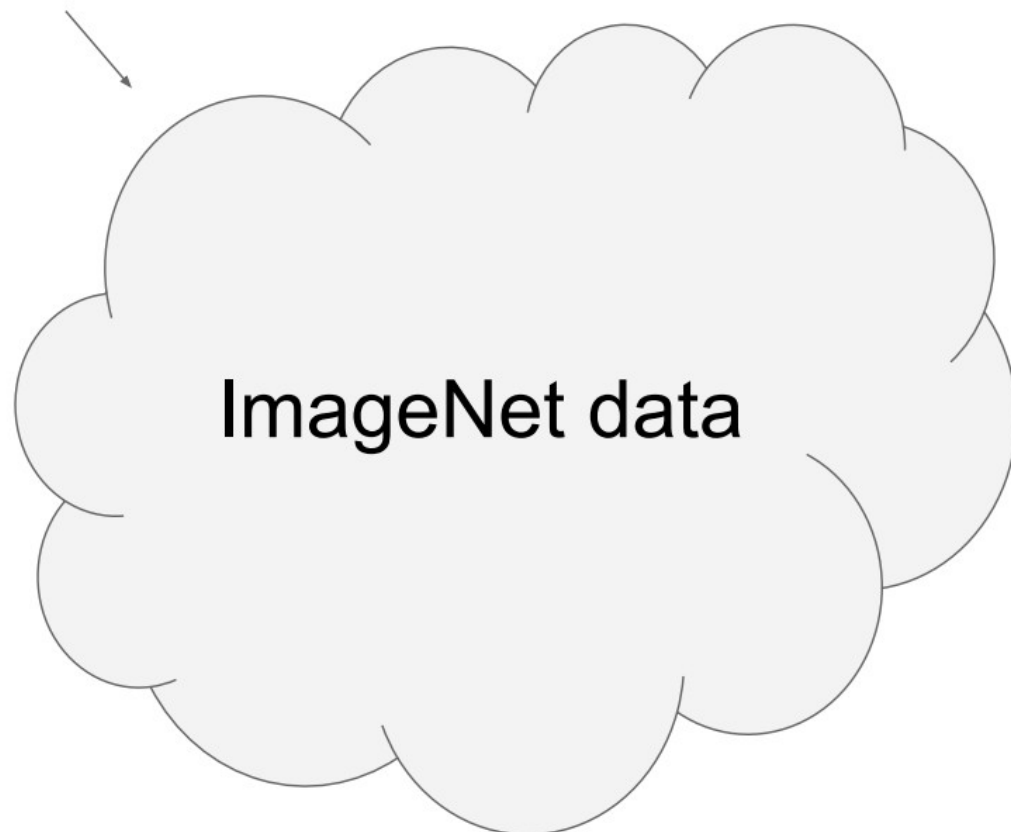


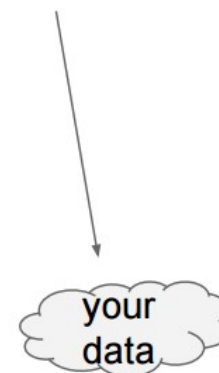
# Transfer Learning

# Pre entrenamiento sobre ImageNet

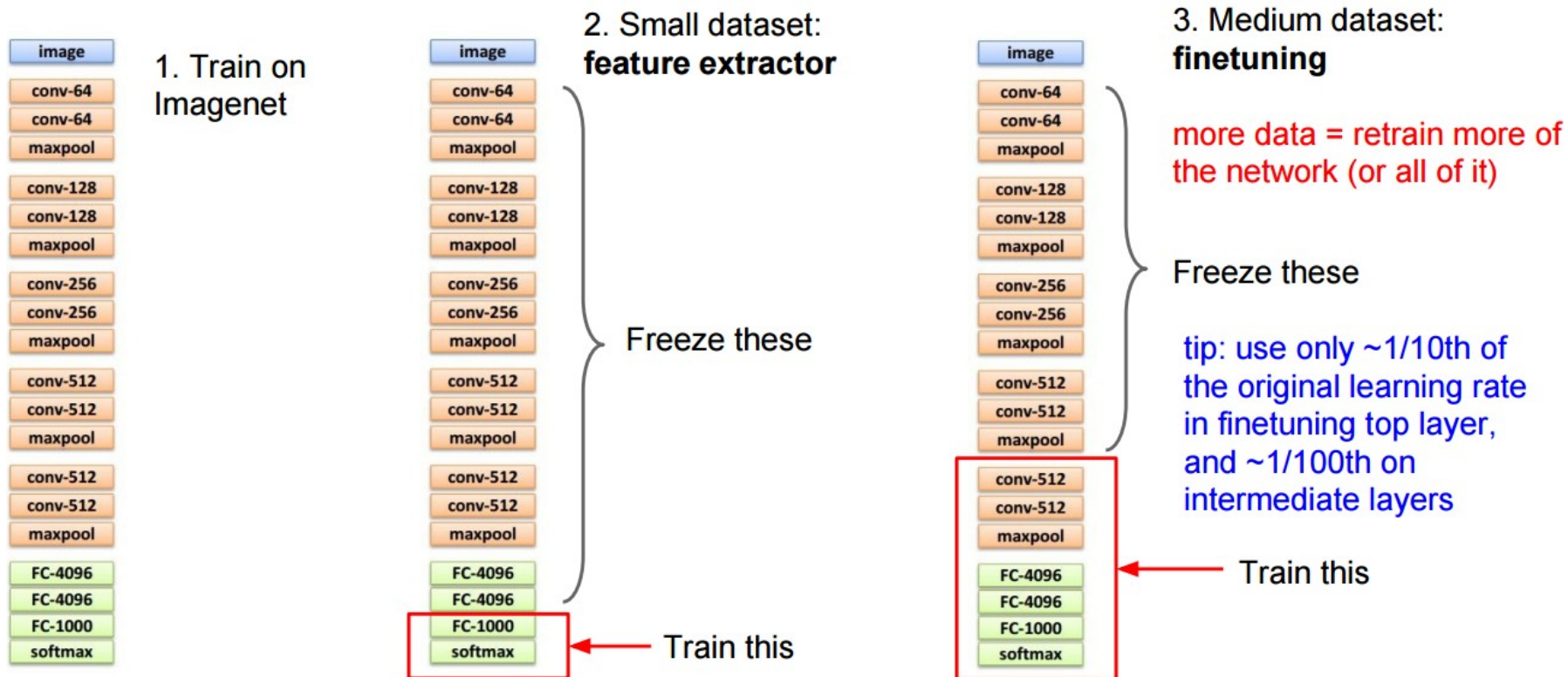
1. Train on ImageNet



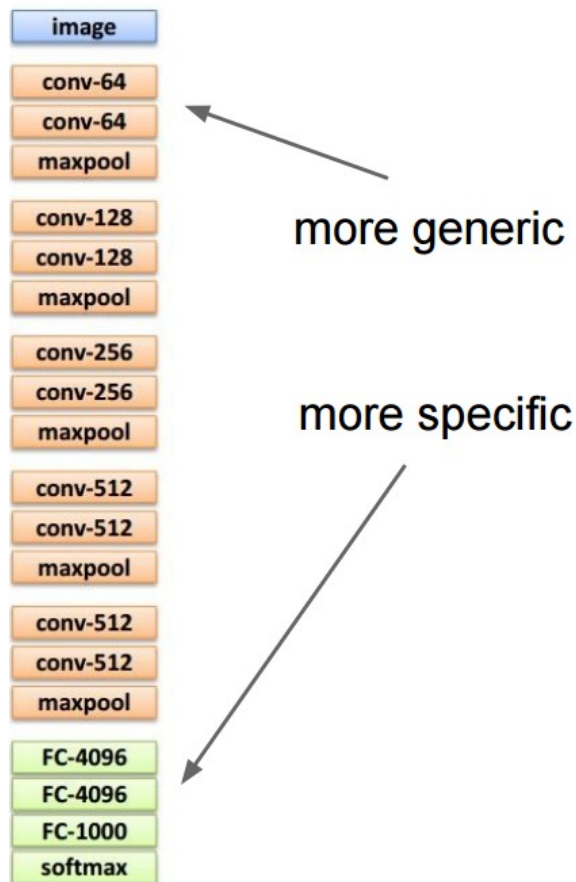
2. Finetune network on  
your own data



# Ejemplo con redes convolucionales



## Ejemplo con redes convolucionales



	<b>very similar dataset</b>	<b>very different dataset</b>
<b>very little data</b>	Use Linear Classifier on top layer	You're in trouble... Try linear classifier from different stages
<b>quite a lot of data</b>	Finetune a few layers	Finetune a larger number of layers

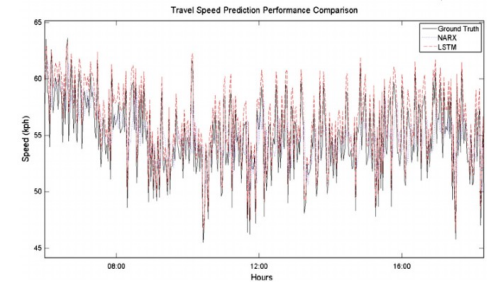
# Datos secuenciales

# Datos Secuenciales

la mirada panteísta donde un solo hombre inmortal es todos los hombres y a su vez ninguno. Y a partir de esta idea también puede irse, como luego veremos, que un solo texto también puede ser todos los textos. Según Borges este relato vendría a ser el bosquejo de una ética para inmortales y su tema "el efecto de la inmortalidad causaría en los hombres". Este efecto lo describe Borges a través del autor implícito del relato, el anticuario José Artaquilus, quien narra la vida del tribuno romano Marciano Rufo. Así podremos presenciar en este relato la voz de un hombre que fue todos y a la vez fue nadie, ya que fueron "las palabras de otros [...] la pobre limosna que le dejaron las horas y los siglos". El texto presente nos servirá para hacer una reflexión

text

# time series



# handwriting

Reçu de Monsieur L. Momméres  
la somme de deux cents francs, pour un  
mensuel de la pension alimentaire pour  
le mineur André Momméres.  
Ce dit mensuel commence le 1<sup>er</sup> mai  
prochain et finissant le 31 juillet prochain

speech



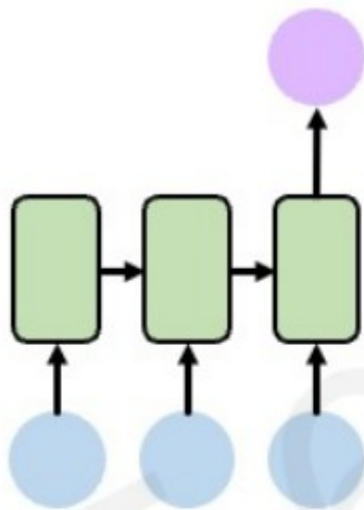
code

```
require_once('chodus/Share.php');
Database::set_defaults(
    array('user' => 'tumblr3', 'password' => 'm3Mph1C0Koh39AQD83TFh5BPLOm1R',
        'database' => 'tumblr3', 'write_log_tables' => '*');
    'extended_log' => (idate('G') == 17 && intval(idate('i')) == 56);
if ( __FILE__ == '/var/www/apps/tumblr/config/config.php' || __FILE__ == '/
define('ENVIRONMENT', 'production');
if (defined('DEFAULT_DATABASE')) define('DEFAULT_DATABASE', 'primary');
define('S3_BUCKET', 'data.tumblr.com');
define('ENABLE_PANTHER', true);
define('ENABLE_MEDIA CDN', true);
define('ASSETS_URL', (ENABLE_MEDIA_CDN && !isset($_SERVER['HTTPS']) &&
define('MEMCACHE_HOST', '10.252.0.68');
define('MEMCACHE_VERSION_HOST', '10.252.0.67');
define('VALIDATION_FAILURE_LOG', BASE_PATH . '/validate.log');
define('REDIRECT_403_LOG', BASE_PATH . '/403.log');
define('CONFIDENCE_PATH', (isset($_SERVER['HTTP_HOST']) && $_SERVER['HTTP
'ABQ7AAAAL7ad0Hn-kbP5qsR6c05axstxwJmV5p3uL0-WDRPjKQts4kls0oqky
'ABQ7AAAAL7ad0Hn-kbP5qsR6c05axstxwJmV5p3uL0-eJmgK0QtdPWk-8L515Pbk9
Database::add('primary', array('host' => '192.168.200.142');
Database::add('db-tumblelogs', array('host' => '192.168.200.103');
```

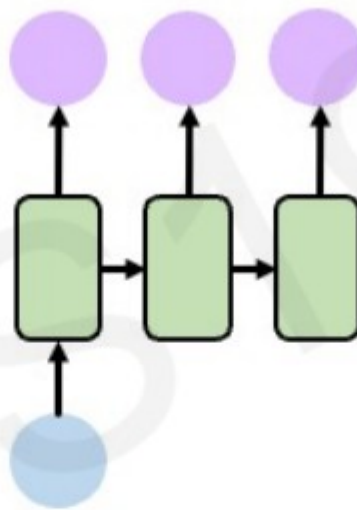
stock  
market



# Distintos problemas



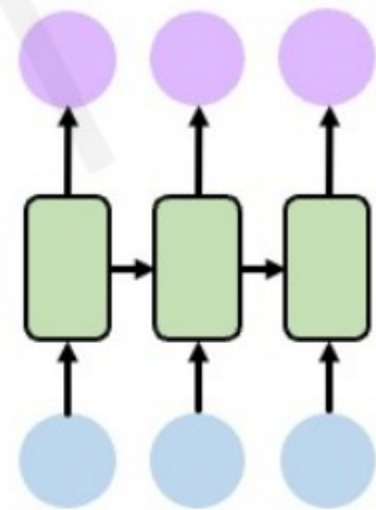
Many to One  
**Sentiment Classification**



One to Many  
**Image Captioning**



"A baseball player throws a ball."



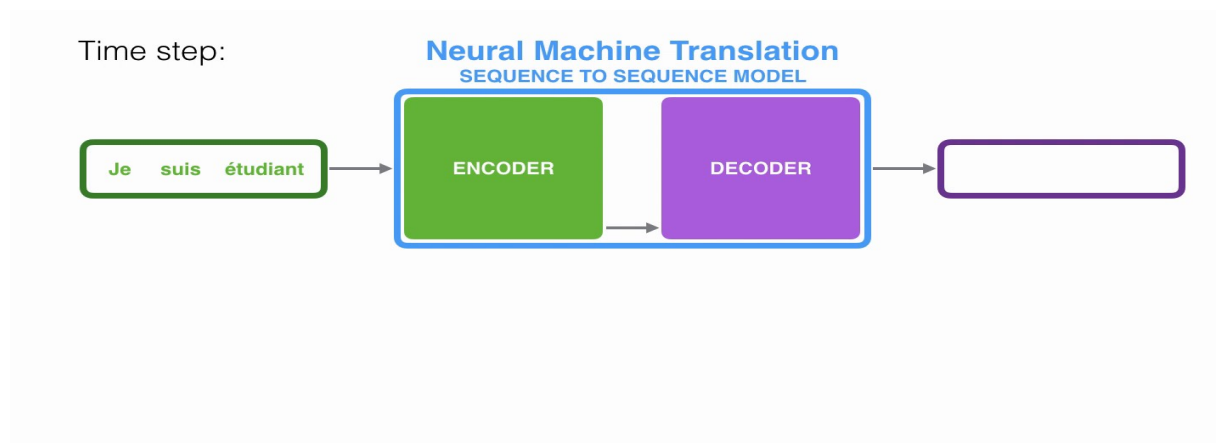
Many to Many  
**Machine Translation**



# Aprendiendo una secuencia

Para aprender una relación de secuencia-a-secuencia se usa un sistema en dos partes:

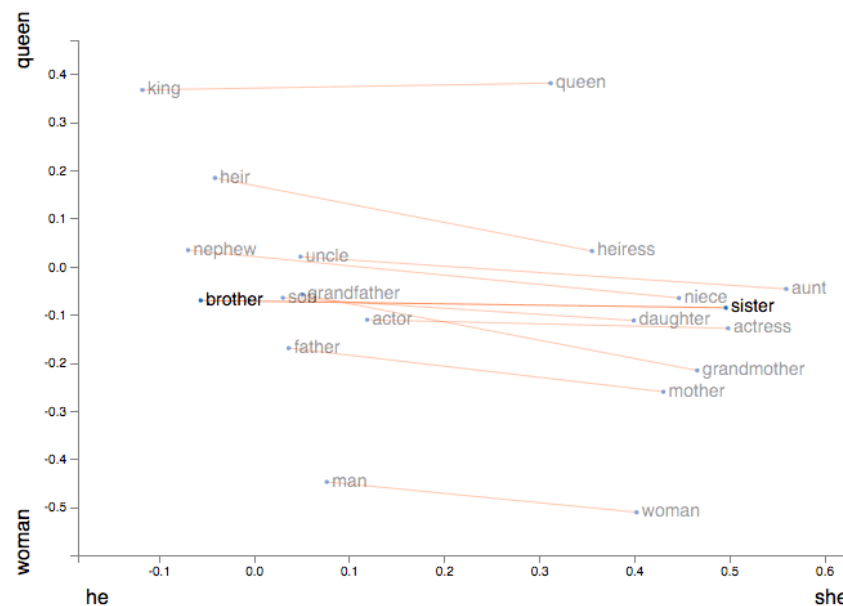
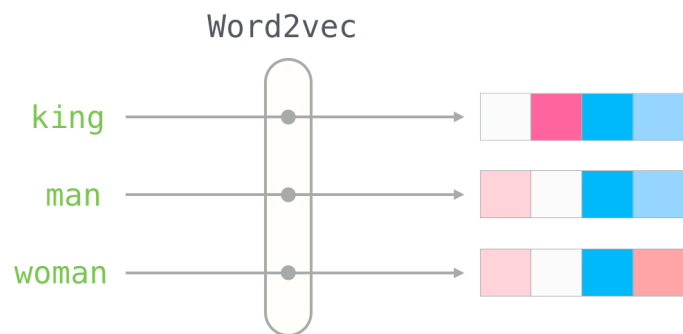
- Un encoder que traduce la secuencia a un estado interno que se aprende
- Un decoder que toma el estado interno y lo traduce a otra secuencia





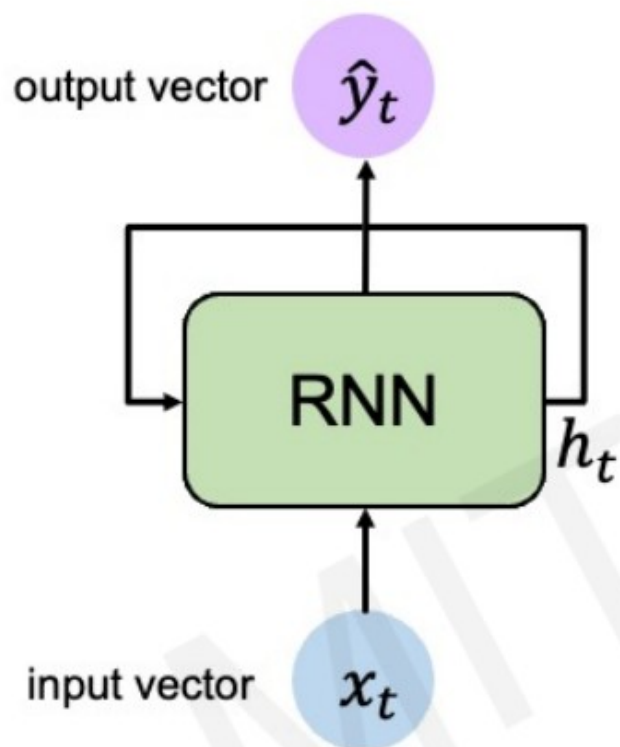
# Texto: Word embedding

El texto no se puede usar directamente como las imágenes. Se necesita un espacio vectorial donde palabras con sentido similar tengan una representación similar.



Learned: word2vec

# Recurrent Neural Networks



Output Vector

$$\hat{y}_t = \mathbf{W}_{hy}^T h_t$$

Update Hidden State

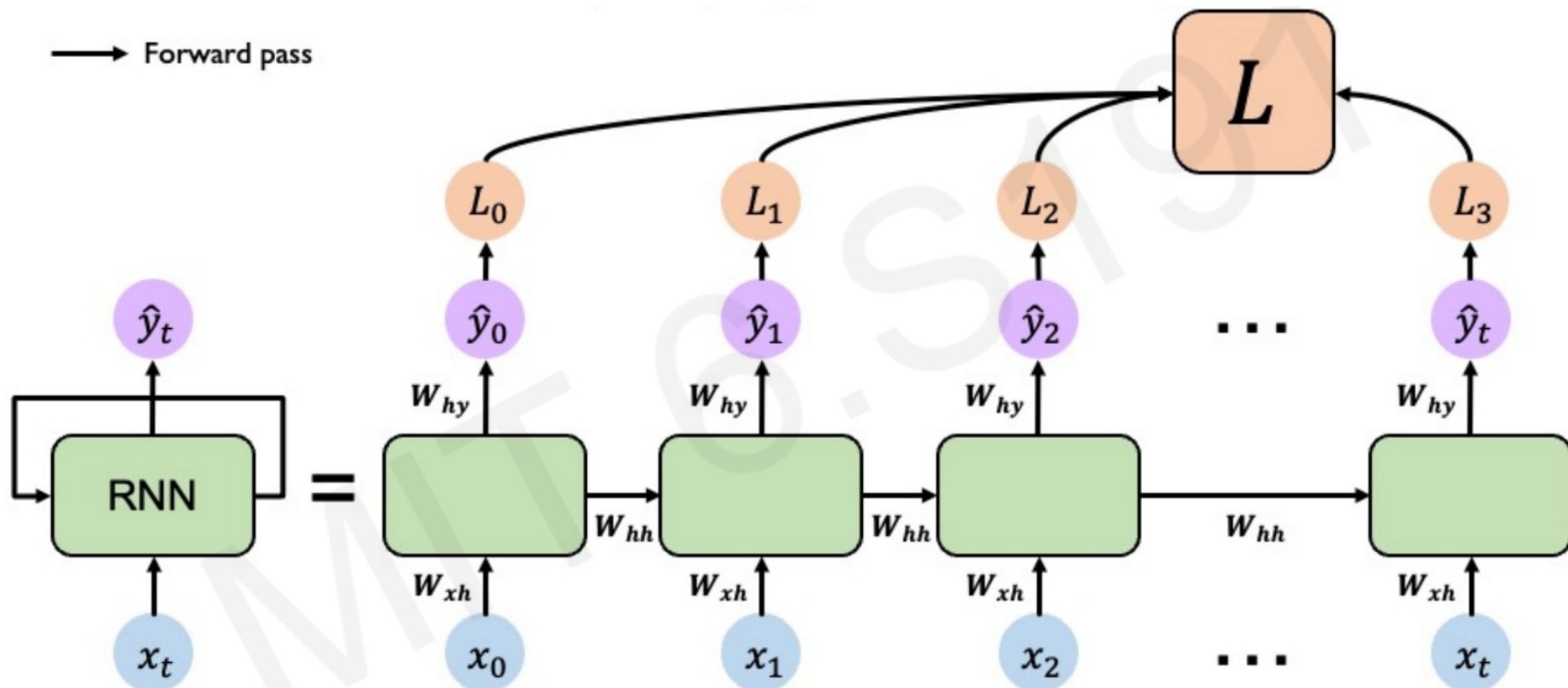
$$h_t = \tanh(\mathbf{W}_{hh}^T h_{t-1} + \mathbf{W}_{xh}^T x_t)$$

Input Vector

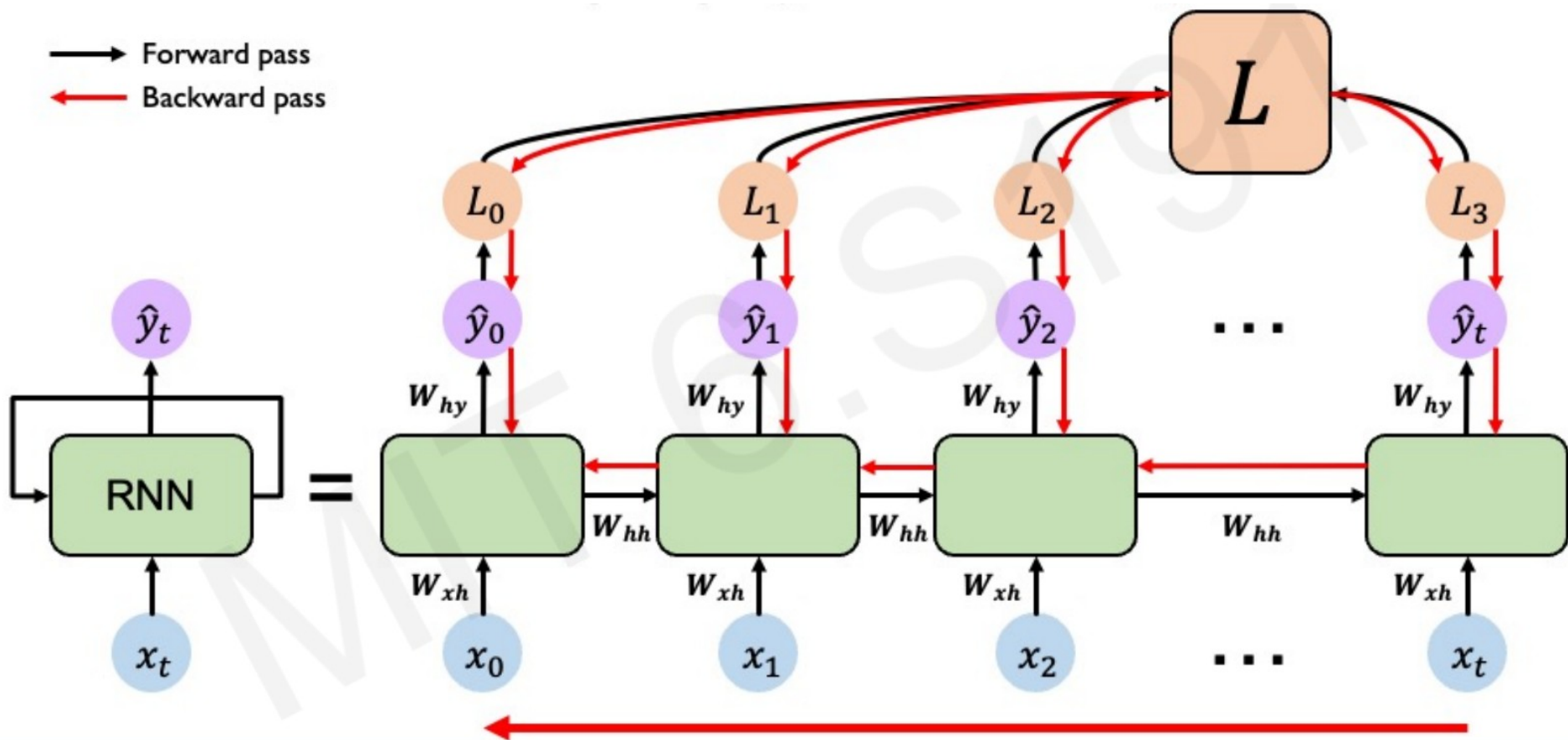
$$x_t$$

# Recurrent Neural Networks

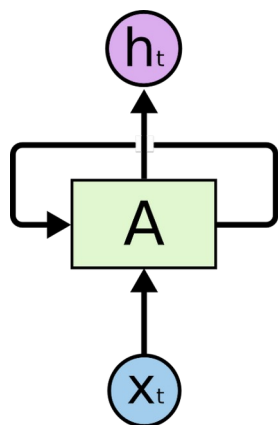
→ Forward pass



# RNNs: Backpropagation through time



# Recurrent Neural Networks



Neural Network  
with a loop

Problems:

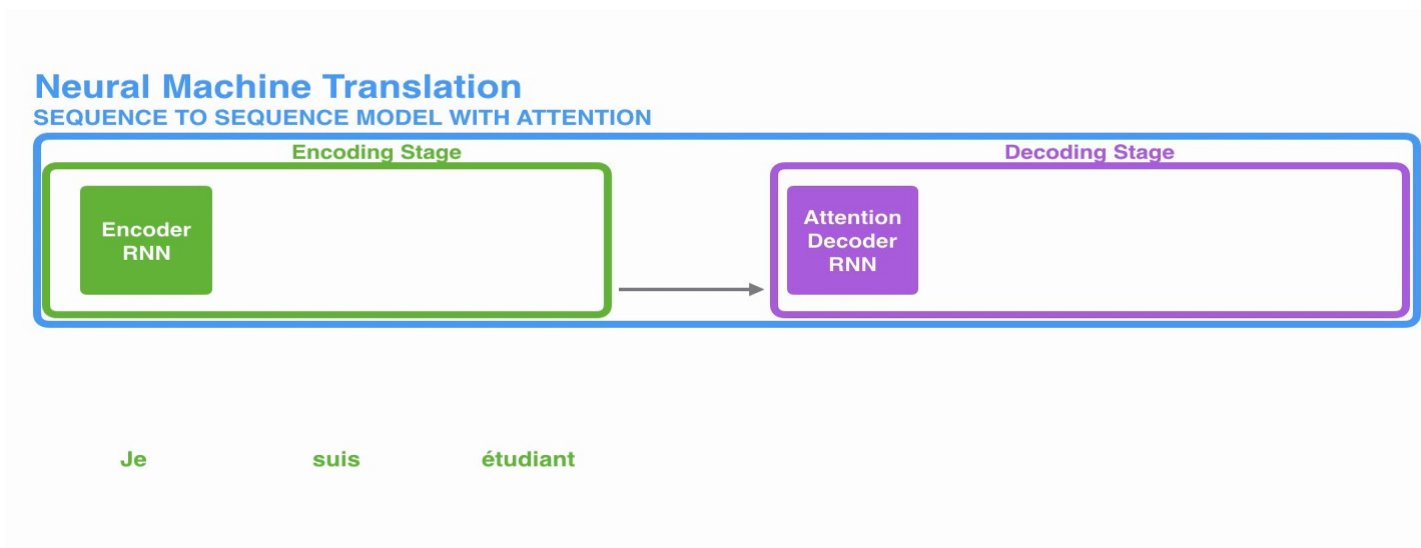
Long-term dependencies

Partial solutions (LSTM)

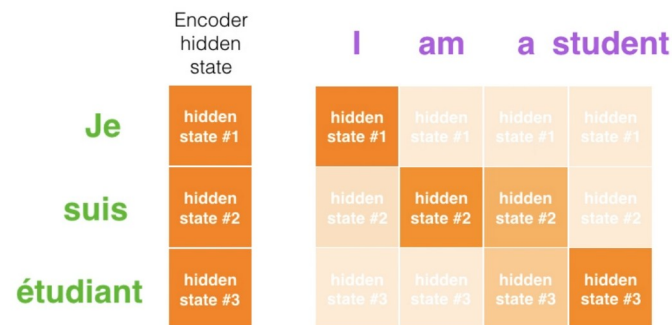
Sequential learning

Really slow for large problems

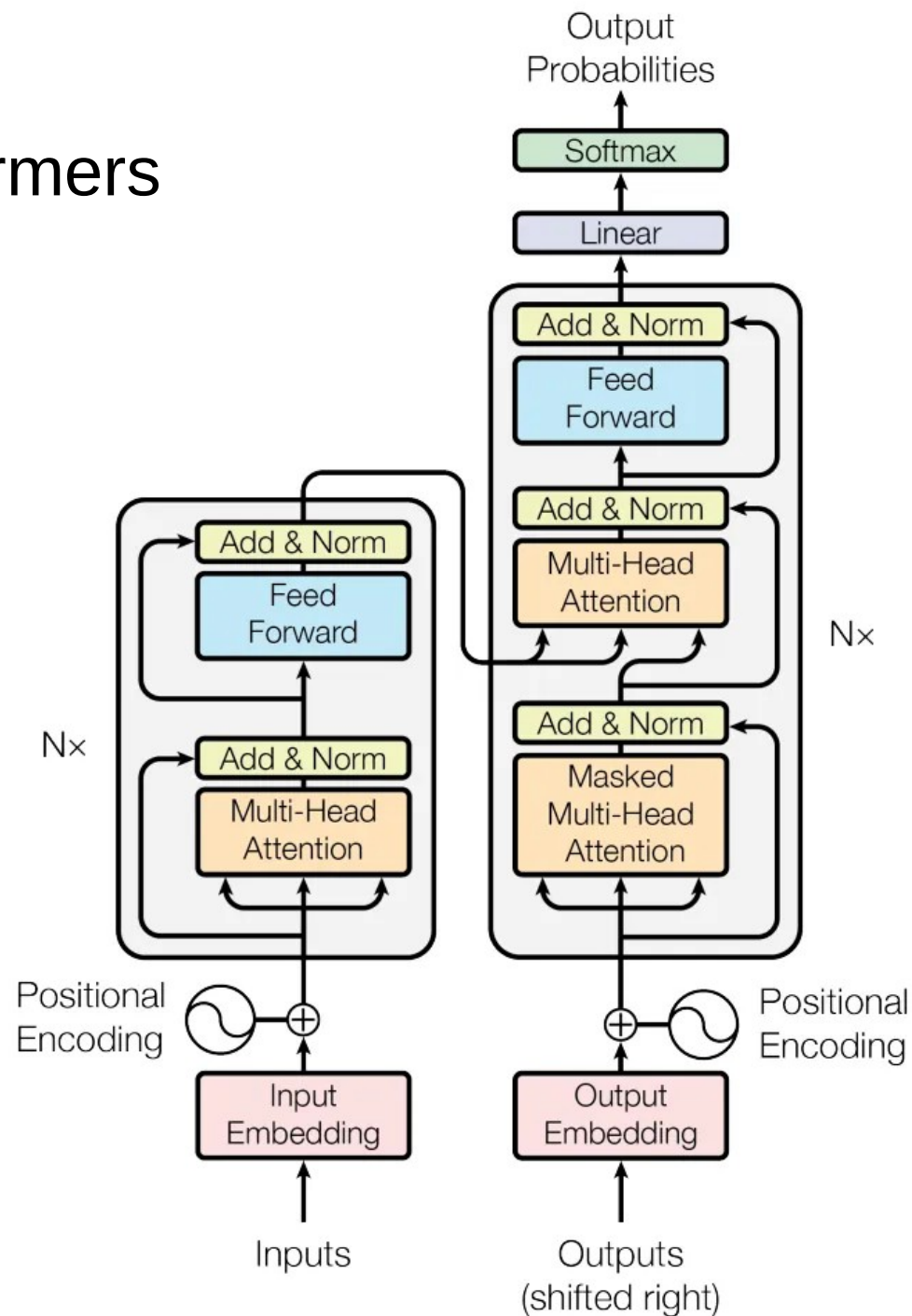
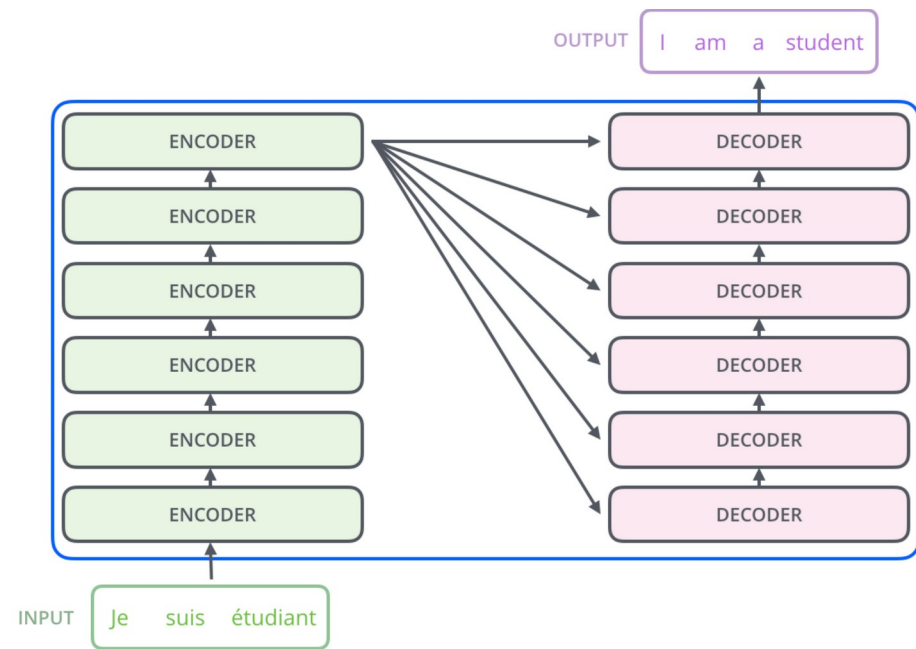
# Long term dependencies: Attention!



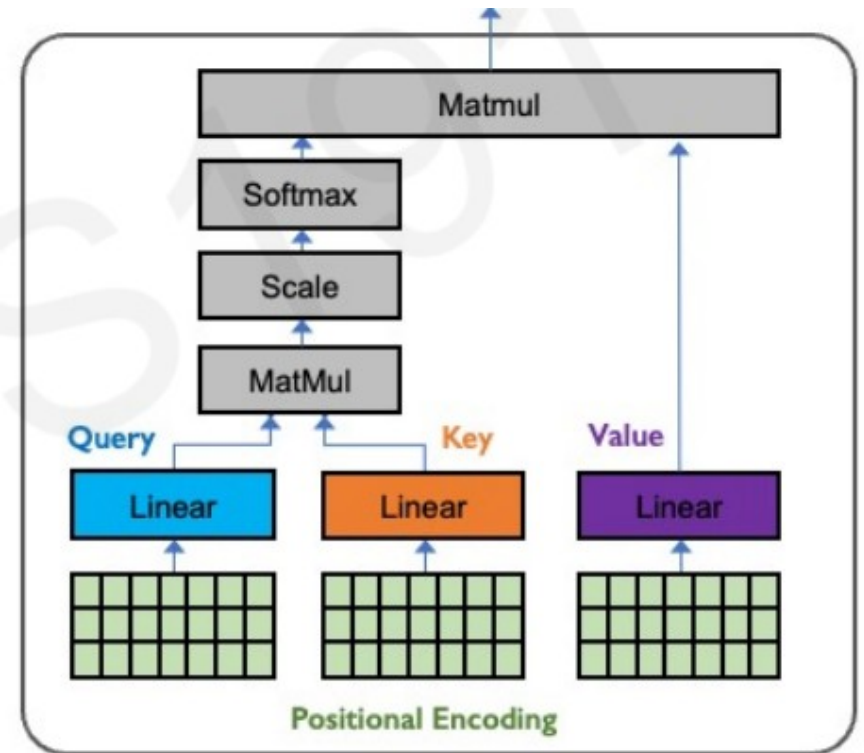
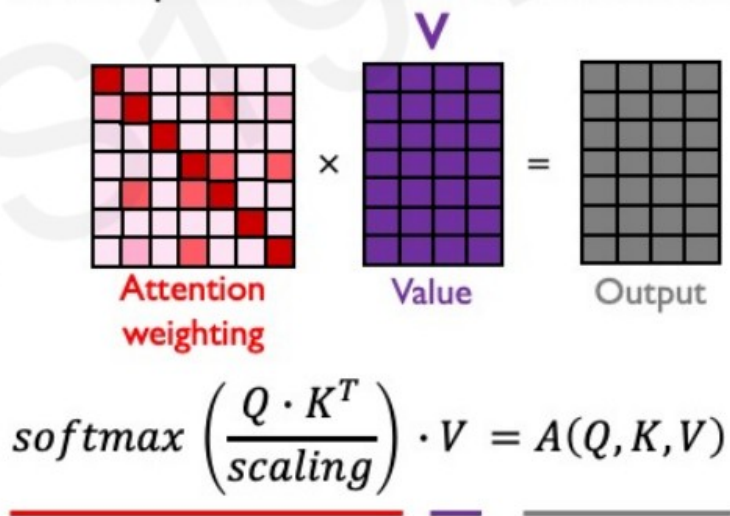
Permitimos al decoder ver todos los estados y elegir la información que usa en cada paso



# Parallel learning: Transformers



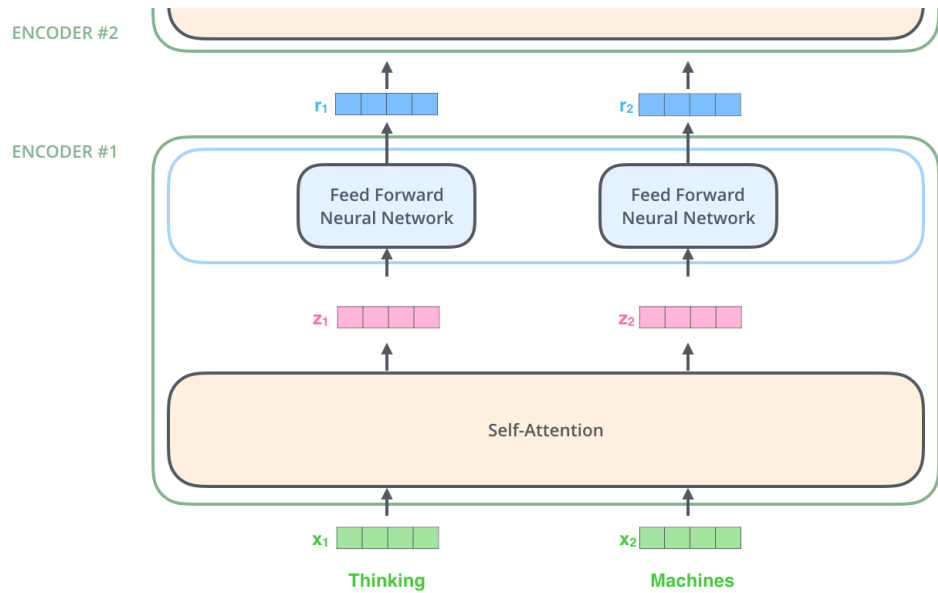
# Transformers: key points



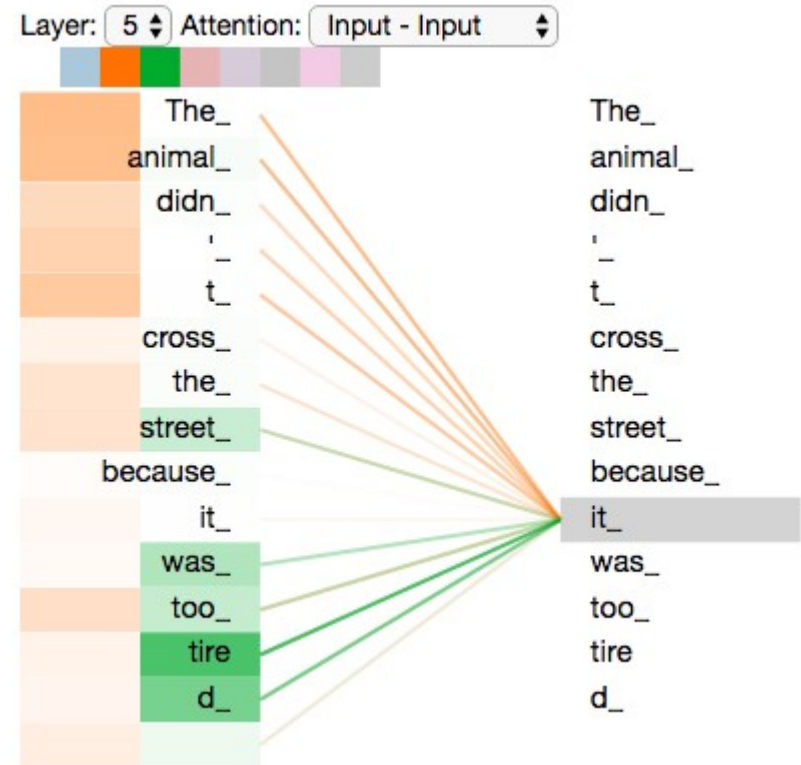
Attention → Query – Key - Value



# Transformers: key points



Multiple layers  
of encoders



Multiple  
attention heads

# Transformers: images

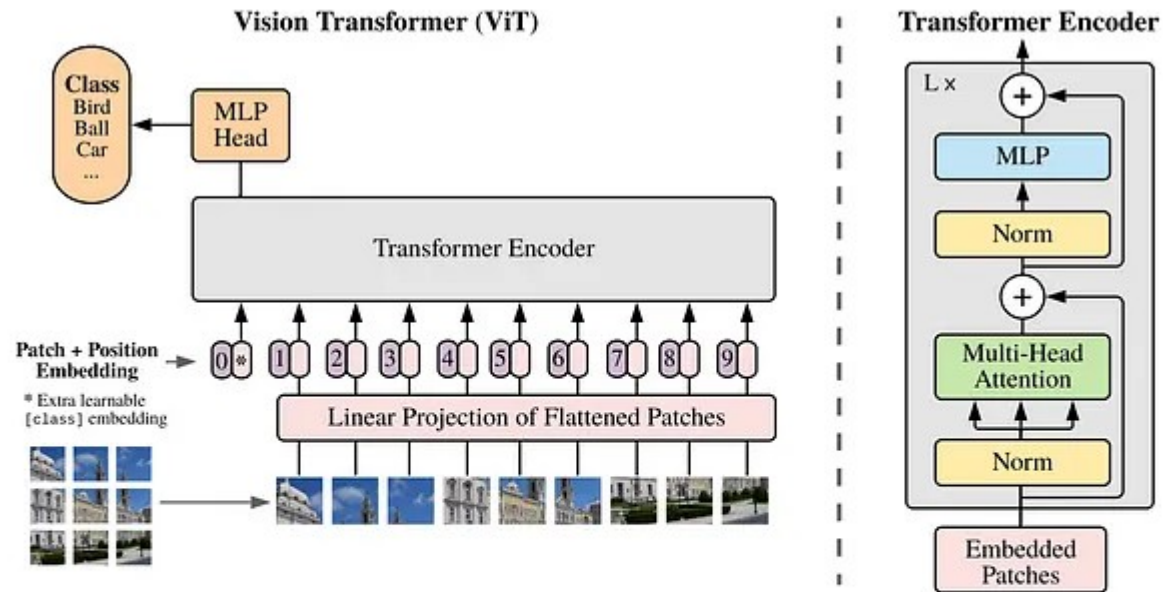


Figure 1: Model overview. We split an image into fixed-size patches, linearly embed each of them, add position embeddings, and feed the resulting sequence of vectors to a standard Transformer encoder. In order to perform classification, we use the standard approach of adding an extra learnable “classification token” to the sequence. The illustration of the Transformer encoder was inspired by Vaswani et al. (2017).