PSL
UNIVERSITÉ PARIS

Word Embeddings and Translation

WE_HGA

Hamza Touzani, Alexis Hummel,
Guillaume Kunsch

Plan

I. Supervised setting
    A. “Classical” approach
        1. Theory
        2. Results
    B. Procrustes method
        a. Motivation
        b. Results
II. Unsupervised setting
    A. Unsupervised theory
    B. Results

Supervised Setting - Classical approach

Final objective

“Hello!” ⟶ “Bonjour!”

Words to Vectors

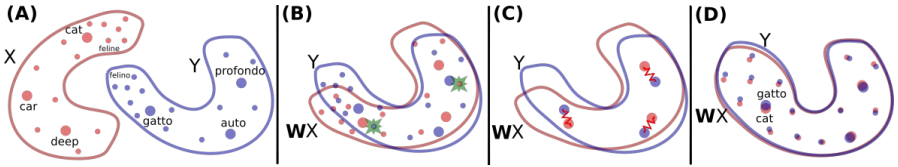
“Hello!” —embedding→ $x_i = \begin{bmatrix} 2{,}3 \\ 3{,}7 \\ \dots \\ 0{,}6 \end{bmatrix} \in \mathbb{R}^d$

Basic idea: similar words are closed in a sentence (Harris, 1954)

Knowing a dictionary of elements $\{x_i, y_i\}$

$$\min_W \sum_i \|Wx_i - y_i\|^2$$

Exploiting Similarities among Languages for Machine Translation, Mikolov et al., 2013



Word Translation Without Parallel Data, Conneau et al., 2017

Supervised Setting - Classical approach

Embedding

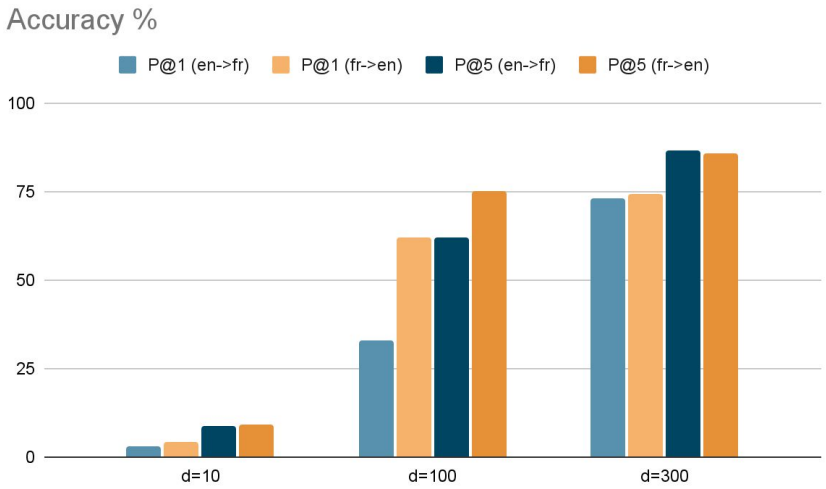
- FastText (from Meta)

Data

- dictionaries from MUSE (Meta)
- Train set : 10k words
- Test set: 1k words

Evaluation

- Cosine distance (not squared loss)
- kNN to find closest words



Supervised Setting - Procrustes method

$$W^\star = \underset{W \in M_d(\mathbb{R})}{\operatorname{argmin}} \|WX - Y\|_{\mathrm{F}}$$

$$W^\star = \underset{W \in O_d(\mathbb{R})}{\operatorname{argmin}} \|WX - Y\|_{\mathrm{F}} = UV^T, \text{with } U\Sigma V^T = \mathrm{SVD}(YX^T)$$

➡ Ensures that the monolingual quality of the embeddings is preserved

See : Chao Xing, Dong Wang, Chao Liu, and Yiye Lin. Normalized word embedding and orthogonal transform for bilingual word translation. Proceedings of NAACL, 2015

Supervised Setting - Procrustes method

	EN-FR P@1	EN-FR P@5	FR-EN P@1	FR-EN P@5	EN-TR P@1	EN-TR P@5	TR-EN P@1	TR-EN P@5
SGD	**73.4**	**86.4**	**75.4**	**85.8**	**75.2**	**86.0**	**83.0**	**93.2**
Procrustes alignement	**79.6**	**90.0**	**81.8**	**88.8**	**86.0**	**93.0**	**92.8**	**96.0**

*Word translation **P@1** and **P@5** for 2 language pairs (using d=300 **fastText** embeddings).*

*We consider **1000 test queries** with **10k target words** for each language pair.*

Generative Adversarial Networks

Two player game

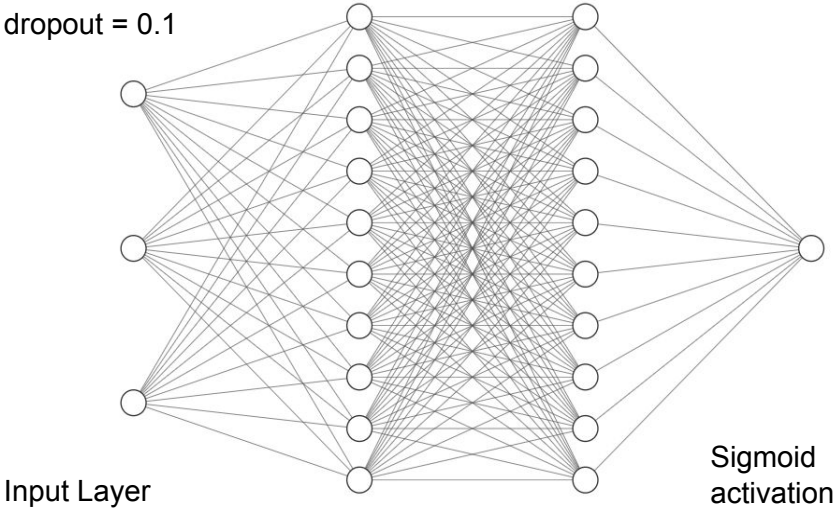
Mapping

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

Discriminator

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

Discriminator



2 Hidden Layers of size 1028

Generator

$$\begin{pmatrix} \ddots & & & & & \\ & \ddots & & & & \\ & & \ddots & & & \\ & & & \ddots & & \\ & & & & \ddots & \\ & & & & & \ddots \end{pmatrix}$$

Linear mapping W of size 300x300

Optimizer for discriminator/generator : SGD

What should we do next ?

Improve our training method for the GAN