Dauphine | PSL ★
UNIVERSITÉ PARIS

**Data science project
Final presentation**

**Exploiting word embeddings
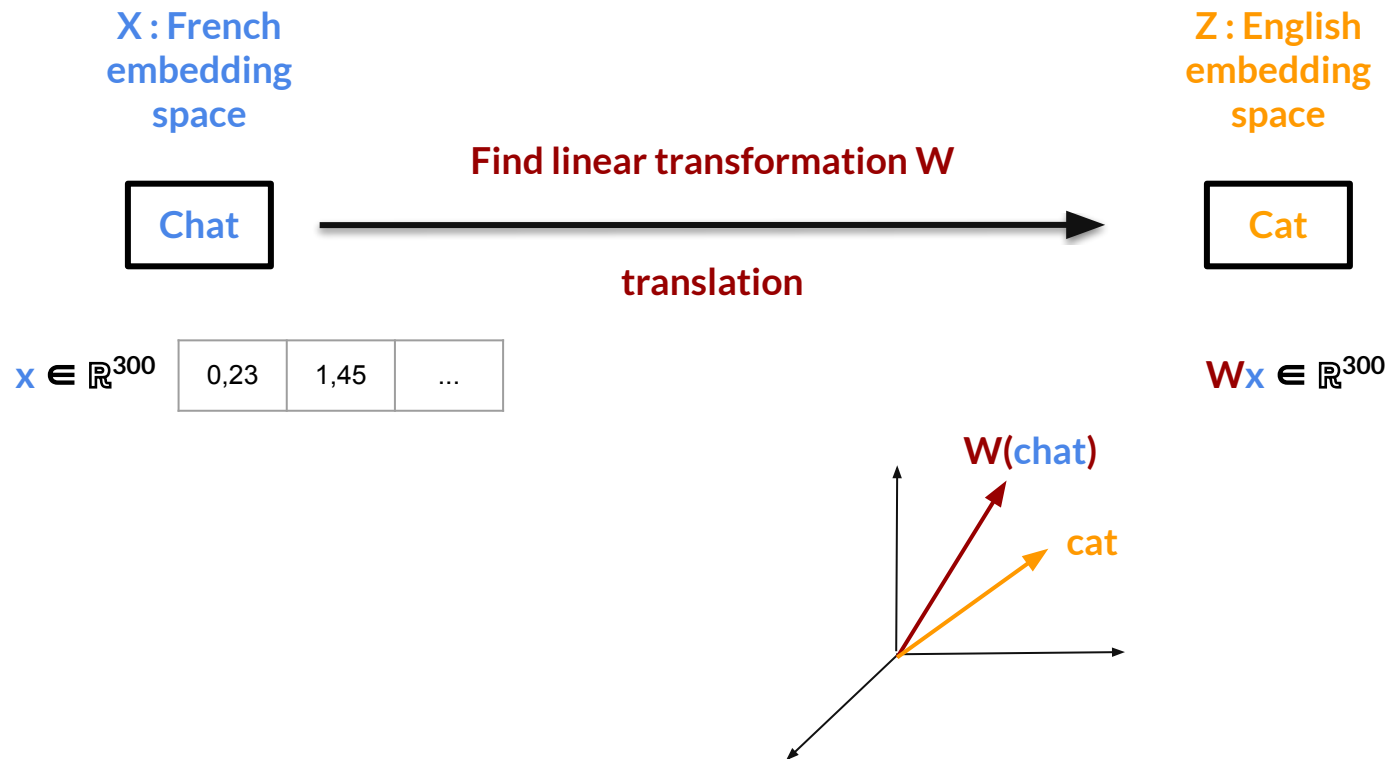for machine translation**

CôngMinh DINH
Louis Monier
Maxence Philbert
Vincent Gouteux

25/11/19

I - Supervised method
Recall

X : French
embedding
space

Z : English
embedding
space

Find linear transformation W

Chat

translation

Cat

$x \in \mathbb{R}^{300}$

0,23	1,45	...

$Wx \in \mathbb{R}^{300}$
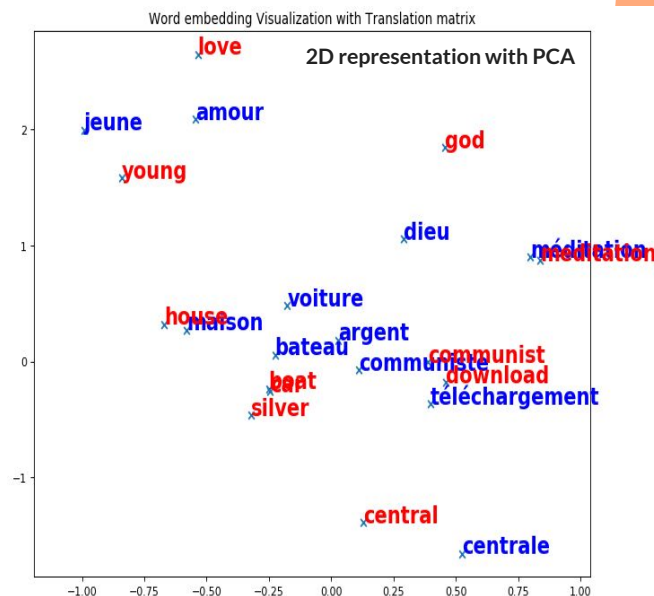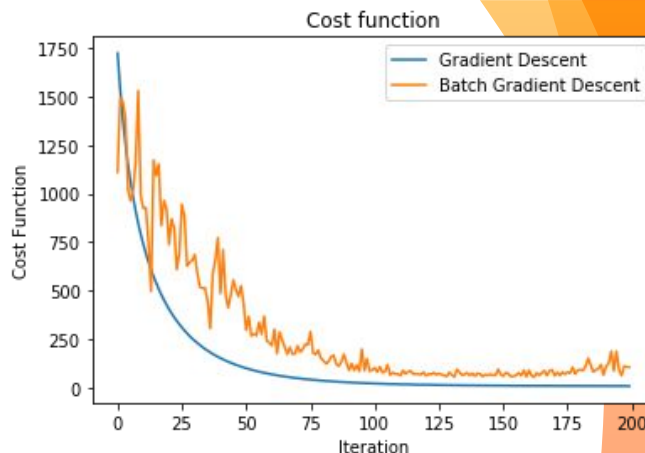
W(chat)

cat

I - Supervised method
Recall

○ Minimization of matrix translation
   (4 methods implemented : SD, SGD, MGD, analytical)

$$\min_{W} \sum_{i=1}^{n} \|Wx_i - z_i\|^2$$

○ Using cosine similarity to find the closest in the target language space

$$similarity : \frac{<x_i, z_i>}{\|x_i\| * \|z_i\|}$$

Cost function

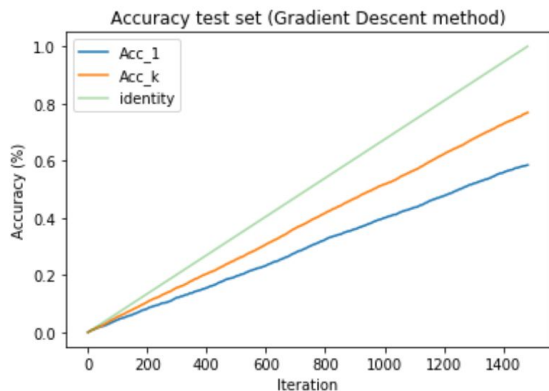
Word embedding Visualization with Translation matrix

2D representation with PCA

Supervised translator : results for different languages
Accuracy top @1/5 words
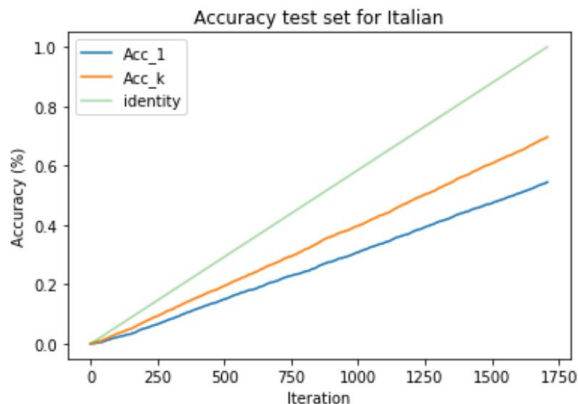
From French to English

Accuracy test set (Gradient Descent method)


```
Gradient descent method :
Final accuracy @1 = 58.46 %
Final accuracy @5 = 76.8 %

Analytical method :
Final accuracy @1 = 60.22 %
Final accuracy @5 = 77.14 %
```
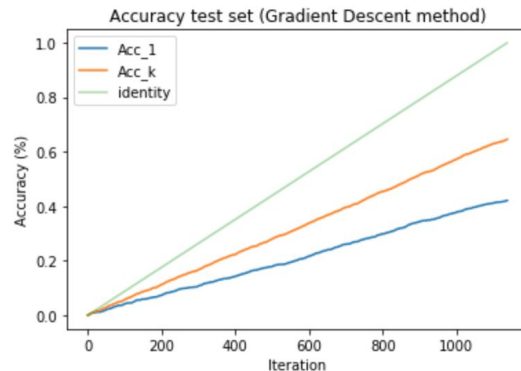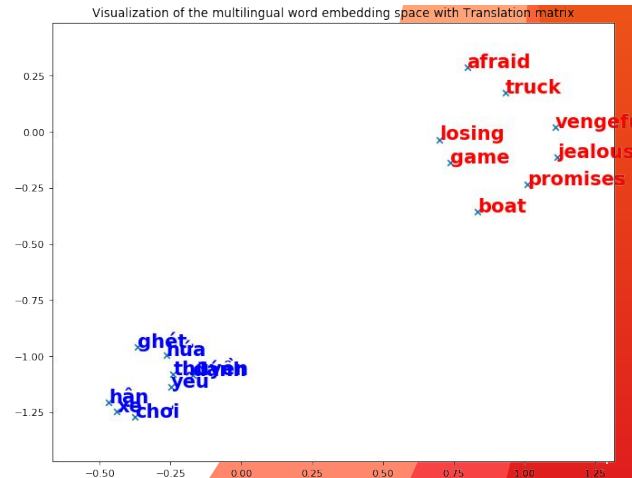
From Italian to English

Accuracy test set for Italian


```
Analytical method :
Final accuracy @1 = 54.45 %
Final accuracy @5 = 69.67 %
```

Good results with supervised translator !

### From Vietnamese to English

Accuracy test set (Gradient Descent method)


```
Gradient descent method :
Final accuracy @1 = 42.09 %
Final accuracy @5 = 64.59 %
```

Visualization of the multilingual word embedding space with Translation matrix

II - Build an efficient <u>unsupervised</u> translator
Generative Adversarial Network (GAN) method

➢ We don't know the traductions ;
  We just have collections of words in source & target spaces
➢ Two adversarial neural networks :
  ○ the *generator* generates new data instances ("fake" data)
  ○ the *discriminator* evaluates data for <u>authenticity</u>
➢ **<u>Objective :</u>** Learn the matrix W by training fake ($W*x$) and real (y) data
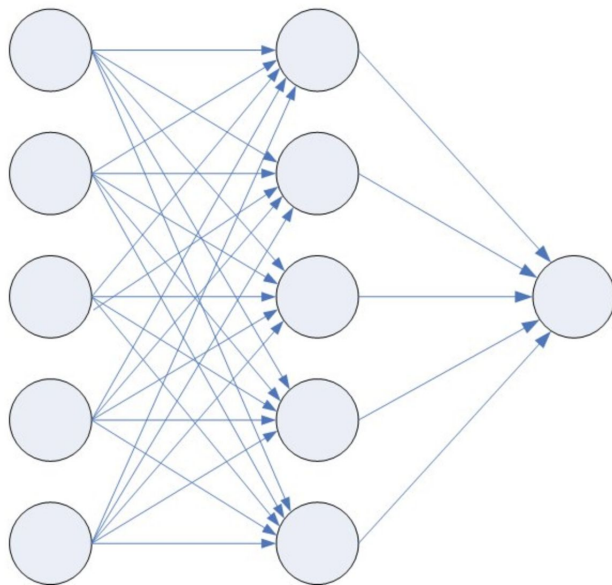➢ GAN's architecture :
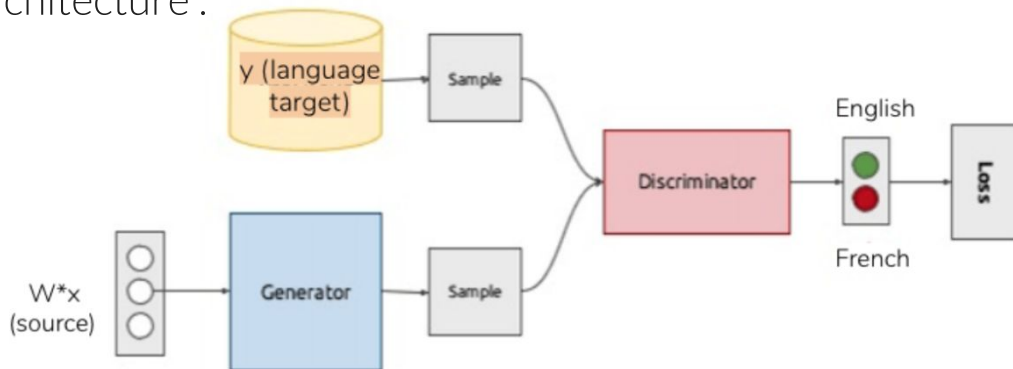
Unsupervised Translator : GAN
Discriminator

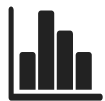**Neural network**

Inputs =

{W(chat);
House;
W(voiture);
W(soleil);
University;
W(étudiant); … }

Outputs =

{P(W(chat) ∈ source);
P(House ∈ source);
P(W(voiture) ∈ source);
P(W(soleil) ∈ source);
P(University ∈ source);
P(W(étudiant) ∈ source); … }

Unsupervised Translator : GAN
Discriminator

$$\mathcal{L}_D(\theta_D|W) = -\frac{1}{n}\sum_{i=1}^{n}\log P_{\theta_D}\left(\text{source} = 1\middle|Wx_i\right) - \frac{1}{m}\sum_{i=1}^{m}\log P_{\theta_D}\left(\text{source} = 0\middle|y_i\right).$$

◦   W matrix is fixed
◦   3 layers Neural Network takes an embedding as input and returns the probability that this embedding comes from source language

◦
◦   **Objective :** minimize the loss, adapt the weights of the network in order to recognize with high precision the language of the embedding

7

Unsupervised Translator : GAN
Generator

```
┌─────────┐                    ┌─────────┐
│    x    │ ─────────────────► │   Wx    │      "fake" data
└─────────┘                    └─────────┘
```

$$\mathcal{L}_W(W|\theta_D) = -\frac{1}{n}\sum_{i=1}^{n}\log P_{\theta_D}\left(\text{source}=0\big|Wx_i\right) - \frac{1}{m}\sum_{i=1}^{m}\log P_{\theta_D}\left(\text{source}=1\big|y_i\right).$$

◦ 1 layer "Neural Network" : takes an embedding x as input
  and returns the "translation" Wx

◦ **Objective :** Minimize the loss + adapt the weights of the network =
 weights of W in order to generate embeddings close to those in target space

Then, train by SGD with BCE loss function

Unsupervised Translator : GAN
Results

Tried to implement the GAN and played with all possible parameters :

- Nb of iterations
- SGD learning rate
- Nb of hidden layers discriminator
- Add smoothing
- Training spaces
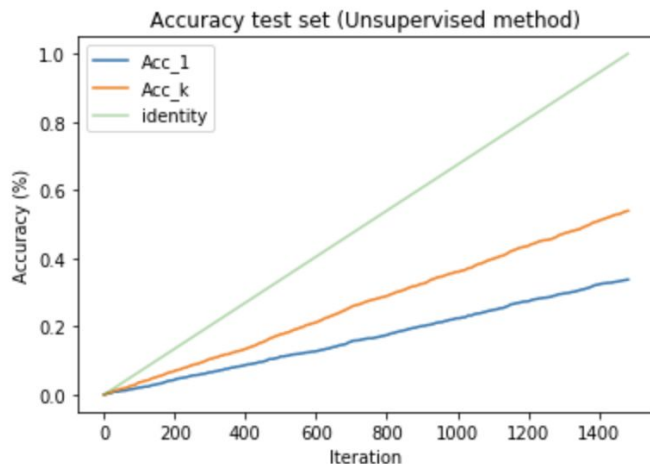- Nb of words in batch
- Initialisation of networks' weights

Generator and Discriminator Loss During Training
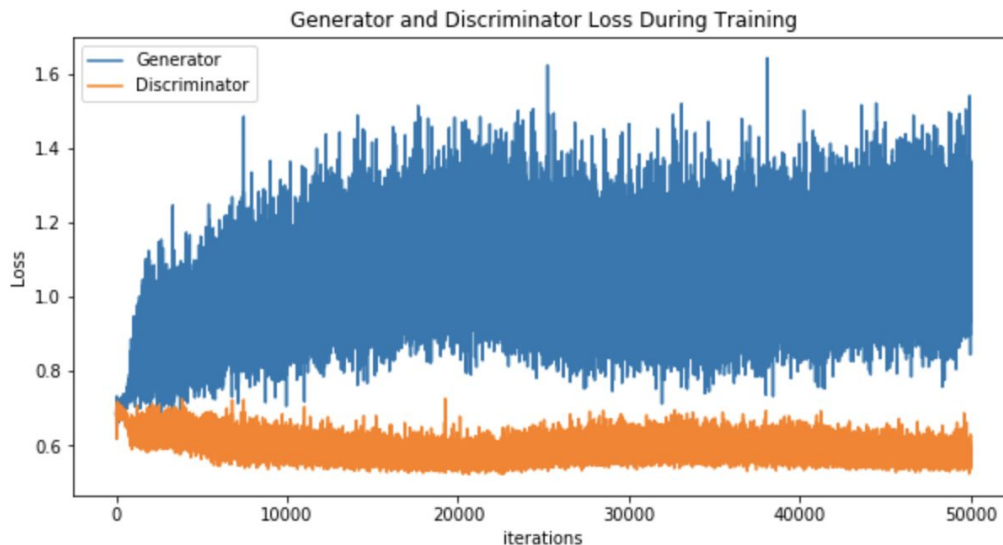
Need a large number of iteration to work (longer the training is, better we converge towards optimal W)

Unsupervised Translator : GAN
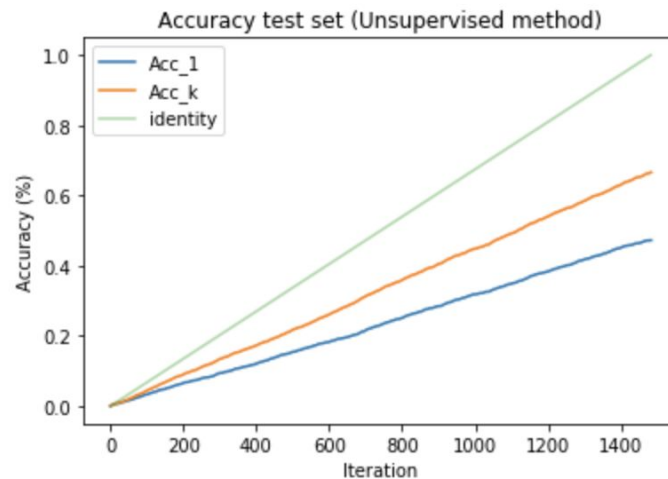## Results

30000 itérations training

50000 itérations training

Not bad !

```
Unsupervised method :
Final accuracy @1 = 33.72 %
Final accuracy @5 = 53.88 %
```

```
Unsupervised method :
Final accuracy @1 = 47.2 %
Final accuracy @5 = 66.55 %
```

```
Gradient descent method :
Final accuracy @1 = 58.46 %
Final accuracy @5 = 76.8 %

Analytical method :
Final accuracy @1 = 60.22 %
Final accuracy @5 = 77.14 %
```

Supervised Accuracy

Not implemented but we should have better results with Cross Domain Similarity Local Scaling measure

Unsupervised Translator : GAN
Results : examples of traductions

```
bateau --> traduction : ['boat', 'boats', 'steamer', 'capsized', 'sailboat']
--
maison --> traduction : ['house', 'cottage', 'townhouse', 'farmhouse', 'residence']
--
argent --> traduction : ['silver', 'gold', 'bronze', 'bullion', 'money']
--
ordinateur --> traduction : ['programmable', 'computer', 'computers', 'workstations', 'interface']
--
dieu --> traduction : ['god', 'divine', 'deity', 'gods', 'worshipping']
--
communiste --> traduction : ['communist', 'communists', 'comintern', 'leftist', 'socialist']
--
satan --> traduction : ['satan', 'sinners', 'evil', 'resurrection', 'antichrist']
--
amour --> traduction : ['love', 'lovers', 'lover', 'longing', 'sorrow']
```

Thanks!

Any questions?