

# Fouille de textes

M2 MIASHS / SSD – Université Grenoble-Alpes

Salah Aït-Mokhtar  
Naver Labs Europe

Contact: [sacours@outlook.com](mailto:sacours@outlook.com)

## Répertoire partagé du cours

Les documents relatifs à ce cours (support de cours, TP, etc.) sont accessibles en ligne avec ce lien:

<https://1drv.ms/u/s!Aksy8Pc5f6z8gocE0yYSalvjRaaX3Q?e=XdGTXB>

### IMPORTANT:

Les étudiants qui décident de suivre ce cours:

- Merci de m'envoyer ([sacours@outlook.com](mailto:sacours@outlook.com)) **nom et prénom**, ainsi que votre **parcours/option/filière** actuel(le)
- Je vous répondrai en incluant le lien du répertoire partagé

## Plan

- Fouille de textes: introduction
- Types d'applications en fouille de textes
- Principales étapes de traitements
- Introduction à quelques techniques de base
  - Hachage
  - Apprentissage automatique supervisé par réseaux de neurones artificiels
- Représentation vectorielle de textes/phrases/mots
- Représentations continues (plongements lexicaux)
- Reconnaissance d'entités nommées (REN)
- Extraction de relations
- Fouille d'opinions

## Fouilles de textes: introduction

### Fouille de textes

- Exploration automatique de textes en vue d'accéder à de l'information

### Fouille de textes / fouilles de données

- En fouille de données, les données sont structurées
- En fouilles de textes, les données ne sont pas ou sont peu structurées

## Why is Text Mining difficult?

Texts are not explicitly structured

- Text = unstructured data
- Text = semi-structured data (XML, HTML, etc)

Human language is **not** a formal language (e.g. programming language)

- Ambiguities
- Variants

## Données structurées

Structurées

- Données représentées dans un format bien défini et sans ambiguïtés
- L'exploitation directe des données (par ex. analyse de données) est possible

Exemple: 1 table CSV

- Chaque ligne représente une donnée de même type (par ex. des personnes)
- Chaque colonne est une variable (caractéristique) d'une donnée (par ex. âge, poids, etc.)

## Textes : données non structurées (1)

### Ambiguïtés

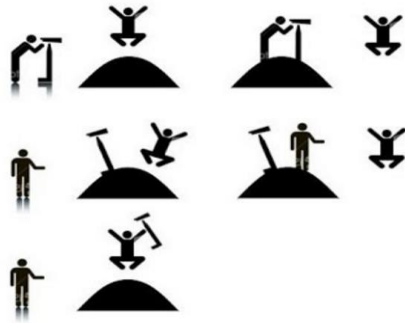
- Segmentation en mots
  - *bien que, aujourd'hui, pomme de terre, ...*
- Lexicales
  - *cours* : pluriel d'un cour d'eau ? enseignement ? pluriel d'une cour de justice ?
  - *tour*, ...
- Syntaxiques
  - *Il parle à la fille du concierge*
  - Complément de « parle » ou modifieur de « fille » ?
- Référentielles (entités nommées ou pronoms ou noms définis)
  - *George Bush* : père ou fils ?
  - *Jean appelle Max car il a faim* : il = Jean ou il = Max ?

## Texts → Ambiguities (1)

- Word segmentation
  - *en lugar de, despite of, pomme de terre, ...*
- Lexical ambiguities
  - *cours* (fr)
    - *curso* (de *agua*) + singular o plural
    - *curso* (*clase*) + singular o plural
    - *tasa* (*cotización*) + singular o plural
    - *correr* + indicativo/imperativo + presente + singular + persona 1 o 2
    - *corte* (tribunal) + plural
    - *patio* + plural
- Reference ambiguities (names, pronouns, nominal expressions)
  - *George Bush* → el padre o el hijo?
  - *John talks to Max because he is hungry*: he = John or he = Max ?

## Texts → Ambiguities: syntax/semantics

*I saw a man on the hill with a telescope.*



1. I saw the man. The man was on the hill. I was using a telescope.
2. I saw the man. I was on the hill. I was using a telescope.
3. I saw the man. The man was on the hill. The hill had a telescope.
4. I saw the man. I was on the hill. The hill had a telescope.
5. I saw the man. The man was on the hill. I saw him using a telescope.

<http://allthingslinguistic.com/post/52411342274/how-many-meanings-can-you-get-for-the-sentence-i>

S. Alt-Mokhtar, 25/09/2019 - !!!v0.8 – version incomplète et non définitive

Cours Fouille de textes, M2 MIASHS/SSD, Grenoble-Alpes

## Textes : données non structurées (2)

### Variabilité

- Une même information peut être exprimée avec de multiples expressions textuelles
- Exemple simple: les dates
  - *Le 21 septembre 2016*
  - *21/09/2016 ou 21/09/16 ou 21/9/2016 ou ...*
  - *21-09-2016 ou 21-09-16 ou 21-9-2016 ou ...*
  - *Le 21 septembre*, ou *21-09* ou *21/09*
  - *5 jours plus tard* (en référence au 16 septembre 2016)
  - *Le troisième mercredi de ce mois de septembre* (si le texte est écrit en septembre 2016)
  - *Mercredi prochain* (si le texte est écrit moins d'une semaine avant le 21/09/2016)
  - *etc. etc.*

S. Alt-Mokhtar, 25/09/2019 - !!!v0.8 – version incomplète et non définitive

Cours Fouille de textes, M2 MIASHS/SSD, Grenoble-Alpes

## Texts → Variants and synonymies

- The same information can be expressed with various textual forms
- Example: dates
  - *September 21, 2016*
  - *21/09/2016 or 21/09/16 or 21/9/2016 or 2016/09/21 ...*
  - *21-09-2016 or 21-09-16 or 21-9-2016 or ...*
  - *September 21, ou 21-09 ou 21/09*
  - *5 days later* (with reference to september 16, 2016)
  - *The third Wednesday of this month* (if text is written in september 2016)
  - *Next Wednesday* (if text is written less than one week before 21/09/2016)
  - *etc. etc.*

## Fouille de textes: motivations

### Grandes quantités de textes disponibles

- Internet
  - Wikipedia, articles de presse ou de blogs, publications scientifiques
  - Descriptions de produits, avis de clients/consommateurs,
  - Tweets, emails, etc.
- Intranet
  - Documentation technique (e.g. industrie aéronautique, automobile)
  - Dossiers des patients (médical), etc.

### Comment exploiter (automatiquement) l'information exprimée dans les textes ?

## FT: pourquoi faire?

### Extraction d'information (Information extraction / IE):

- Rechercher et extraire des informations spécifiques
- E.g. (médical): Extraire et structurer les critères d'éligibilité dans les essais cliniques

### Questions/réponses (Question answering / QA):

- L'utilisateur pose une question, le système extrait des passages de textes qui contiennent la réponse
- E.g. *De quel pays sont les vins de Ribera del Duero?*

Extrait Wikipedia

Le **ribera-del-duero**<sup>1</sup> est un vin **espagnol**, situé dans la région viticole de Castille et Léon, qui a une **AOC** (*Denominación de origen en español*).

### Fouille d'opinions (Opinion mining):

- Opinion des gens concernant une personne, une organisation, un produit ou un service : positive/negative?
- E.g. *Que pense les consommateurs des vins de Pardilla (Ribera del Duero)?*

etc.

## Plan

- Fouille de textes: introduction
- **Types d'applications en fouille de textes**
- Principales étapes de traitements
- Introduction à quelques techniques de base
  - Automates et transducteurs à états finis (AEF/TEF)
  - Hachage
  - Apprentissage automatique supervisé par réseaux de neurones artificiels
- Représentation vectorielle de textes/phrases/mots
- Représentations continues (plongements lexicaux)
- Application à l'étiquetage morpho-syntaxique
- Reconnaissance d'entités nommées (REN)
- Fouille d'opinions

## Types de d'applications en fouille de textes

- Tâches d'extraction de textes
  - Résultat : sélection de textes ou de parties de textes
- Tâches fondées sur la notion de similarité
  - Résultat : groupement de textes, assignation de catégories ou de caractéristiques
- Tâches d'extraction d'informations (structurées)
  - Résultat: production d'informations structurées

## Tâches d'extraction de textes

### Définition

- Tâches dont le but est de produire des textes en vue d'une exploitation manuelle (humaine)
- Le but de ces tâches n'est pas de structurer ou d'annoter les textes d'origine

### Exemples

- Recherche d'information/documentaire
- Résumé automatique de textes
- Systèmes de question/réponse (Q/R) ?



## Tâches fondées sur la notion de similarité

### Définition

- Tâches dont le but est d'assigner aux textes des catégories ou des caractéristiques
- Le but est d'annoter les textes ou des parties des textes

### Exemples

- Catégorisation ou classification de textes
- Regroupement automatique de documents (clustering)
- Identification automatique de la langue
- Caractérisation automatique des auteurs de textes (identification des auteurs, du genre, de la classe d'âge, etc.)
- Identification de mots-clé
- Fouille d'opinions (positive/négative/neutre)

## Tâches d'extraction d'informations (structurées)

Reconnaissance d'entités nommées (REN)

Extraction de relations entre EN

Extraction d'information (EI) : REN + extraction de relations

Fouille d'opinions par aspects

# Reconnaissance d'entités nommées (REN)

- Identification et extraction d'occurrences de noms de personnes, lieux, organisations, etc. dans les textes.
  - Les dates et certaines données numériques sont souvent considérées.
  - L'ensemble des types d'EN peut inclure des éléments spécifiques à un domaine, ex. noms de gènes en biomédical.

• Exemple:

*Barack Hussein Obama II (born August 4, 1961) is the 44th and current President of the United States. He is the first African American to hold the office. Obama previously served as a United States Senator from Illinois, from January 2005 until he resigned after his election to the presidency in November 2008.*

*A native of Honolulu, Hawaii, Obama is a graduate of Columbia University and Harvard Law School, where he was the president of the Harvard Law Review...*

## Information Extraction (IE)

IE: automatic extraction of structured information from unstructured and/or semi-structured text documents.

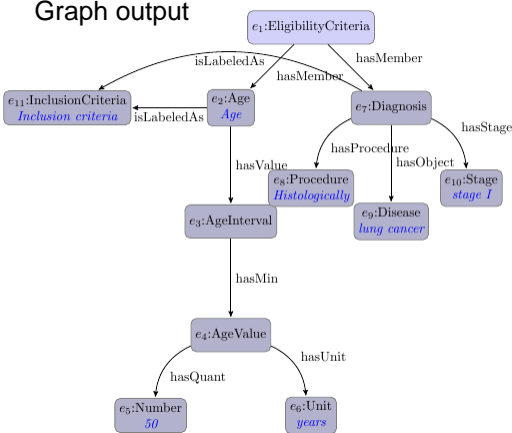
### Input text

#### Criteria

Inclusion criteria:

- Age greater than 50 years
- Histologically proven Stage I lung cancer...
- ...

### Graph output



# Extraction de relations entre EN

## Exemple

Barack Hussein Obama II (born August 4, 1961) is the 44th and current President of the United States. He is the first African American to hold the office. Obama previously served as a United States Senator from Illinois, from January 2005 until he resigned after his election to the presidency in November 2008.

A native of Honolulu, Hawaii, Obama is a graduate of Columbia University and Harvard Law School, where he was the president of the Harvard Law Review...

## Relations

Barack Hussein Obama II    dateOfBirth    August 4, 1961

Barack Hussein Obama II    presidentOf    United States

Obama    placeOfBirth    Honolulu, Hawaii

Obama    graduatedFrom    Columbia University

Obama    graduatedFrom    Harvard Law School

...

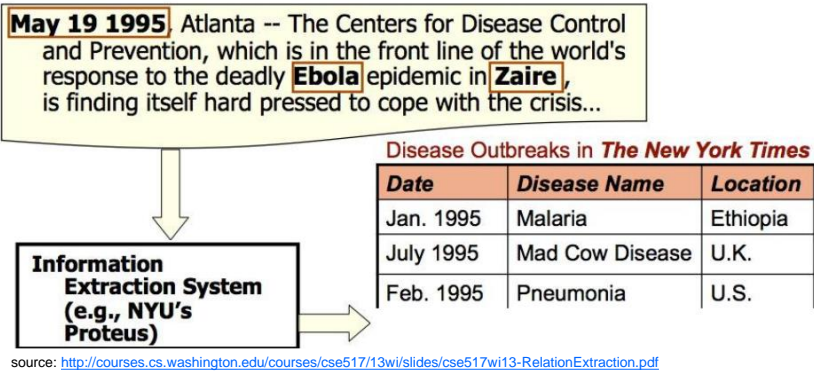
# Types de relations: domaine général

Relations	Examples	Types	
Affiliations			
	Personal	<i>married to, mother of</i>	PER → PER
	Organizational	<i>spokesman for, president of</i>	PER → ORG
	Artifactual	<i>owns, invented, produces</i>	(PER   ORG) → ART
Geospatial			
	Proximity	<i>near, on outskirts</i>	LOC → LOC
	Directional	<i>southeast of</i>	LOC → LOC
Part-Of			
	Organizational	<i>a unit of, parent of</i>	ORG → ORG
	Political	<i>annexed, acquired</i>	GPE → GPE

source: <http://courses.cs.washington.edu/courses/cse517/13wi/slides/cse517wi13-RelationExtraction.pdf>

source: <http://courses.cs.washington.edu/courses/cse517/13wi/slides/cse517wi13-RelationExtraction.pdf>

## Types de relations: domaine spécifique



## Aspect Based Sentiment Analysis (ABSA)

Exemple (Avis sur des restaurants)

“The food was great, the margaritas too, but the waitress was too busy being nice to her other larger party than to take better care of my friend and me.”


ABSA

Terms	Aspect	Polarity
food	FOOD	positive
margaritas	DRINKS	positive
waitress	SERVICE	negative

# Example: Aspect-Based Sentiment Analysis (ABSA)

## ABSA: Fine-grained opinion annotation

- Extract **sentiments** from user generated **comments** on **social media**
- Not only globally, but at the level of **aspects** of **entities** (movies, restaurants, cell phones,...)
- **Aspects are features** of an entity (*service, food* in a restaurant; *screen, battery* of a cell phone,...)



Flora N

280 136

Reviewed 3 days ago

### Very good food but poor service

This restaurant is just near by the sea and the terrace is amazing. We tasted nice (but too small) quinoa salad and mussels which were excellent. Everything could have been great except our waiter who was almost rude. Beautiful sea view didn't make us forget how much is was unpleasant. To bad!

Term	Aspect	Polarity
Quinoa salad, mussels	Food Quality	positive
Quinoa salad, mussels	Food Quantity	negative
Terrace, sea view	Location	positive
waiter	Service	negative

## Plan

- Fouille de textes: introduction
- Types d'applications en fouille de textes
- **Principales étapes de traitements**
- Introduction à quelques techniques de base
  - Hachage
  - Apprentissage automatique supervisé par réseaux de neurones artificiels
- Représentation vectorielle de textes/phrases/mots
- Représentations continues (plongements lexicaux)
- Application à l'étiquetage morpho-syntaxique
- Reconnaissance d'entités nommées (REN)
- Fouille d'opinions

## Principales étapes de traitements en FT

Acquisition des documents

Nettoyage (filtrage des parties non-pertinentes, balises, ponctuations, ...)

Segmentation

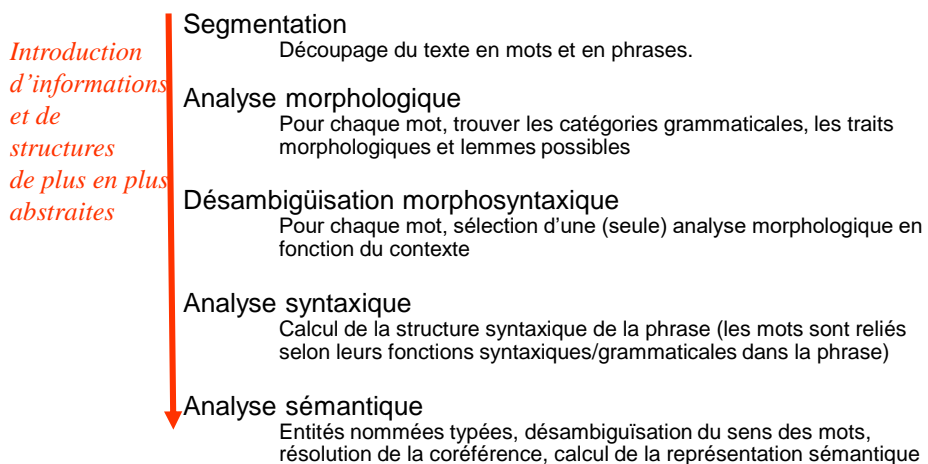
Représentation

Traitement (classification, regroupement, etc.)

Et optionnellement : d'autres traitements linguistiques

- Etiquetage morpho-syntaxique
- Analyse syntaxique
- Analyse sémantique

## Les niveaux d'analyse linguistiques



# Segmentation

Découpage du texte en une suite de « tokens » (mots, symboles de ponctuation, etc.), découpage des phrases

Exemple:       « *Aujourd’hui, Jean n’a pas mangé de pommes de terre.* »

Aujourd’hui  
,  
Jean  
n’  
a  
pas  
mangé  
de  
pommes de terre  
.

# Analyse morphologique

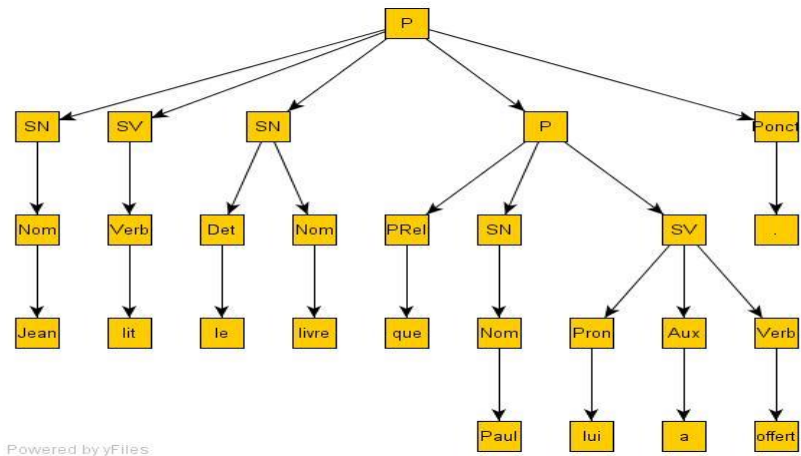
Mots du texte	Analyse morphologique	
	lemme	+ traits morpho-syntaxiques
Aujourd’hui	aujourd’hui	+Adv+Temp
,	,	+Ponct+Virg
Jean	Jean	+Nom+Propre+Sg+Masc
ne	ne	+Adv+Neg
lit	lit	+Nom+Sg+Masc
lire	lire	+Verb+IndP+P3+Sg
pas	pas	+Adv+NegPas
pas	pas	+Nom+SgPl+Masc
le	le	+Det+Art+Sg+Masc
le	le	+Pron+Sg+Masc+P3+Acc
journal	journal	+Nom+Sg+Masc
.	.	+Ponct+Point

# Etiquetage morphosyntaxique

Mots du texte	Etiquetage morphosyntaxique (garder une seule analyse morphologique)	
Aujourd'hui	aujourd'hui	+Adv+Temp
,	,	+Ponct+Virg
Jean	Jean	+Nom+Propre+Sg+Masc
ne	ne	+Adv+Neg
lit	lit	+Nom+Sg+Masc
lire	lire	+Verb+Avoir+IndP+P3+Sg
pas	pas	+Adv+NegPas
	pas	+Nom+SgPl+Masc
le	le	+Det+Art+Sg+Masc
	le	+Pron+Sg+Masc+P3+Acc
journal	journal	+Nom+Sg+Masc
.	.	+Ponct+Point

# Analyse syntaxique: structure de constituants (ou syntagmes)

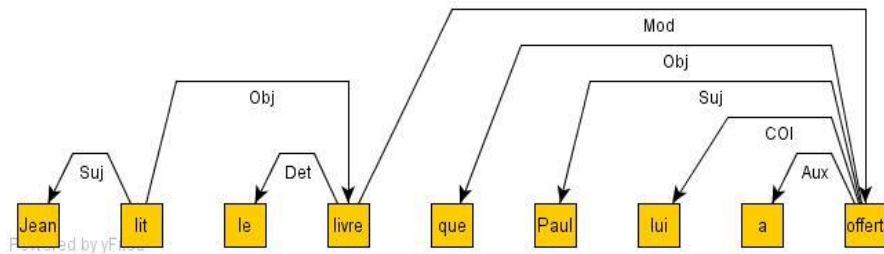
« Jean lit le livre que Paul lui a offert. »





## Analyse syntaxique: structure de dépendances

« Jean lit le livre que Paul lui a offert. »



SUJ(lit,Jean), OBJ(lit,livre), DET(livre,le), MOD(livre,offert)  
SUJ(offert,Paul), OBJ(offert,que), COI(offert,lui), AUX(offert,a)

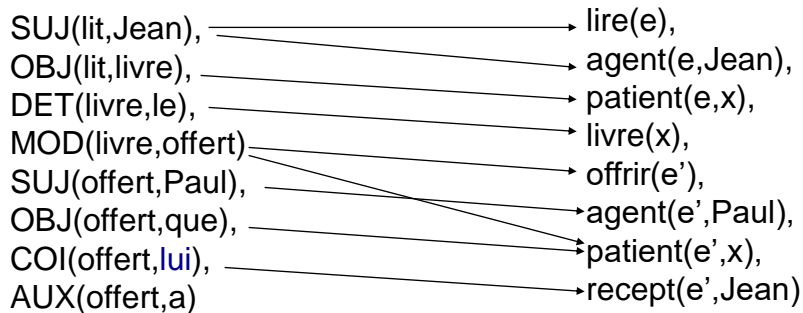
## Analyseur syntaxique en dépendances

Démo en ligne:

- <https://www.connexor.com/nlplib/?q=demo/syntax>
- Plusieurs langues (dont le français)

## Analyse sémantique: passer des fonctions syntaxiques à une représentation sémantique

« Jean lit le livre que Paul lui a offert. »



Fonctions syntaxiques

Formes logiques

## Etiquetage de rôles sémantiques (Semantic Role Labeling (SRL))

- Etiquetage des arguments sémantiques des mots prédicatifs
  - Les prédicats sont souvent des verbes, mais également des noms et des adjectifs
  - Exemples (prédicat mis en bleu, arguments soulignés):
    - Marie **mange** une pizza
    - Le ministre **a déclaré** à la presse que le chômage baissera au 1<sup>er</sup> trimestre
    - Le candidat, **fier** de sa prestation, a déclaré que...
    - La construction **de l'immeuble** par la mairie a été décidé...
  - Chaque argument « remplit » un rôle sémantique
- Les rôles sémantiques ne sont pas des relations syntactiques (grammaticales)
  - Jean **ouvre** la porte avec la clé. : sujet="Jean", agent="Jean", patient="la porte"
  - La clé **ouvre** la porte. : sujet="La clé", instrument="La clé"
  - La porte **ouvre** sur le jardin. : sujet="La porte", patient="« la porte »"

## Etiquetage de rôles sémantiques (Semantic Role Labeling (SRL))

- Le type et le nombre des rôles sémantiques dépendent du sens du prédicat et de l'application en vue
  - Rôles sémantiques généraux
    - Agent, Patient, Instrument, Lieu, Temps, Manière
  - Rôles sémantiques spécifiques
    - E.g. Locuteur, Message, Thème, etc.
- Etiqueter les rôles sémantiques revient à répondre à des questions:  
Qui fait quoi à qui, quand, où et comment ?

## Exemple : Le cadre (frame) “Motion” de Framenet

FrameNet: <http://framenet.icsi.berkeley.edu/>

### Motion

#### Definition:

Some entity (**Theme**) starts out in one place (**Source**) and ends up in some other place (**Goal**), having covered some space between the two (**Path**). Alternatively, the **Area** or **Direction** in which the **Theme** moves or the **Distance** of the movement may be mentioned.

That kite you see just to the right of his head was **MOVING** around pretty fast but the camera seemed to catch it ok.

There are several accounts of the stench **DRIFTING** to shore from the ships in the middle of the river

Dust particles **FLOATING** about made him sneeze uncontrollably.

The grill, unsecured, **ROLLED** a few feet across the yard.

The swarm **WENT** away to the end of the hall.

#### Lexical Units

*blow.v, circle.v, coast.v, drift.v, float.v, fly.v, glide.v, go.v, meander.v, move.v, roll.v, slide.v, snake.v, soar.v, spiral.v, swerve.v, swing.v, travel.v, undulate.v, weave.v, wind.v, zigzag.v*

# Coréférence

Pour que l'analyse sémantique se fasse, il faut également calculer les liens de coréférence dans le texte

**Coréférence:** relation entre expressions linguistiques, identiques ou différentes, qui réfèrent à la même entité du monde

Exemple:

- *Nicolas Sarkozy et le premier ministre visitent aujourd'hui une clinique parisienne. Le président y rencontrera les représentants du personnel soignant. Il annoncera des mesures nouvelles pour l'amélioration des soins.*

## Types de coréférences

**Cataphore pronominale:**  
référence vers l'avant sur  
President George W. Bush7

**Anaphore pronominale:**  
référence vers l'arrière sur  
President George W. Bush7

**Anaphore nominale**  
référence sur  
President George W. Bush7

By choosing **Paul Wolfowitz**<sub>0</sub> for the post of **World Bank**<sub>2</sub>, president right after **he**<sub>7</sub>, nominated **John Bolton**<sub>4</sub> **US**<sub>5</sub> ambassador to **the United Nations**<sub>6</sub>, **President George W. Bush**<sub>7</sub> has signaled **his**<sub>7</sub> determination to send **his**<sub>7</sub> administration 's hardliners to the forefront of the international arena.

...

Despite **his**<sub>0</sub> assurances , **Wolfowitz**<sub>0</sub> does not come off as a specialist on poverty and international development issues .

...

For **his**<sub>4</sub> part , **Bolton**<sub>4</sub> who will defend the **US**<sub>5</sub> administration 's foreign policy at **the United Nations**<sub>6</sub> , has at times in the past been tough on the world body.

"There is no such thing as **the United Nations**<sub>6</sub>," **he**<sub>4</sub> stated in 1994.

...

**Bush**<sub>7</sub> said Wednesday that **the United States**<sub>5</sub> and **its**<sub>5</sub> European allies would seek **UN**<sub>6</sub> **Security Council**<sub>95</sub> action against **Iran**<sub>14</sub> if **Tehran**<sub>97</sub> rejected incentives to limit **its**<sub>97</sub> nuclear programs. "The understanding is, **we**<sub>7</sub> go to **the Security Council**<sub>95</sub> if **they**<sub>97</sub> reject the offer. And **I**<sub>7</sub> hope **they**<sub>97</sub> don't. **I**<sub>7</sub> hope **they**<sub>97</sub> realize the world is clear about making sure that **they**<sub>97</sub> don't end up with a nuclear weapon," **the US**<sub>5</sub> **prsident**<sub>7</sub> said.

## Extraction d'information et étapes de traitement

Amélioration de la qualité de l'extraction lorsque des traitements linguistiques sont effectués

Méthode de base : sac-de-mots (Words)

### Zhou et al. 2005 results

Features	P	R	F
Words	69.2	23.7	35.3
+Entity Type	67.1	32.1	43.4
+Mention Level	67.1	33.0	44.2
+Overlap	57.4	40.9	47.8
+Chunking	61.5	46.5	53.0
+Dependency Tree	62.1	47.2	53.6
+Parse Tree	62.3	47.6	54.0
+Semantic Resources	63.1	49.5	55.5

source: <http://courses.cs.washington.edu/courses/cse517/13wi/slides/cse517wi13-RelationExtraction.pdf>

## Plan

- Fouille de textes: introduction
- Types d'applications en fouille de textes
- Principales étapes de traitements
- **Introduction à quelques techniques de base**
  - Hachage
  - Apprentissage automatique supervisé par réseaux de neurones artificiels
- Représentation vectorielle de textes/phrases/mots
- Représentations continues (plongements lexicaux)
- Application à l'étiquetage morpho-syntaxique
- Reconnaissance d'entités nommées (REN)
- Fouille d'opinions

## Plan

- Fouille de textes: introduction
- Types d'applications en fouille de textes
- Principales étapes de traitements
- **Introduction à quelques techniques de base**
  - **Hachage**
    - Apprentissage automatique supervisé par réseaux de neurones artificiels
- Représentation vectorielle de textes/phrases/mots
- Représentations continues (plongements lexicaux)
- Application à l'étiquetage morpho-syntaxique
- Reconnaissance d'entités nommées (REN)
- Fouille d'opinions

## Hachage

Association d'un entier unique (clé de hachage) à une chaîne de caractères

- Idéalement, 2 chaînes distinctes devrait avoir des clés de hachage différentes
- Sinon: collision

Intérêt

- Accès direct à la chaîne dans un ensemble (ou à des informations qui lui sont associés) → la clé de hachage est un index
  - Sinon: recherche séquentielle

Exemple de fonction de hachage (dans Java)

- Si  $s$  est une chaîne,  $n$  sa longueur et  $s[i]$  le code numérique du caractère à la position  $i$  :

$$h(s) = \sum_{i=0}^{n-1} s[i] \cdot 31^{n-1-i}$$

## Plan

- Fouille de textes: introduction
- Types d'applications en fouille de textes
- Principales étapes de traitements
- **Introduction à quelques techniques de base**
  - Hachage
  - **Apprentissage machine supervisé par réseaux de neurones artificiels**
- Représentation vectorielle de textes/phrases/mots
- Représentations continues (plongements lexicaux)
- Application à l'étiquetage morpho-syntaxique
- Reconnaissance d'entités nommées (REN)
- Fouille d'opinions

## Classification par apprentissage machine supervisé

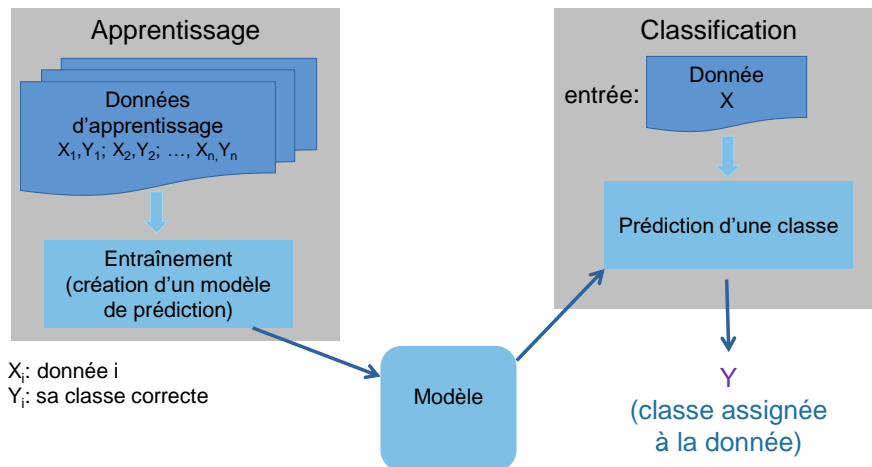
### Problème

- Un ensemble de données (événements/instances/items/...) qui doivent être classées
  - On doit assigner une classe/catégorie à chaque donnée
  - Les données peuvent être une séquence (ordonnée) ou non
  - **Supervisé** → on dispose d'exemples de classification
    - Un ensemble de données correctement étiquetées avec leurs classes respectives

### Approche

- Utiliser des exemples (données déjà classifiées) pour apprendre un modèle (phase d'entraînement)  
→ apprentissage supervisé
- Le modèle permet ensuite de « prédire » une classe pour des données nouvelles (non observées jusque-là).

## Classification par apprentissage machine supervisé



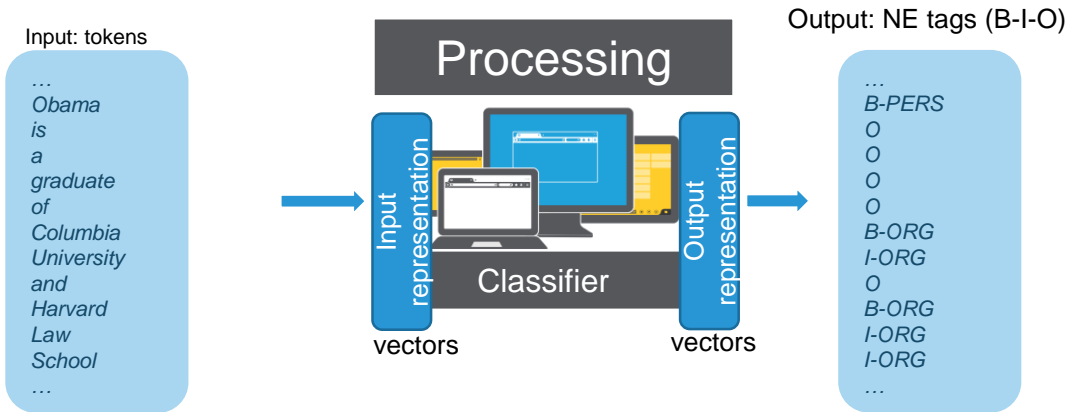
## Exemple de classification par apprentissage

- **Données:**
  - Dates sur une année (par exemple: le 13 novembre 2019)
- **Problème de classification:**
  - Pour une date dans le futur, **prédire la qualité de circulation** sur la Rocade Sud de Grenoble
- **3 classes possibles pour chaque donnée (c-à-d chaque match):**
  - **Fluide**
  - **Moyenne** (encombrée, circulation un peu lente)
  - **Difficile** (bouchons, blocages)
- **On dispose d'exemples dans le passé (données d'entraînement):**
  - Par ex. la qualité de la circulation pour chaque jour des 20 dernières années.



# Text Mining Tasks as Classification Problems

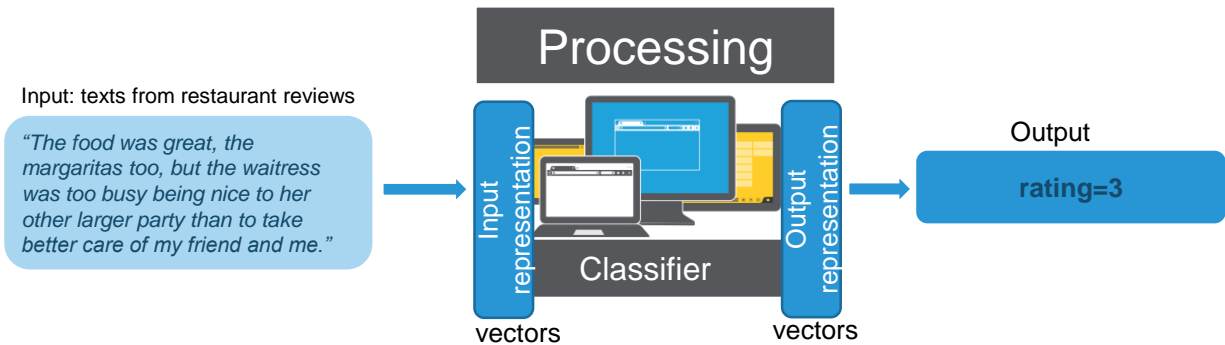
## Reconnaissance d'entités nommées (REN)



**Classes:** O, B-PERS, I-PERS, B-ORG, I-ORG, B-LOC, I-LOC, etc...

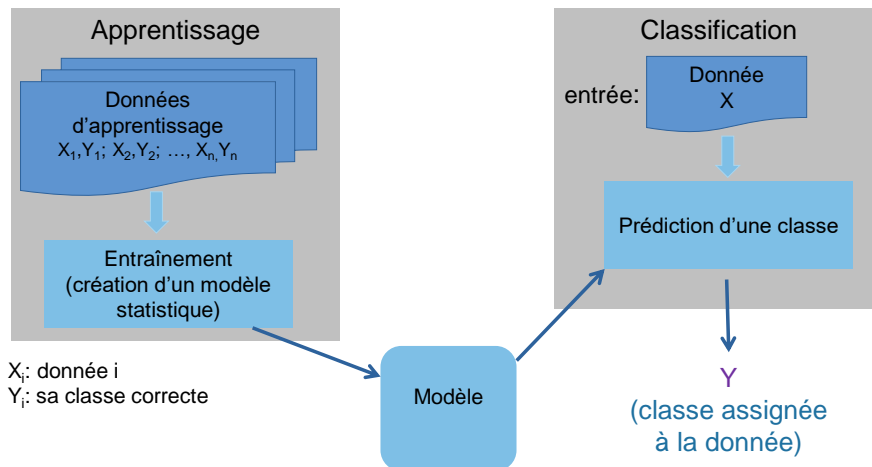
# Text Mining Tasks as Classification Problems

## Sentiment analysis



**Classes:** rating=1, rating=2, rating=3, rating=4, rating=5

## Classification par apprentissage machine supervisé



## Classification par apprentissage: représentation des données

### Vecteur de caractéristiques (ou attributs, ou traits)

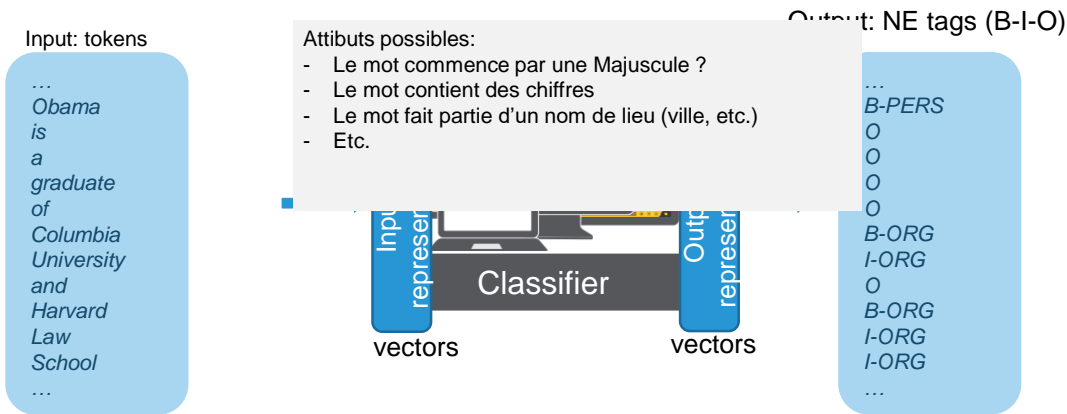
- Une donnée est représentée sous forme de vecteur
  - $X_i = \langle x_{i1}, x_{i2}, \dots, x_{in} \rangle$
- Chaque élément  $x_{ij}$  décrit une caractéristique de la donnée, supposée pertinente pour sa classification
- Le choix des attributs est très important
  - Très déterminant pour la qualité du modèle de classification

### Exemple d'attributs (qualité de la circulation sur la Rocade Sud)

- Jour est férié ou pas
- Saison (Hiver ? Été ?)
- Période de vacances ou pas
- Météo (pluie, neige)
- Fermeture voie sur berge ou pas
- Etc...

# Example of features for NER

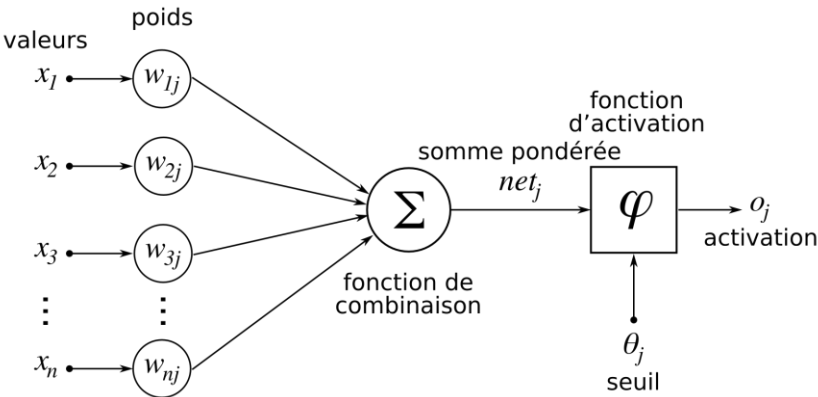
## Reconnaissance d'entités nommées (REN)



**Classes:** O, B-PERS, I-PERS, B-ORG, I-ORG, B-LOC, I-LOC, etc...

# Perceptron : neurone artificiel

(F. Rosenblatt, 1957)



Source du schéma: Wikipedia

## Perceptron et classification binaire (1)

### Classification binaire

- 2 classes possibles

### Exemples

- Diagnostic médical pour une maladie spécifique
  - Donnée: un patient représenté avec un ensemble de caractéristiques
    - Sexe, poids, âge, valeurs de tests sanguins, historique familial, etc.
  - Classes possibles: positif / négatif
- Fonction booléenne OR
  - Donnée: 1 paire de bits: (0,0) ou (0,1) ou (1,0) ou (1,1)
  - Classes possibles: 0 / 1

## Perceptron et classification binaire (2)

- Une donnée  $x$  est représentée comme un vecteur
$$x = (x_1, x_2, x_3, \dots, x_n)$$
  - Chaque dimension représente un attribut ou trait de la donnée, supposé être pertinent pour sa classification
  - $X$ : ensemble des données
- Un ensemble  $Y$  de 2 classes possibles :  $\{c_1, c_2\}$ 
  - Généralement représentées avec  $\{-1, +1\}$  ou  $\{0, 1\}$
- Problème de classification binaire :
  - A chaque donnée  $x$  de  $X$  on doit associer une classe  $c$  de  $Y$

## Perceptron et classification binaire (3)

### Fonction de combinaison $f$ :

Elle associe un nombre réel à chaque donnée  $x$ :

- $f(x) = w^T x$
- $w$  : vecteur de « poids » ( $w_1, w_2, w_3, \dots, w_n$ )
  - Poids (importance) de chaque trait
- $$f(x) = w_1 \cdot x_1 + w_2 \cdot x_2 + \dots + w_n \cdot x_n$$
$$= \sum_{i=1}^n w_i \cdot x_i + w_b$$

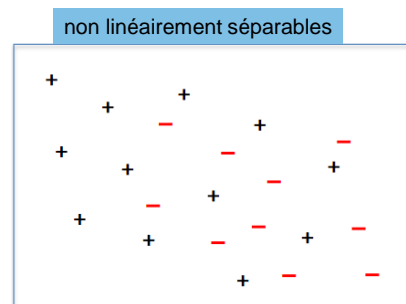
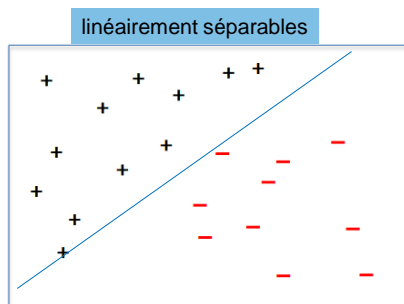
= la somme pondérée des valeurs de tous les traits pour la donnée  $x$ , plus le biais

**Paramètres du modèle** : vecteur des poids (des traits)

## Perceptron et classification binaire (4)

### Hypothèse de travail

- Les données sont linéairement séparables



## Perceptron et classification binaire (4)

### Prédiction linéaire:

- Choisir un seuil en fonction de la représentation des classes (sorties)
  - $\{-1, +1\} \rightarrow \text{seuil} = 0$
  - $\{0, 1\} \rightarrow \text{seuil} = 0.5$
- Pour chaque donnée  $x$ , choisir la classe  $c_{res}$  en fonction de la somme pondérée des valeurs de traits et du seuil:
  - $c_{res} = 1$  si  $\sum_{i=0}^n w_i \cdot x_i > 0$
  - $c_{res} = -1$  si  $\sum_{i=0}^n w_i \cdot x_i < 0$
- **Procédure** : pour chaque donnée  $x$ :
  1. Construire la représentation vectorielle de  $x$
  2. Calculer  $f(x) = \sum_{i=1}^n w_i \cdot x_i$
  3. Si  $f(x) > 0$ ,  $c_{res} = +1$  sinon  $c_{res} = -1$

Les poids des traits sont donc déterminants. Comment les choisir ?

## Perceptron et classification binaire (5)

### Calcul des poids des traits par apprentissage sur le corpus d'entraînement :

1. Initialisation: donner des valeurs aléatoires aux poids (entre -1 et +1)
2. Répéter (itérations)
  - Réordonner aléatoirement les exemples d'entraînement
  - Pour chaque donnée  $x$ , et sa classe  $c_{ref}$  (classe de référence)
    - Construire le vecteur de traits de  $x$
    - Calculer  $c_{res}$  avec la fonction de combinaison et le seuil (voir pages précédentes)
    - Si  $(c_{res} \neq c_{ref})$  : étiquetage incorrect ! Ajuster les valeurs des poids :
      - Pour chaque trait  $i=1, \dots, n$ , mettre à jour les poids:
        - $W_i \leftarrow W_i + (y_{ref} - y_{res}) \cdot x_i$

## Perceptron et classification multiclass (1)

- Un ensemble  $Y$  de  $m$  classes ou étiquettes possibles :  $\{c_1, c_2, c_3, \dots, c_m\}$ , avec  $m > 2$
- Problème de classification :
  - A chaque donnée  $x$  de  $X$  on doit associer une étiquette  $c$  de  $Y$

## Perceptron et classification multiclass (2)

**Fonction  $f$**  : associe à chaque paire  $(x, c)$  (une donnée et une étiquette quelconque) un nombre réel :

- $f(x, c) = w_c^T \cdot x$
- $w_c$  : vecteur de « poids » ( $w_{1c}, w_{2c}, w_{3c}, \dots, w_{nc}$ )
  - Poids (importance) de chaque trait par rapport à la classe  $c$  (biais inclus)
- $$f(x, c) = w_{1c} \cdot x_1 + w_{2c} \cdot x_2 + \dots + w_{nc} \cdot x_n$$
$$= \sum_{i=1}^n w_{ic} \cdot x_i$$

= la somme pondérée des valeurs de tous les traits pour la donnée  $x$

**Paramètres du modèle** : matrice des poids (Traits x Classes)

## Perceptron et classification multiclass (3)

### Prédiction linéaire:

- Pour chaque donnée  $x$ , choisir l'étiquette  $c_{res}$  qui conduit à une valeur maximale de la somme pondérée des valeurs de traits (biais inclus)

$$c_{res} = \operatorname{argmax}_c \sum_{i=0}^n w_{ic} \cdot x_i$$

- **Procédure** : pour chaque donnée  $x$  :
  1. Construire le vecteur de traits de  $x$
  2. Pour chaque classe  $c$  de l'ensemble des classes :
    - Calculer  $f(x,c) = \sum_{i=0}^n w_{ic} \cdot x_i$
  3. Sélectionner la classe qui conduit à la valeur maximale de  $f(x,c)$

Les poids des traits sont donc déterminants ! Comment les choisir ?

## Perceptron et classification multiclass (4)

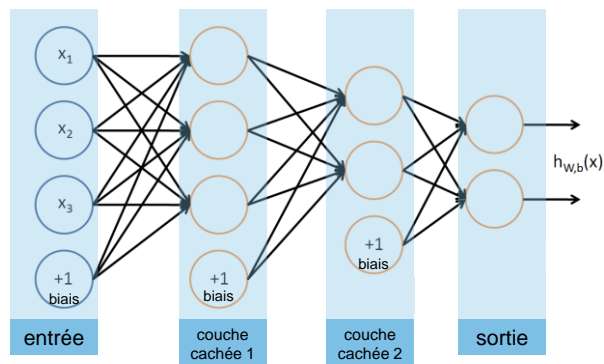
### Calcul des poids des traits par apprentissage sur le corpus d'entraînement :

1. Initialisation: donner des valeurs aléatoires aux poids
2. Répéter (itérations)
  - Réordonner aléatoirement les données d'entraînement
  - Pour chaque donnée  $x$ , et son étiquette  $c_{ref}$  (étiquette de référence)
    - Construire le vecteur de traits de  $x$
    - Calculer  $c_{res} = \operatorname{argmax}_c \sum_{i=1}^n w_{ic} \cdot x_i$  (voir pages précédentes)
    - Si ( $c_{res} \neq c_{ref}$ ) : étiquetage incorrect ! Ajuster les valeurs des poids :
      - Pour chaque trait  $i=1, \dots, n$ , avec  $x_i \neq 0$ , mettre à jour les poids:
        - $w_{icref} \leftarrow w_{icref} + 1$
        - $w_{icres} \leftarrow w_{icres} - 1$



## Réseaux de neurones multicouches

- En anglais: feedforward networks (FFN)
- Cas particulier: le perceptron multi-couches (en anglais: multi-layer perceptron (MLP))
  - Une ou plusieurs couches « cachées » entre l'entrée et la sortie
  - Chaque neurone d'une couche est connecté à chaque neurone de la couche suivante



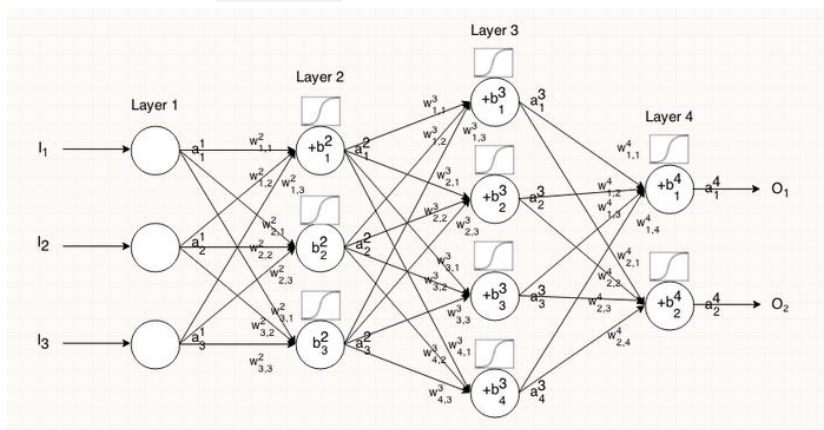
S. Ait-Mokhtar, 25/09/2019 - !!!v0.8 – version incomplète et non définitive

Cours Fouille de textes, M2 MIASHS/SSD, Grenoble-Alpes

## Activation (sortie) d'un neurone

$$a_j^i = \sigma \left( \sum_k (w_{jk}^i \cdot a_k^{i-1}) + b_j^i \right)$$

indice de la couche  $i$   
 indice du neurone dans sa couche  $j$   
 fonction d'activation  $\sigma$   
 poids de la  $k$ -ème entrée pour le neurone  $j$  de la couche  $i$   
 biais du neurone  $j$  de la couche  $i$   
 activation du  $k$ -ème neurone de la couche précédente ( $i-1$ )



S. Ait-Mokhtar, 25/09/2019 - !!!v0.8 – version incomplète et non définitive

Cours Fouille de textes, M2 MIASHS/SSD, Grenoble-Alpes

## Notation

$$a_j^i = \sigma \left( \underbrace{\sum_k (w_{jk}^i \cdot a_k^{i-1}) + b_j^i}_{z_j^i} \right)$$

Notation plus générale  
matrice des poids de la couche i  
×  
vecteur de sortie de la couche précédente (i-1)

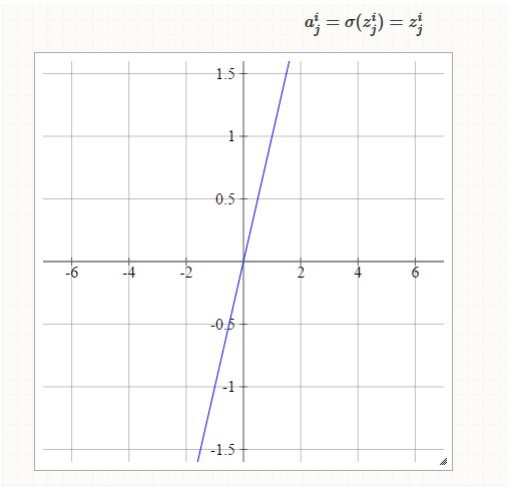
$$a^i = \sigma(w^i \times a^{i-1} + b^i)$$

## Fonctions d'activation

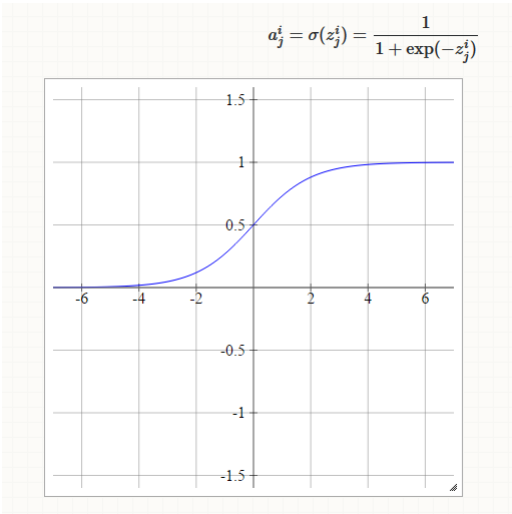
Plusieurs fonctions d'activation sont possibles

- Identité (linéaire: la somme est propagée sans modification)
- Non linéaires
  - Sigmoidé
  - Tangente hyperbolique (tanh)
  - ReLU (Rectified Linear Unit)
  - etc.

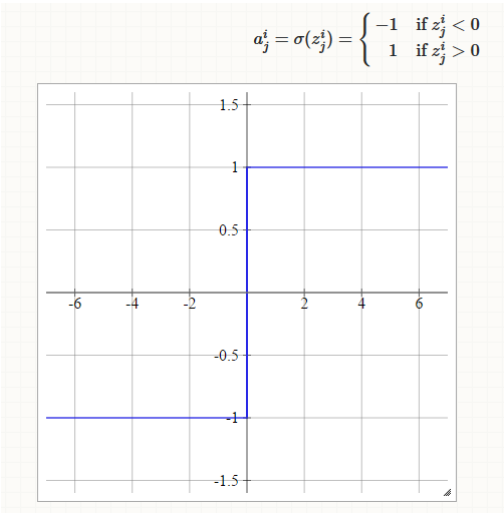
Fonction d'activation: **identité**



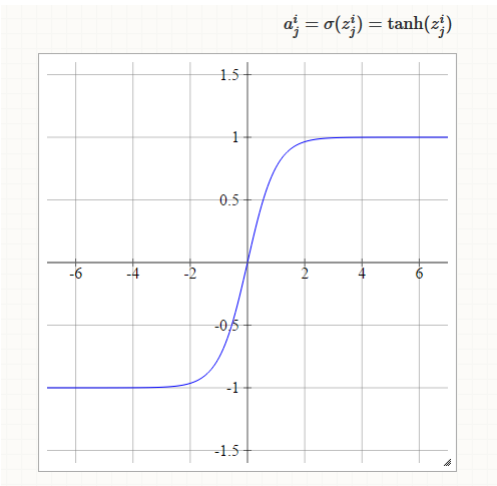
Fonction d'activation : **sigmoïde**



# Fonction d'activation : bipolaire

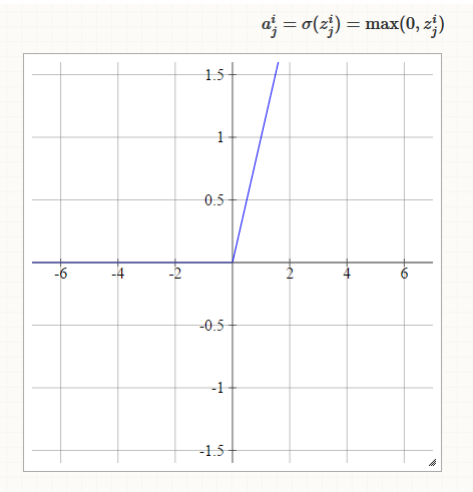


# Fonction d'activation : tanh

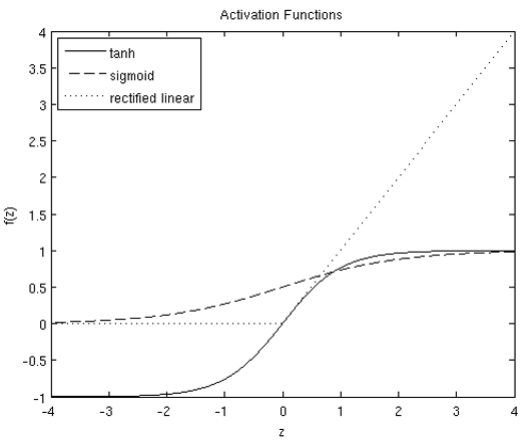


# Fonction d'activation : ReLU ou Ramp

ReLU = Rectified Linear Unit



# Fonctions d'activation : comparaison



## Cas particulier: couche Softmax

### Couche de sortie

- Prend le vecteur de sortie de la couche précédente
- Retourne un vecteur de même dimension dont les valeurs sont normalisées entre 0 et 1
  - Normalise les K sorties de la couche précédente entre 0 et 1
- Valeurs de la sortie interprétables comme des probabilités
  - Cas de la classification en m classes :  $c_0, c_1, \dots, c_j, \dots, c_{m-1}$
  - Valeur de la sortie j = probabilité que l'entrée soit de la classe  $c_j$

$$a_j^i = \frac{\exp(z_j^i)}{\sum_k \exp(z_k^i)}$$

## Entraînement par rétropropagation

### Rétropropagation (*backpropagation* ou *backprop*)

- Ajustement des poids de la dernière couche en fonction de l'erreur de sortie
- Répercussion sur les poids de la couche précédente
- Récursion jusqu'aux poids de la 1<sup>ère</sup> couche (qui suit la couche d'entrée)

### Utilisation de l'algorithme de la descente du gradient (*stochastic gradient descent* – SGD)

- But : minimisation de la fonction d'erreur
- Par calcul des dérivées partielles de la fonction d'erreur

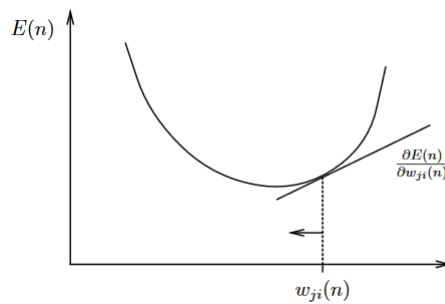
### Plus de détails :

- <http://wcours.gel.ulaval.ca/2010/h/IFT3901/default/5notes/RetroPerceptron.pdf>
- <http://neuralnetworksanddeeplearning.com/chap1.html> (en anglais)

## Rétropropagation et descente du gradient

### Intuition

- Ajuster les poids dans la direction inverse du gradient pour atteindre un minimum (local) de l'erreur



## Entraînement par rétropropagation

### Fonction d'erreur (ou de coût/perte – **loss function**)

- Mesure de la différence entre valeur prédite par le modèle et valeur correcte
- Dépend du type de classification

## Fonction d'erreur: cas multi-classe mono-label

Classification multi-classes mono-label (1 seule classe possible par instance)

- Fonction d'activation: **softmax**

Fonction d'erreur: entropie croisée (**categorical cross-entropy**):

$$L(y, \hat{y}) = - \sum_{j=0}^M \sum_{i=0}^N (y_{ij} * \log(\hat{y}_{ij}))$$

## Fonction d'erreur: cas binaire, ou multi-classe multi-label

Classification binaire, ou multi-classes multi-labels

- Fonction d'activation: **sigmoid**

Erreur d'entropie croisée binaire (**binary cross-entropy**):

$$L(y, \hat{y}) = - \frac{1}{N} \sum_{i=0}^N (y * \log(\hat{y}_i) + (1 - y) * \log(1 - \hat{y}_i))$$



## Librairies pour les réseaux de neurones

### Plusieurs librairies existent

- Java: Deeplearning4j, Encog, Neuroph, Joone, etc.
- Python: PyTorch, Tensorflow / Keras, MxNet, etc.

### Langage de programmation choisi pour le cours et le TP

- **Python**
  - Permet un prototypage rapide
  - Disponibilité d'un large choix de librairies/modules pour le TAL et l'IA
- Important: version **Python 3.7.x**

## Librairies Python pour le cours/TP

- Réseaux de neurones/Deep learning: **Keras**
  - Simplicité du code, bonne documentation
  - Intègre Tensorflow
  - Large choix de types/architectures de RNs
- Pour le prétraitement des documents texte et représentations
  - nltk
  - spacy
  - gensim

Un guide d'installation de Python 3.7 (via Anaconda) et des librairies sera fourni pour le TP

# Exemples avec Keras en Python

## Plan

- Fouille de textes: introduction
- Types d'applications en fouille de textes
- Principales étapes de traitements
- Introduction à quelques techniques de base
  - Hachage
  - Apprentissage automatique supervisé par réseaux de neurones artificiels
- **Représentation vectorielle de textes/phrases/mots**
- Représentations continues (plongements lexicaux)
- Reconnaissance d'entités nommées (REN)
- Fouille d'opinions

## Classification et fouille de textes

Les tâches de FT peuvent être modélisées comme des problèmes de classification

Nature des données à représenter

- Dépend de la nature de la tâche
- Exemples
  - Catégorisation de documents: données = textes entiers
  - Reconnaissances d'EN: données = mots ou tokens
  - Extraction de relations entre EN: données = couples d'EN
  - etc.

Nécessité de représenter les textes/phrases/mots avec des vecteurs

## Reconnaissance d'entités nommées (REN)

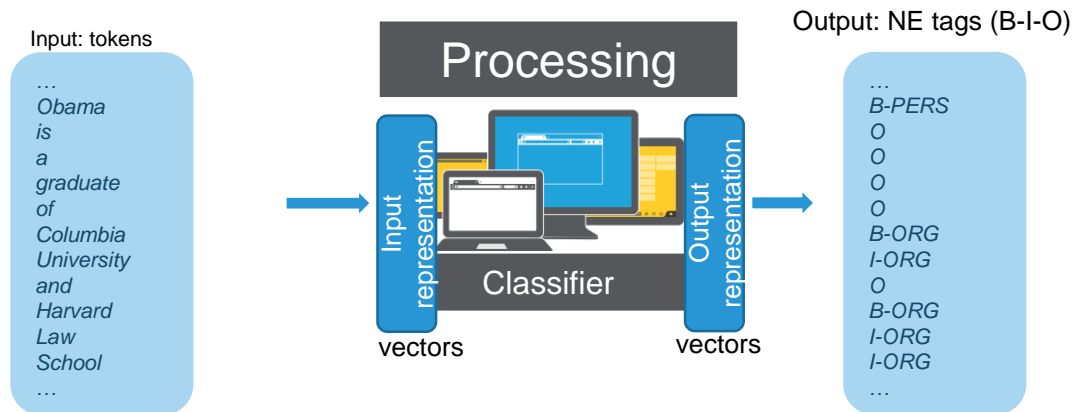
- Identification et extraction d'occurrences de noms de **personnes**, **lieux**, **organisations**, etc. dans les textes.
  - Les **dates** et certaines **données numériques** sont souvent considérées.
  - L'ensemble des types d'EN peut inclure des éléments spécifiques à un domaine, ex. noms de gènes en biomédical.
- Exemple:

***Barack Hussein Obama II** (born **August 4, 1961**) is the 44th and current President of the **United States**. He is the first African American to hold the office. **Obama** previously served as a **United States** Senator from **Illinois**, from **January 2005** until he resigned after his election to the presidency in **November 2008**.*

*A native of **Honolulu, Hawaii**, **Obama** is a graduate of **Columbia University** and **Harvard Law School**, where he was the president of the **Harvard Law Review**...*

# Text Mining Tasks as Classification Problems

## Reconnaissance d'entités nommées (REN)



**Classes:** O, B-PERS, I-PERS, B-ORG, I-ORG, B-LOC, I-LOC, etc...

## Extraction de relations entre EN

### Exemple

*Barack Hussein Obama II* (born *August 4, 1961*) is the 44th and current President of the *United States*. He is the first African American to hold the office. *Obama* previously served as a *United States* Senator from *Illinois*, from *January 2005* until he resigned after his election to the presidency in *November 2008*.

A native of *Honolulu, Hawaii*, *Obama* is a graduate of *Columbia University* and *Harvard Law School*, where he was the president of the *Harvard Law Review*...

### Relations

<i>Barack Hussein Obama II</i>	dateOfBirth	<i>August 4, 1961</i>
<i>Barack Hussein Obama II</i>	presidentOf	<i>United States</i>
<i>Obama</i>	placeOfBirth	<i>Honolulu, Hawaii</i>
<i>Obama</i>	graduatedFrom	<i>Columbia University</i>
<i>Obama</i>	graduatedFrom	<i>Harvard Law School</i>

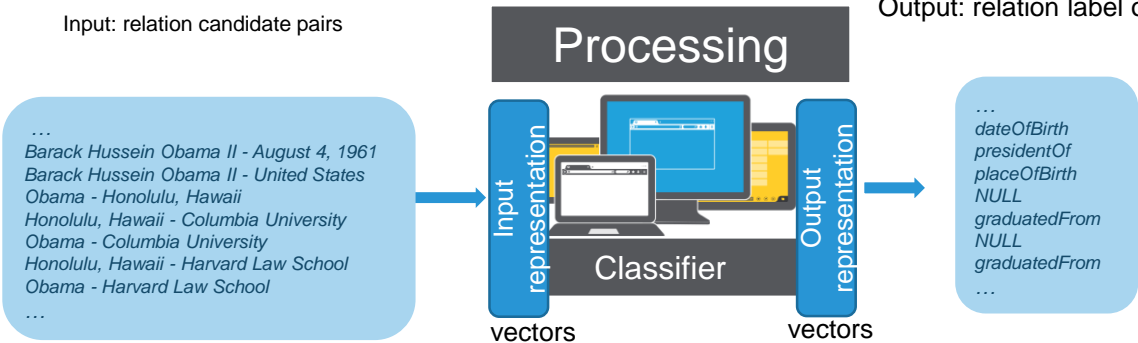
...

# Text Mining Tasks as Classification Problems

## Extraction de relations

Input: relation candidate pairs

Output: relation label or NULL



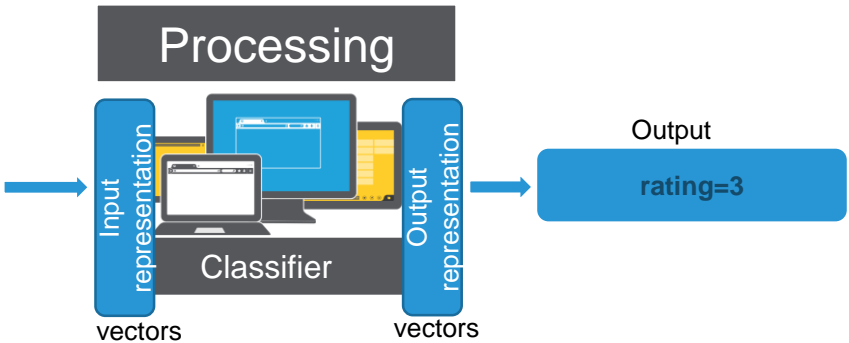
**Classes:** *NULL and the set of relation types*

# Text Mining Tasks as Classification Problems

## Sentiment analysis

Input: texts from restaurant reviews

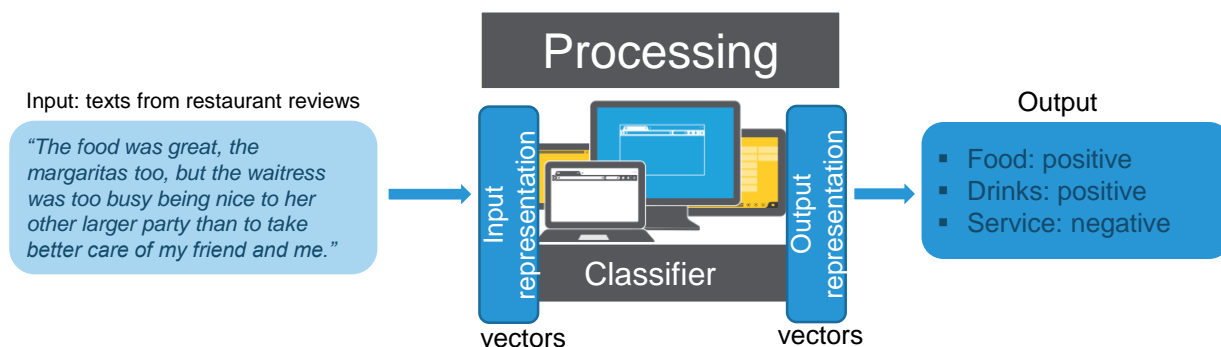
"The food was great, the margaritas too, but the waitress was too busy being nice to her other larger party than to take better care of my friend and me."



**Classes:** *rating=1, rating=2, rating=3, rating=4, rating=5*

## Text Mining Tasks as Classification Problems

### Aspect-based sentiment analysis (ABSA)



**Classes:** *Food:positive, Food:negative, Food:neutral, Drinks:positive, Drinks:negative, etc...*

S. Ait-Mokhtar, 25/09/2019 - !!!v0.8 – version incomplète et non définitive

Cours Fouille de textes, M2 MIASHS/SSD, Grenoble-Alpes

## Exploitation de la représentation

### Classification de textes/phrases/mots

- Le vecteur de représentation est l'entrée d'un classifieur (par ex. réseau de neurones)

### Regroupement (clustering)

- Calcul de la similarité/dissimilarité entre deux textes sur la base de la distance entre leurs vecteurs de représentation

### Recherche d'information / requête

- Calcul de la similarité/dissimilarité entre 1 requête et 1 document

### Distance entre les 2 vecteurs

- Exemple: distance euclidienne

$$d(X, Y) = \sqrt{\sum_{i=1}^n (y_i - x_i)^2}$$

S. Ait-Mokhtar, 25/09/2019 - !!!v0.8 – version incomplète et non définitive

Cours Fouille de textes, M2 MIASHS/SSD, Grenoble-Alpes

# Représentation vectorielle de textes

## Représentations creuses (approche traditionnelle)

- Vecteurs dont les valeurs correspondent à des traits (features)
- Les traits sont **prédéfinis et choisis explicitement** par le développeur du modèle
- Chaque indice du vecteur correspond à un trait
- En général, de nombreux traits auront la valeur 0 (d'où le qualificatif « creux »)

## Représentations denses / continues

- Vecteurs dont les valeurs correspondent à des traits (features) MAIS...
- Les traits sont **appris** (définis automatiquement lors de l'apprentissage)
- Les traits ont des valeurs continues (par exemple entre -1 et +1)

# Représentations creuses

- Traits **prédéfinis et choisis** par le développeur du modèle
- Exemples de traits
  - Mots d'un vocabulaire (le trait indique alors la présence ou l'absence de ces mots dans le texte)
  - N-grammes de mots
  - Traits typographiques (présence de chiffres, majuscules/minuscules, symboles de ponctuation, etc.)
  - Couples de mots liés syntaxiquement
  - Catégories sémantiques (présence de mots appartenant à ces catégories)
    - Exemple de catégories:
      - Nom de ville: { *Grenoble, Thiers, Madrid, Cologne, ...* }
      - Prénom: { *Marie, Alex, Ludovic, ...* }
      - Titre (civil, professionnel ou honorifique): { *Mme, M., Dr, Lieutenant, Maître, ...* }
      - Collectif: { *association, organisation, collectif, union, front, ...* }

## Représentation sac-de-mots (bag of words / BoW) (1)

Exemple de représentation creuse

Représentation de base : traits = mots ou termes

Les mots utilisés comme traits sont optionnellement:

- Normalisés (casse, lemmatisation et/ou racinisation)
- Filtrés: ignorer les mots grammaticaux/vides (*stopwords*)
  - articles, auxiliaires, prépositions, etc.

N-grammes

- Séquence de n mots contigus
- Généralement: bi-grammes ou tri-grammes

## Représentation sac-de-mots (bag of words / BoW) (2)

Valeurs possibles pour les traits

- Binaires: 1 ou 0 en fonction de la présence ou non du terme
- Valeur de fréquence (normalisée) du terme
- **Valeur TF-IDF**

Vecteur de taille (forcément) limitée = N

- Sélectionner les N traits les plus fréquents dans le corpus d'entraînement



## TF-IDF d'un terme $t$ pour un document $d$ (1)

TF-IDF = Term Frequency – Inverse Document Frequency

- Mesure de l'importance d'un terme dans un document relativement à une collection de document
  - Pondération de la fréquence brute

$$TF\text{-}IDF(t,d) = \mathbf{TF}(t,d) \times IDF(t)$$

- **$TF(t,d)$**  = Mesure de la « fréquence » d'un terme/mot  $t$  dans un document  $d$ 
  - Plusieurs variantes :
    - Binaire: 1 ou 0
    - Fréquence brute:  $f_t$
    - Valeur de fréquence normalisée:  $1 + \log(f_t)$
    - Valeur normalisée par le max:  $K + K (f_t / \max \text{ des fréquences de tous les termes})$ 
      - ( $K = 0.5$ )

## TF-IDF d'un terme $t$ pour un document $d$ (2)

TF-IDF = Term Frequency – Inverse Document Frequency

$$TF\text{-}IDF(t,d) = TF(t,d) \times \mathbf{IDF}(t,D)$$

- **$IDF(t,D)$**  = Mesure de la « spécificité » d'un terme dans une collection de documents  $D$ 
$$IDF(t,D) = \log( N / IDt )$$

$N$  = nombre de documents dans la collection  $D$

$IDt$  = nombre de documents dans  $D$  contenant le terme  $t$

- Plus un terme est spécifique à un petit nombre de documents, plus son score TF/IDF sera grand

## Construction du vecteur de traits (1)

Il faut d'abord préparer le vectoriseur

Phase de préparation du vectoriseur (fit)

- On segmente le texte d'entraînement en une séquence de tokens (tokenization)
- Optionnel: normalisation (mise en minuscule, lemmatisation/racinisation des mots, etc.)
- Optionnel: filtrage
- Construction du vocabulaire des mots  $V$ : index [mot  $\rightarrow$  indice]
  - Index associant à chaque mot un entier  $i$  entre 0 et  $|V|-1$  ( $|V|$  étant la taille du vocabulaire)

## Construction du vecteur de traits (2)

Vectoriseur préparé (vocabulaire  $V$  et index [mot  $\rightarrow$  indice] déjà construits)

Vectorisation d'un (nouveau) texte:

- Segmenter le texte d'entraînement en une séquence de tokens (tokenization)
- Optionnel: normalisation (mise en minuscule, lemmatisation/racinisation des mots, etc.)
- Optionnel: filtrage
- Créer un vecteur  $x$  de taille  $|V|$ , initialisé à 0
- Pour chaque token  $t$  du texte, s'il fait partie du vocabulaire du vectoriseur:
  - Assigner une valeur à  **$x[\text{token2ind}[t]]$** 
    - Valeur binaire, de fréquence etc.
  - Le token  $t$  correspond au trait à la position  **$\text{token2ind}[t]$**  du vecteur  $x$
- Retourner le vecteur  $x$

## Exercise: vectorisation binaire, par fréquence et TF/IDF

Problème: classification de phrases → texte = phrase

Représentation sac-de-mots:

1. Mots simples + filtrage des mots grammaticaux
2. Vocabulaire de (1) + bi-grammes

**Textes entraînement** (avis TripAdvisor):

*d1: Un accueil, un service, des plats, des vins, des desserts de qualité.  
d2: La carte évolue et les vins évoluent.  
d3: Les plats du jour sont souvent attractifs et même excellents.  
d4: Offre végétarienne tout à fait satisfaisante.*

**Textes à vectoriser** (avis TripAdvisor):

*Très bon accueil.  
Service rapide.  
Pizza excellente car la pâte est fine et croustillante.  
Tiramitsu fait maison.*

## Construction d'un vecteur de traits par hachage (1)

### Exemple

- SpeedyFX (Forman et Kirshenbaum, 2008)

```
boolean[] extractWordSet(text):
1 boolean fv[] = new boolean[N];
2 int wordhash = 0;
3 foreach (byte ch: text) {
4     int code = codetable[ch];
5     if (code != 0) { // isWord
6         wordhash = (wordhash>>1) + code;
7     } else {
8         if (wordhash != 0) {
9             fv[wordhash % N] = 1;
10            wordhash = 0;
11        }
12    }
13 }
14 return fv;
```

## Construction d'un vecteur de traits par hachage (2)

### SpeedyFX

- Construction de la table de test/normalisation de la casse

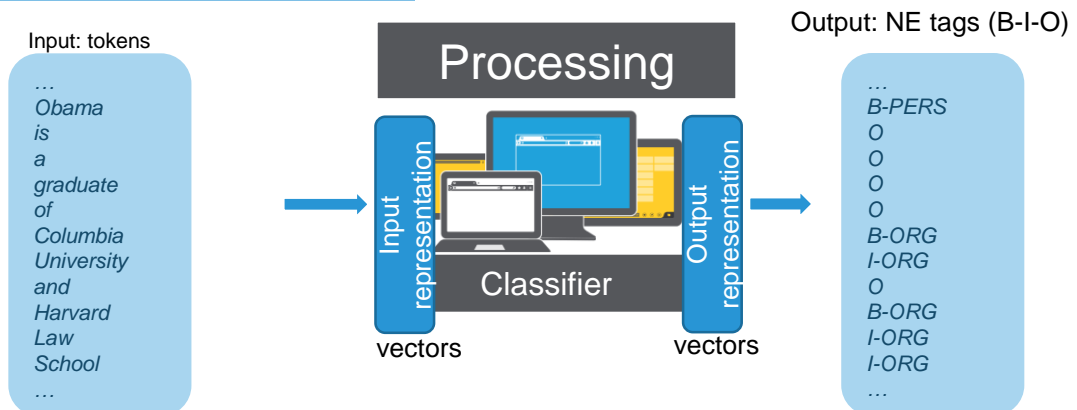
```
prepTable() :  
1 int rand[256] = 256 fixed random integers;  
2 foreach (ch: 0..255) {  
3     codetable[ch] = isWord(ch) ?  
4         rand[toLowerCase(ch)] : 0;  
5 }
```

### Extensions nécessaires pour

- L'encodage UTF8
- Exercice: inclusion de n-grammes (par ex. bigrammes)

## NER

### Reconnaissance d'entités nommées (REN)



**Classes:** O, B-PERS, I-PERS, B-ORG, I-ORG, B-LOC, I-LOC, etc...

# Reconnaissance d'entités nommées (REN)

- Identification et extraction d'occurrences de noms de personnes, lieux, organisations, etc. dans les textes.
  - Les dates et certaines données numériques sont souvent considérées.
  - L'ensemble des types d'EN peut inclure des éléments spécifiques à un domaine, ex. noms de gènes en biomédical.

• Exemple:

Barack Hussein Obama II (born August 4, 1961) is the 44th and current President of the United States. He is the first African American to hold the office. Obama previously served as a United States Senator from Illinois, from January 2005 until he resigned after his election to the presidency in November 2008.

A native of Honolulu, Hawaii, Obama is a graduate of Columbia University and Harvard Law School, where he was the president of the Harvard Law Review...

# Reconnaissance d'entité nommées (REN)

Elle rejoint début 1894 le laboratoire des recherches physiques de Gabriel Lippmann, au sein duquel la Société d'encouragement pour l'industrie nationale lui a confié des travaux de recherche sur les propriétés magnétiques de différents aciers. Elle y travaillait dans des conditions étroites et recherche donc une façon de mener à bien ses propres travaux. Le professeur Józef Kowalski de l'Université de Fribourg lui fait alors rencontrer lors d'une soirée Pierre Curie, qui est chef des travaux de physique à l'École municipale de physique et de chimie industrielles et étudie également le magnétisme, avec lequel elle va travailler.

Lors de cette collaboration se développe une inclination mutuelle entre les deux scientifiques. Marie rentre à Varsovie, pour se rapprocher des siens, et dans le but d'enseigner et de participer à l'émancipation de la Pologne, mais Pierre Curie lui demande de rentrer à Paris pour vivre avec lui. Le couple se marie à Sceaux, le 26 juillet 1895.

Extrait de l'article Wikipédia sur Marie Curie (date: 06/01/2014).

- Classification de chaque mot
- Notation B-I-O (début-intérieur-extérieur de la mention)
  - Vecteur de représentation pour chaque mot

...	
mais	O
Pierre	B-PERS
Curie	I-PERS
lui	O
demande	O
...	

## Application de l'apprentissage supervisé à la REN

### Donnée à classer

- **Chaque token** (mot) d'une **séquence** (car l'ordre, et donc le contexte, sont pertinents)

### Ensemble de classes possibles

- *O, B-PERS, I-PERS, B-ORG, I-ORG, B-LOC, I-LOC*

### Exemples pour l'apprentissage

...	...
Obama	B-PERS
is	O
a	O
graduate	O
of	O
Columbia	B-ORG
University	I-ORG
and	O
Harvard	B-ORG
Law	I-ORG
School	I-ORG
...	...

## Traits pour la REN (1)

### Traits locaux

- Traits de type BoW
  - Token actuel, tokens du contexte gauche et droit
  - N-grammes
- Traits liés à la forme
  - Mot en maj/min
  - Suffixes, préfixes
  - Motifs de normalisation des caractères (*word shapes*)
    - Marie → Xxxxx, 15/12/2014 → dd/dd/yyyy
- Traits extérieurs
  - Catégories syntaxiques, cluster de mots, etc.
  - Ressources encyclopédiques (DBPedia, Wikipedia, etc.)
    - Mot dans un nom de personne, organisation (entreprises, fédérations, équipes, groupes, etc.) lieu (pays, villes, provinces, lacs, etc.)

## Traits et performance en REN (1)

### Effets de certains traits sur les performances

- (Tkachenko et Simanovsky, 2012)

CoNLL-2003		
$w_0$	25.24%	22.04%
$w_{-1}, w_0, w_1$	83.41%	74.82%
$w_{-1}, w_0, w_1,$ $w_{-1} \& w_0, w_0 \& w_1$	81.20%	72.26%
$w_{-2}, w_{-1}, w_0, w_1, w_2$	82.31%	73.73%

Table 1: Evaluation of context in NER;  $w$  — token,  $a \& b$  — conjunction of features  $a$  and  $b$ .

CoNLL-2003		
$w_0$	25.24%	22.04%
$w_0$ + suffixes and prefixes	87.41%	78.59%
$w_0 + s_0$	86.70%	79.16%
$w_0 + s_{-1}, s_0, s_1,$ $s_{-1} \& s_0, s_0 \& s_1, s_{-1} \& s_0 \& s_1$	87.67%	81.37%
All Local Features	88.91%	82.89%

Table 2: Evaluation of local features in NER;  $w$  — token,  $s$  — shape,  $a \& b$  — conjunction of features  $a$  and  $b$ .

## Traits et performance en REN (2)

(Tkachenko et Simanovsky, 2012)

CoNLL-2003		
$w_0$ + Wikipedia gaz.	56.35%	53.98%
$w_0$ + Wikipedia gaz. + disambig.	84.73%	77.72%
$w_0$ + DBpedia gaz.	84.06%	75.40%
$w_0$ + DBpedia gaz. + disambig.	83.62%	75.14%
$w_0$ + Wikipedia & DBPedia gaz.	85.21%	78.16%

Tous les traits combinés :

CoNLL-2003		
All features	93.78%	91.02%

## Perceptron et REN (1)

- Une donnée (i.e. un token)  $x$  est représentée comme un vecteur

$$x = (x_1, x_2, x_3, \dots, x_n)$$

- Chaque dimension représente un attribut ou trait du token, supposé être pertinent pour la classification du token
- Chaque coordonnée dans une dimension représente la présence ou l'absence du trait pour ce token :
  - Valeurs des coordonnées
    - $x_i=1$  si le token ou mot possède le trait  $i$ ,
    - $x_i=0$  sinon
- $X$  : ensemble des données
- Un ensemble  $Y$  de  $m$  classes ou étiquettes possibles :  $\{c_1, c_2, c_3, \dots, c_m\}$
- Problème de classification :
  - A chaque token  $x$  de  $X$  on doit associer une étiquette  $c$  de  $Y$

## Perceptron et REN(2)

**Fonction  $f$**  : associe à chaque paire  $(x, c)$  (un mot et une étiquette quelconque) un nombre réel :

- $f(x, c) = w_c^T \cdot x$
- $w_c$  : vecteur de « poids » ( $w_{1c}, w_{2c}, w_{3c}, \dots, w_{nc}$ )
  - Poids (importance) de chaque trait par rapport à la classe  $c$  (biais inclus)

$$\begin{aligned} f(x, c) &= w_{1c} \cdot x_1 + w_{2c} \cdot x_2 + \dots + w_{nc} \cdot x_n \\ &= \sum_{i=1}^n w_{ic} \cdot x_i \end{aligned}$$

= la somme pondérée des valeurs de tous les traits pour le token  $x$

**Paramètres du modèle** : matrice des poids (Traits x Classes)



## Perceptron et REN (3)

### Prédiction linéaire:

- Pour chaque token  $x$ , choisir l'étiquette  $c_{res}$  qui conduit à une valeur maximale de la somme pondérée des valeurs de traits (biais inclu)

$$c_{res} = \operatorname{argmax}_c \sum_{i=0}^n w_{ic} \cdot x_i$$

- **Procédure** : pour chaque token  $x$  d'une phrase:
  1. Construire le vecteur de traits de  $x$
  2. Pour chaque étiquette  $c$  de l'ensemble des étiquettes:
    - Calculer  $f(x, c) = \sum_{i=0}^n w_{ic} \cdot x_i$
  3. Sélectionner l'étiquette avec la valeur  $f(x, c)$  maximale

Les poids des traits sont donc déterminants ! Comment les choisir ?

## Perceptron et REN (4)

### Calcul des poids des traits par apprentissage sur le corpus d'entraînement :

1. Initialisation: donner des valeurs aléatoires aux poids
2. Répéter (itérations)
  - Réordonner aléatoirement les phrases du corpus d'entraînement
  - Pour chaque phrase  $p$ 
    - Pour chaque token  $x$  de  $p$ , et son étiquette  $c_{ref}$  (étiquette de référence)
      - Construire le vecteur de traits de  $x$
      - Calculer  $c_{res} = \operatorname{argmax}_c \sum_{i=1}^n w_{ic} \cdot x_i$  (voir pages précédentes)
      - Si ( $c_{res} \neq c_{ref}$ ) : étiquetage incorrect ! Ajuster les valeurs des poids :
        - Pour chaque trait  $i=1, \dots, n$ , avec  $x_i \neq 0$ , mettre à jour les poids:
          - $w_{icref} \leftarrow w_{icref} + 1$
          - $w_{icres} \leftarrow w_{icres} - 1$

## Exercice : REN (reconnaissance d'entités nommées)

Considérer la phrase suivante:

*Marie Dubois part à Londres sur un vol Air France.*

Ensemble d'étiquettes morphosyntaxiques

O, B-PERS, I-PERS, B-ORG, I-ORG, B-LOC, I-LOC

### Tâches

1. Proposer un ensemble de 10 attributs (locaux et contextuels) pertinents pour la représentation des tokens en vecteurs de traits pour la REN
2. Représenter chaque token de la phrase 1 avec son vecteur de traits
3. En supposant que tous les poids des traits sont initialisés à 1, procédez à l'étiquetage des tokens de la phrase en appliquant la prédiction linéaire à base de perceptron
4. Appliquez l'algorithme d'apprentissage des poids du perceptron en supposant que le corpus d'entraînement contient seulement cette phrase (cas simplifié).

## Plan

- Fouille de textes: introduction
- Types d'applications en fouille de textes
- Principales étapes de traitements
- Introduction à quelques techniques de base
  - Hachage
  - Apprentissage automatique supervisé par réseaux de neurones artificiels
- Représentation vectorielle de textes/phrases/mots
- **Représentations continues**
- Reconnaissance d'entités nommées (REN)
- Fouille d'opinions

## Représentations continues

### Plongements (*embeddings*)

- Représentation d'éléments discrets avec des vecteurs denses
- Obtenus par analyse distributionnelle ou par rétro-propagation dans les modèles neuronaux
- Éléments représentés : mots, catégories syntaxiques, phrases, etc.

### Intérêt

- Représentations compactes
- Plus fine/robuste que la représentation classique (sac-de-mots, 1-en-N)
- Permet des calculs de similarités (mots reliés sémantiquement)

### Plongements lexicaux (*word embeddings*): plongements pour les mots

- Plusieurs modèles proposés, word2vec, Glove, Fasttext

## Plongements lexicaux (word embeddings)

### Pour représenter les mots avec des vecteurs denses

#### Souvent pré-entraînés:

- Obtenus par entraînement non supervisé sur de grandes quantités de textes bruts
- Utilisés ensuite pour des tâches spécifiques (par ex. REN, extraction de relation, fouille d'opinions...)
  - Figés (tels quels, sans modification pour la tâche cible)
  - Avec entraînement additionnel sur la tâche

#### Deux catégories principales

- Plongements statiques (non contextuels)
- Plongements contextuels

## Plongements lexicaux statiques (non contextuels)

### Plongements non contextuels

- Les vecteurs sont calculés lors de l'entraînement sur les textes bruts
- Ensuite, un mot aura toujours le même vecteur quel que soit son contexte
- Word2vec: <https://code.google.com/archive/p/word2vec/>
- Glove: <https://nlp.stanford.edu/projects/glove/>
- Fasttext: <https://fasttext.cc/docs/en/crawl-vectors.html> (modèles pré-entraînés disponibles pour 157 langues)

## Plongements lexicaux contextuels

### Plongements contextuels

- Le modèle est entraîné sur des textes bruts, mais ensuite
- Il permet d'associer à chaque mot un vecteur qui dépend du contexte du mot  
→ La représentation d'un mot dépend des autres mots de son contexte
- Exemples:
- Elmo (Embeddings from Language Models)
  - <https://github.com/HIT-SCIR/ELMoForManyLangs>
- BERT (Bidirectional Encoder Representations from Transformers)
  - <https://github.com/google-research/bert>

## Plongements lexicaux : ressources pour le TP

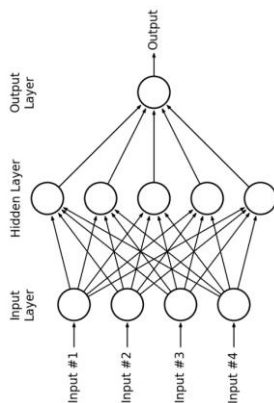
Librairies Python: gensim

Modèles word2vec pré-entraînés à télécharger

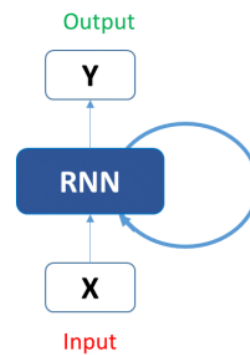
- Français
  - <https://github.com/Kyubyong/wordvectors>
- Anglais
  - <https://github.com/Kyubyong/wordvectors>

## Réseau de neurones récurrents (RNN)

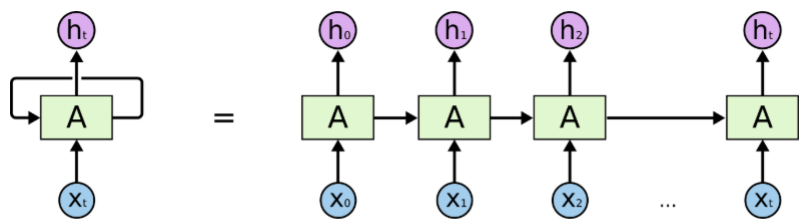
### Feedforward network / Multilayer perceptron (MLP)



### Recurrent Neural Network (RNN)

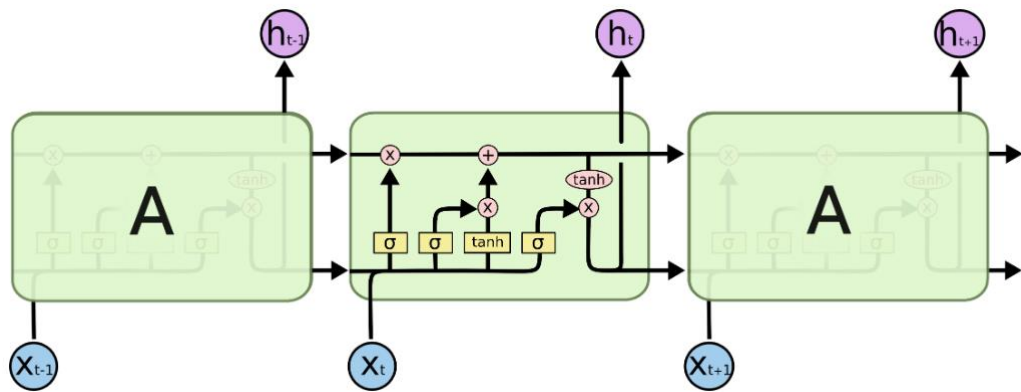


# RNN représenté en séquence



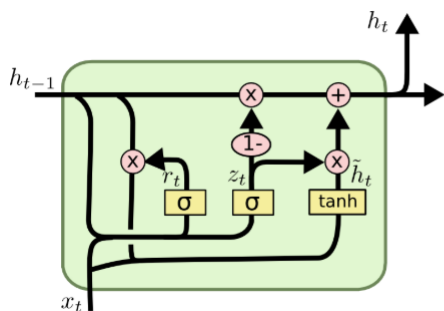
An unrolled recurrent neural network.

# Exemple de RNNs: les LSTMs



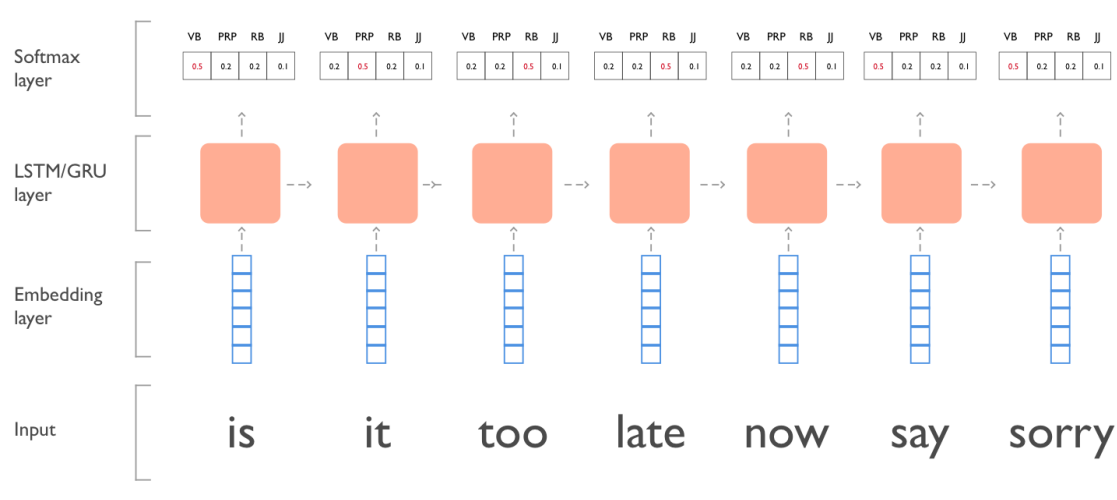
Source: <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>

# Exemple de RNNs: les GRUs



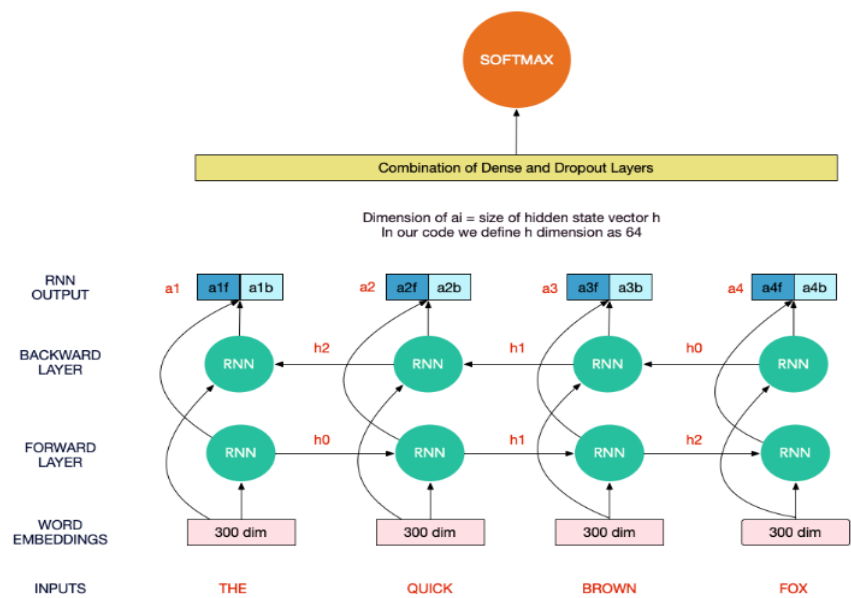
$$z_t = \sigma \left( W_z \cdot [h_{t-1}, x_t] \right)$$
$$r_t = \sigma \left( W_r \cdot [h_{t-1}, x_t] \right)$$
$$\tilde{h}_t = \tanh \left( W \cdot [r_t * h_{t-1}, x_t] \right)$$
$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

## RNN (LSTM or GRU)



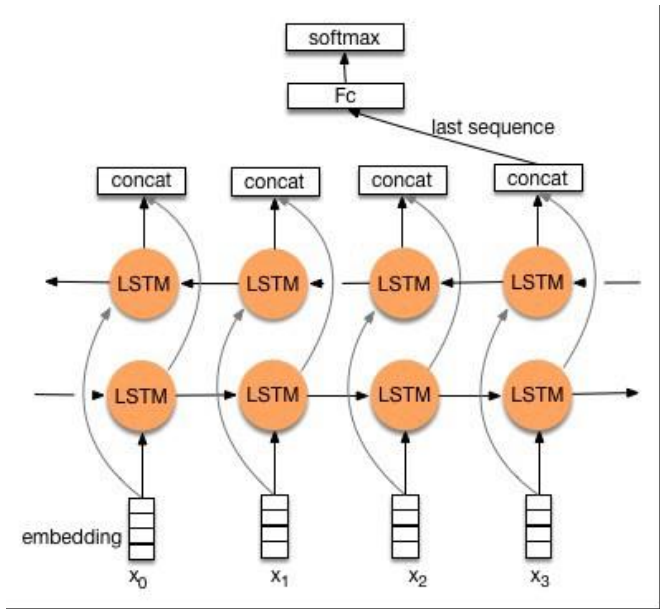
Source: <https://towardsdatascience.com/taming-lstms-variable-sized-mini-batches-and-why-pytorch-is-good-for-your-health-61d35642972e>

Bidirectional RNN



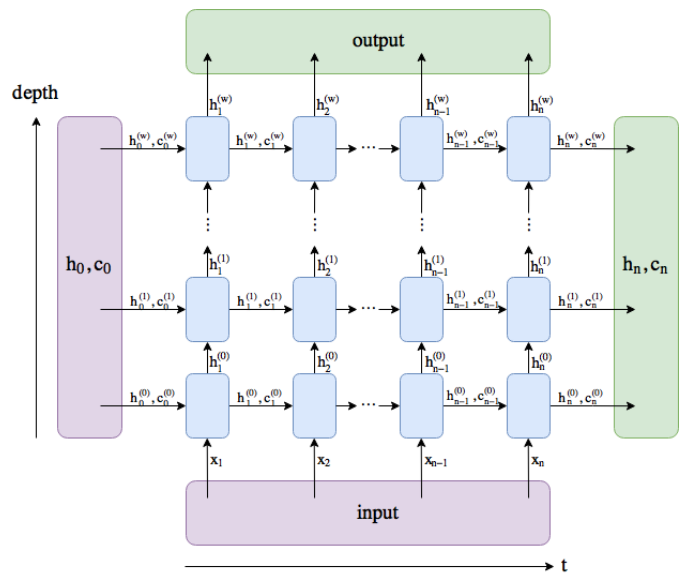
Source: [https://mlwhiz.com/blog/2019/03/09/deeplearning\\_architectures\\_text\\_classification/](https://mlwhiz.com/blog/2019/03/09/deeplearning_architectures_text_classification/)

BiLSTM





## LSTMs empilés (plusieurs couches)



## Représentation de mots à base de caractères

Utile pour les mots inconnus (absent du Vocabulaire d'entraînement):

- néologisme, noms propres, nombres, erreurs d'orthographe, etc.

