# Cours

Charles Vin

Date

## 1 Généralité

**Métrique**

- Taux de reconnaissance : $\frac{bonnepred}{totalpred}$
- Précision pour une classe $c$ : $\frac{N^C_{correct}}{N^C_{predits}}$
- Rappel / Recall pour une classe $c$ : $\frac{N^C_{correct}}{N^C_{tot}}$
- F1 : need plus de vulgarisation
- ROC : Faux positif VS Vrais positif
- AUC :

**Problème d'équilibre des classes**

- Accuracy poubelle $\rightarrow$ Utiliser les autres metrics : AUC / ROC
- Ré-équilibrer le jeux de données : supprimer des données dans la classe majoritaire
- Fonction de coût : pénaliser plus les erreurs dans la classe minoritaire (cours 1 diapo 51)

**Problème de dimension** Ajouter un terme sur la fonction coût (ou vraisemblance) pour pénaliser le nombre (ou le poids) des coefficients utilisés pour la décision. (Cours 1 diapo 50)

**TF-IDF encoding** if word k is in most documents, it is probably useless $\rightarrow$ TF-Idf encoding : Donne plus de poids au keyword et un petit peu moins au stopword. A combiner avec blacklist **EXPLIQUER PLUS LES MATH?? car ça à l'air important on 'lutilise tout le temps**

**Zipf law**

## 2 Liste des pré-processing

### 2.1 Noisy entity removal

- Ponctuation, capital/lower case
- Stop word
- Rare word (less than a threshold)

### 2.2 Text Normalization

- Lemmatization : Lions $\rightarrow$ Lion, are $\rightarrow$ be
- Stemming

### 2.3 Word standardisation

Regular expression, e.g. for removing "." or expanding words' contractions ("I'll" ⇒ "I will")

# 3 Bag of Word

## 3.1 Stengths and drawbacks

Avantage :
- Easy light fast
- Opportunity to enrich (Context encoding, Part Of Speech)
- Efficient implementation
- Still very effective with classification

Limite :
- Loose document / sentence structure : mitigated with N-gram
- Several task missing : POS tagging, text generation
- Semantic gap : On peut pas utiliser la distance euclidienne pour mesurer la différence sémantique

## 3.2 Classification

**Naive Bayes** Rapide, interprétable, naturellement multiclasse. Perf à améliorer. Bien filtrer les stop word. Extention par robustesse (?)

**Classifieur linéaire** Scalable, attention sensible au dimension

# 4 Apprendre la sémantique

- Première approche : WordNet : trop statique (nouvelle expression, hashtag, vocab tecnique) et fait à la main
- Deuxième approche : Mesure de co-occurence : "Fair **and** legitimate", "Fair **but** brutal" $\rightarrow$ Marche bien + combo avec BoW

# 5 Unsupervised approches

## 5.1 LSA : Latent Semantic Analysis

But : réduire la dimension, réduidre le bruit. On un un vocabulaire de taille $d$, $N$ document, une LSA avec $k$ plus grande valeur propre, qui forme $k$ grand "concept" du corpus.

On décompose la matrice des occurence $X \in \mathbb{Q}^{N \times d}$ en
- $U \in \mathbb{R}^{k \times d}$ : une ligne de $U$ représente un vecteur de poids par mot dans chacun des $k$ concept.
- $\Sigma \in \mathbb{R}^{k \times k}$ valeur propre $\rightarrow$ **Tri par ordre croissant**pour garder les $k$ plus grande valeur (comme en BIMA)
- $V \in \mathbb{R}^{d \times k}$ Un colonne de $V$ représente l'intensité des relations entre le document $j$ et chaque concept
- Est-ce que $V$ et $U$ sont pareil??
- Comment on classif un nouveau document la dedans? Soit $q$ un nouveau document ou une query, $\hat{q} = \sigma^{-1} U^{-1} q$

Un ligne de la matrice de la matrice d'ocurence $t_i$, une colonne $d_i$. $t_i^T t_p$ corrélation de ces deux terme sur tout le corpus $\rightarrow XX^t$. $d_j^T d_q$ corrélation entre ces deux document pour tout terme $\rightarrow X^T X$

### 5.1.1 Stength and drawbacks

- Bon combo avec t-SNE.
- Entièrement basé sur BoW : même problème plus +
  - Not robust to stop word (=high singular values)
  - Problème forme négative
- Problème avec les mots rare dans les petits corpus (?)
- Peu combiner des dimension qui non pas de lien entre elle (combiner linéairement voiture et camion ok mais pas voiture et bouteille)

— Mauvaise prise en compte des doubles sens des mots, car un mot = un point de l'espace sémantique

### 5.2 Kmean

Comme d'hab, à combo avec LSA pour réduire les dimensions.

### 5.3 Dérivé LSA

— P-LSA : K-mean mais avec un assignement soft mesurer par la probabilité
— LDA : On rajoute un prior : modèle plus complexe → Les méthodes EM → on utilise MCMC pour estimer les vraissemblance
— D'autre extension bien complexe

## 6 Word Embeddings & Neural network

*Word that appear in similar contexts in text tend to have similar meanings*

### 6.1 Words2Vec

Implémentation de l'idée précédente : Modèle auto-supervisé, input OH encoding ; + backpropagation
— CBoW : Predict surrounding words (context) from central word ; work well for frequent words
— Skipgram : Predict context from central word ; work well for rare words

On note :
— Soft max en sortie sur $|V|$ (taille du vocab) valeur → Lourd → Evitable par négative sampling je crois ??
— Analogie dans l'espace latent $H$ : Féminin/Masculin, Capitale, sémantique
— **Difficulté pour aller au delà des mots** : Comment encoder une phrase entière dans $H$ → moyenne, sommes, min, max

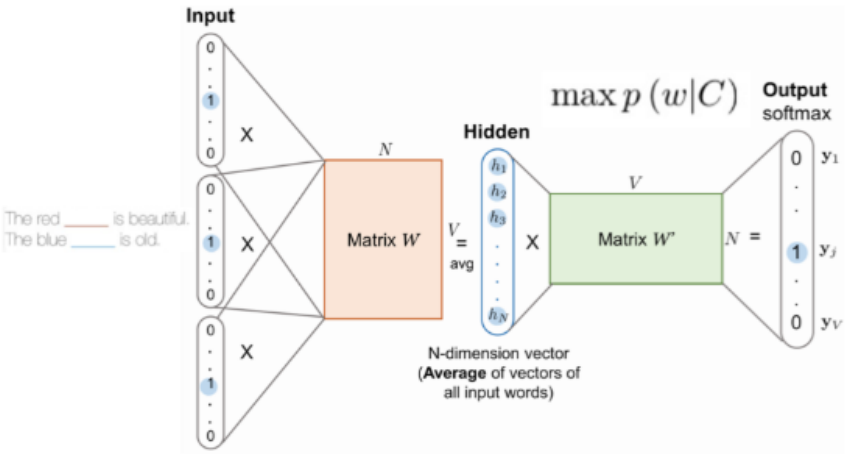


Figure 1 – CBoW

#### 6.1.1 Glove

— Input : matrice de co-occurence + accès aux word embeding
— On veut explicitement que la proximité dans l'espace latent représente la co-occurence des mots
— Fonction de cout et de prédiction de GloVe ≈ PLSA (espace latent pour les co-ocurences) + Contexte local + les words embeding

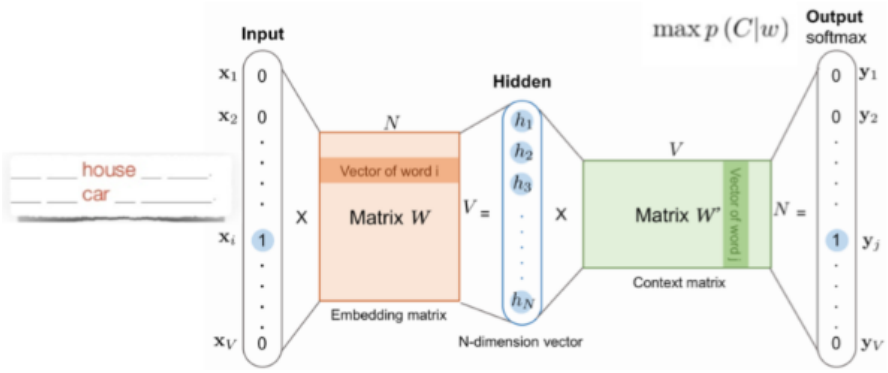


Figure 2 – Skip-Gram

## 6.2 Neural Networks

### 6.2.1 CNN

How to scale/classify with Word Embeding ? → Convolutional Neural networks

1. Latent représentation of the word
2. Couche de convolution
3. → Une représentation d'un document dans une dimension fixe
4. → Classication : Régréssion, NN w/ soft max, ...

+ une back propagation sur la layer de convolution, l'espace de représentation latent.

**Problème** : 2 mois de train

### 6.2.2 RNNs

— Code des séquences **entière** => parfait pour du texte
— Forme de mémoire du passé dans le réseau
— Problème de profondeur et d'évanouissement du gradient

3 types de classification :

— One to many : Image captionning (input une image, sortie plusieurs mots)
— Many to one : Sentiment clasification, Question answering (how many ...)
— Many to many : Text generation, translation (input anglais sortie français), speech2text

Evolution :

— Architecture spéciale contre la disparition du gradient : LSTM, GRU == long term memory
— Bi-directionnal RNNs : incorporate information from words both preceding and following

Toujours problème d'évanouissement du gradient + goulot d'étranglement pour encoder tout une phrase

## 6.3 Attention model & Transformers

— Encode l'information globale (!= CNN que local)
— Matrice d'attention == encodage de
  — Chaque mot encodé avec Key, Query, Value
  — la similarité de chaque mot de la phrase
  — inclusion du contexte globale
— Multi-head attention : on stack en parallèle les self-attention layer + combinaison
— Perte de la position : Fully connected layers : permutation invariant → Encodage manuel

### 6.3.1 BERT

Comme word2vec avec de l'attention : pré-train pour apprendre l'espace latent → Fine tune sur other task.

Token de classe CLS $t_0$ qui représente la phrase entière dans un espace latent $\rightarrow$ utile pour prédire la classe avec des modèle classique (regression ect) $\rightarrow$ **Plus le problème de W2V avec la moyenne ou la somme**

### 6.3.2 Transformers Decoder

GPT

— Prédire le mot suivant en fonction du contexte.

— Masked Attention : hide (mask) information about future tokens from the model. On donne la phrase au fur et à mesure pour pas tout apprendre d'un coup. On peut pas utiliser les mots d'après vu qu'on génère