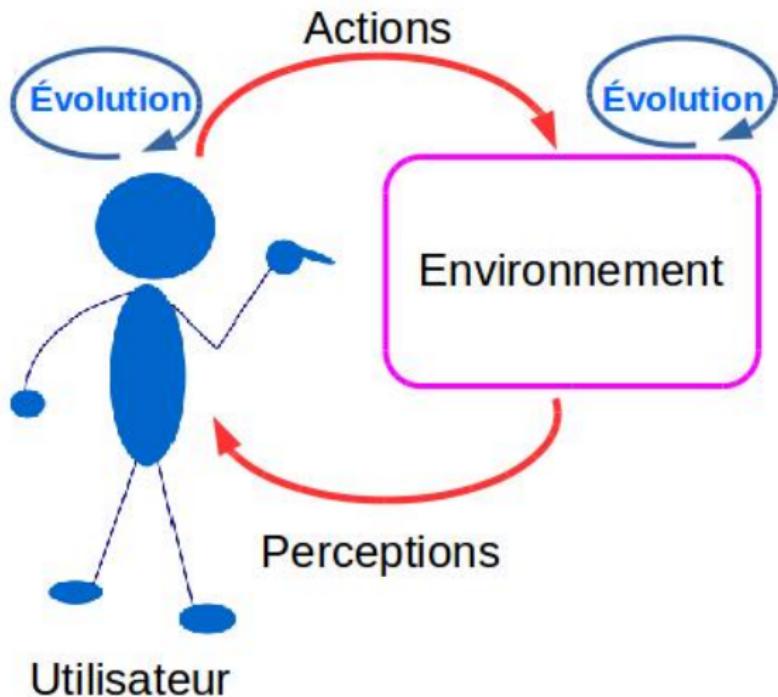


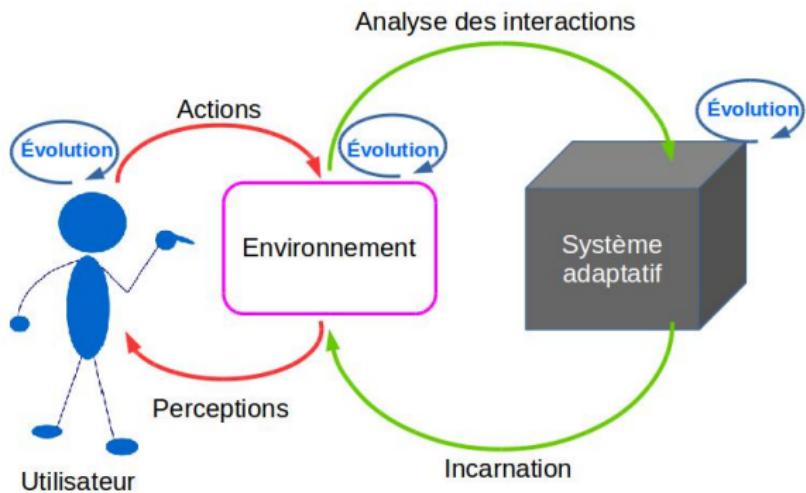
# Environnements adaptatifs

Automne 2022

# Adaptation



# Adaptation



## Source

Environnement, matériel, tâche, données, utilisateur

## Cible

Contenu, organisation, quantité d'information, procédure

## Temporalité

à la demande de l'utilisateur, selon l'état du système,  
périodiquement

# Auto-organisation de réseaux

# Auto-organisation de réseaux

## Objectif

- Exploration d'une base de données de type "DBpedia"
- $10^3$  à  $10^5$  articles
- Sait-on ce qu'on y cherche ?

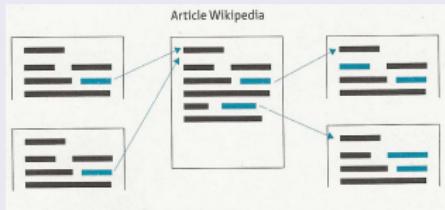
- Visualisation des résultats
- Structuration sémantique de la base

- Deux objets sémantiquement proches devront être spatialement proches
- Objet "Eglise" plus proche de l'objet "Chapelle" que de l'objet "Couteau"

# Auto-organisation de réseaux

## Les données

### Représentation métier



### Représentation textuelle

```
<XML>
<Link 1 />
<Link 2 />
<Link 3 />
...
</XML>
```

### Représentation orientée graphe / graphique



# Auto-organisation de réseaux

## Principe

### Formalisme basé sur les forces

- Modélisation des objets de la base par des masses ponctuelles associées à des objets 3d
- Objectif, situation → force

### Forces de répulsion

- Interaction entre objets très différents
- Amélioration de la lisibilité

### Forces d'attraction

- Interaction entre objets sémantiquement similaires

# Auto-organisation de réseaux

## Répulsion entre objets

- $i, j$  : 2 éléments
- $P_i, P_j$  : position des éléments

$\Delta \vec{v}_j = \vec{F}$  avec :

- $\vec{F} = k P_j \vec{P}_i$
- $k = 9\sigma s(\frac{1}{s+1} + \frac{s-3}{4})/d$
- $\sigma < 0$
- $s = (\frac{d}{r_0})^{\frac{1}{ramp}}$

# Auto-organisation de réseaux

Attraction entre objets liés

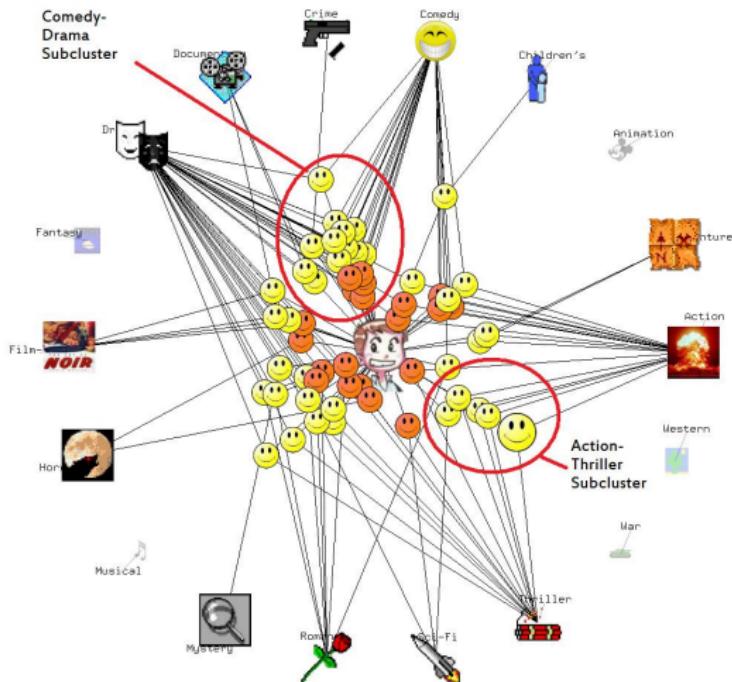
- $\Delta \vec{v}_j = 0.5 \vec{F}$
- $\Delta \vec{v}_i = -0.5 \vec{F}$

Avec

- $\vec{F} = \frac{1}{2} stiffness * (1 - damping) \vec{P_j} T$
- $T = P_i + \lambda \frac{\vec{P_i} \vec{P_j}}{||\vec{P_i} \vec{P_j}||}$

- damping : amortissement
- stiffness : rigidité

# Auto-organisation de réseaux



# Système auto-organisé énactif temps réel

# Système auto-organisé énactif temps réel

Objectif



## Question scientifique

- Présenter rapidement les informations pertinentes % à une problématique ?

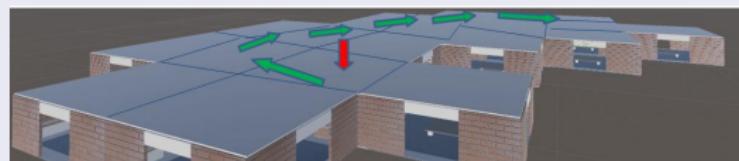
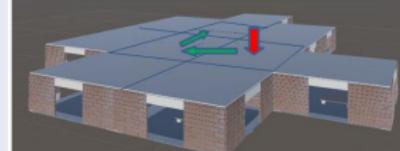
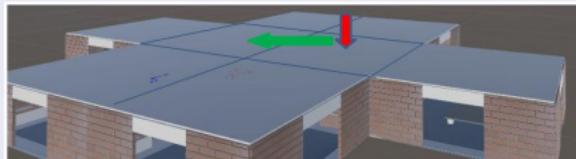
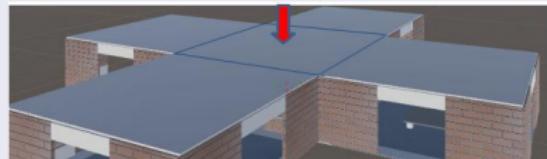
## Idée

Combinaison de :

- Métaphore 3d (musée virtuel)
- Co-construction, auto-organisation

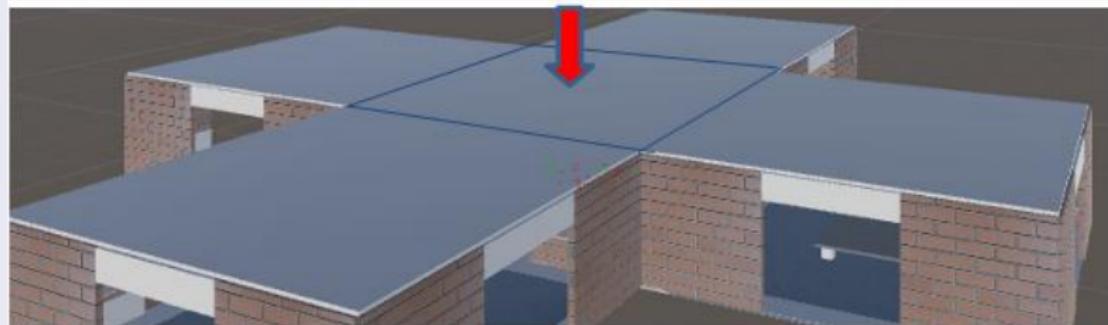
# Système auto-organisé énactif temps réel

## Exemple



# Système auto-organisé énactif temps réel

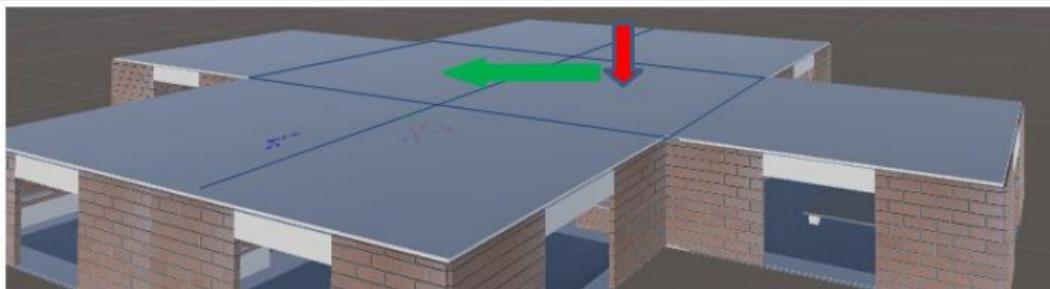
Exemple : étape 1



- Utilisateur dans la salle vide  $S_0$
- L'utilisateur interagit avec les mots disponibles :
  - le système attribue comme thème à  $S_0$  le mot "Seigneur"
  - le système attribue comme thèmes aux salles voisines les mots "Tour", "Eglise", "Pierre"

# Système auto-organisé énactif temps réel

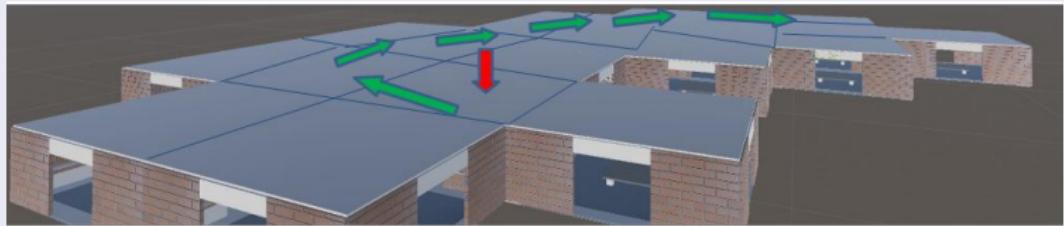
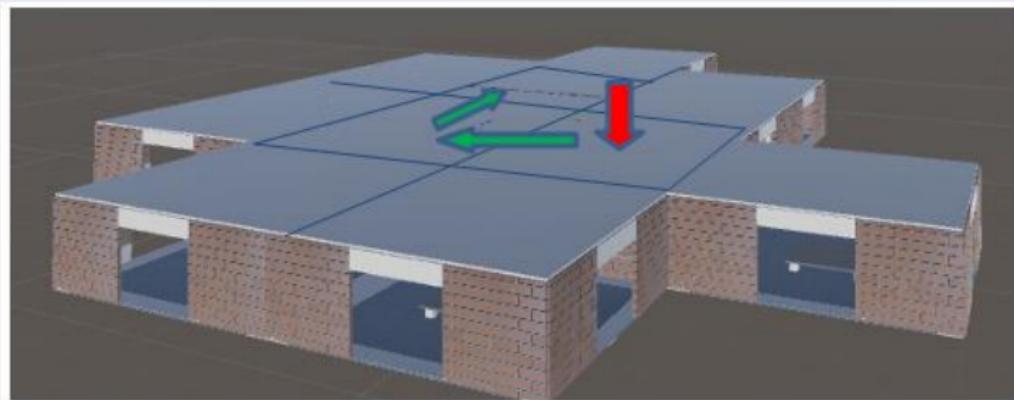
Exemple : étape 2



- L'utilisateur se déplace dans la salle ayant le thème "Eglise"
- Aussitôt une salle adjacente est créée, dotée du thème "Bâtiment"
- L'utilisateur entre dans cette nouvelle salle

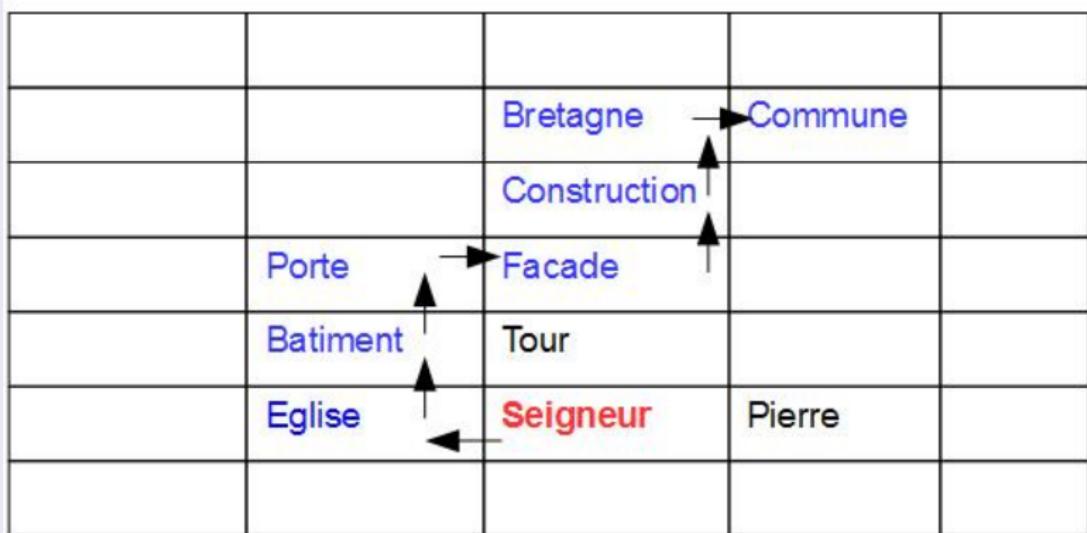
# Système auto-organisé énactif temps réel

Exemple : étape 3 et ultérieures



# Système auto-organisé énactif temps réel

Contexte créé



# Système auto-organisé énactif temps réel

## Vues de l'environnement



### Objets visualisés

- Salles connectées NS/EO
- Images des objets sur étagères
- Mots-clés en sur-impression

### Interactions

- Navigation 3d
- Sélection d'objets
- Sélection de mots

- Quelles salles ?
- Quels objets

# Système auto-organisé énactif temps réel

## Données

- Matrice de salles
- Graphe d'association entre mots-clés
- Graphe d'association

# Système auto-organisé énactif temps réel

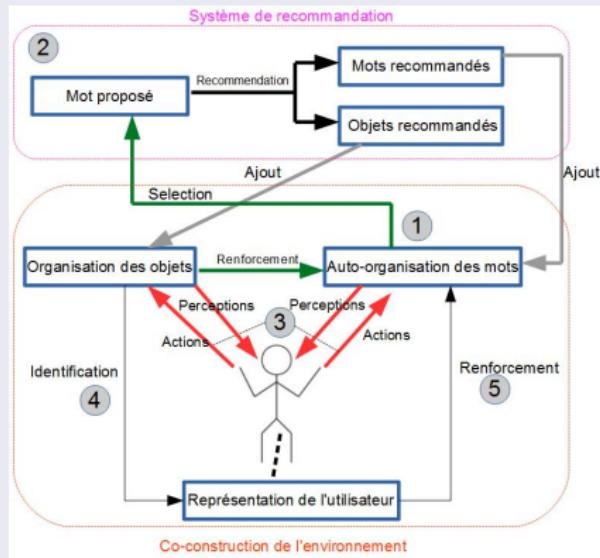
## Choix des salles

- assurer la continuité des liens contextuels
  - évolution de la construction selon les déplacements de l'utilisateur
- 
- score des salles : 0 : moyen, 1 : fort, 2 : très fort
  - salle retenue : parmi les meilleurs scores

salles voisines occupées	présence util. voisinage	score
> 0	oui	2
= 0	oui	1
> 0	non	0

# Système auto-organisé énactif temps réel

## Choix des objets



# Système auto-organisé énactif temps réel

## Les mots

### Espace des mots

- Espace inclus dans l'envt 3d
- Au plus N incarnations de mots
- 2 "points" marqueurs : intérêt ( $M^+$ ) / désintérêt ( $M^-$ )

### Mots

- Mots incarnés, situés
- Pertinence  $P(w)$

$$P(w) = \frac{a}{1 + D(w)} + \frac{b}{1 + A(w)} + cI(w)$$

- $D(w)$  : distance de  $w$  au marqueur  $M^+$
- $A(w)$  : durée de présence de  $w$  dans l'espace des mots

# Système auto-organisé énactif temps réel

Les mots : comportements

## Classement

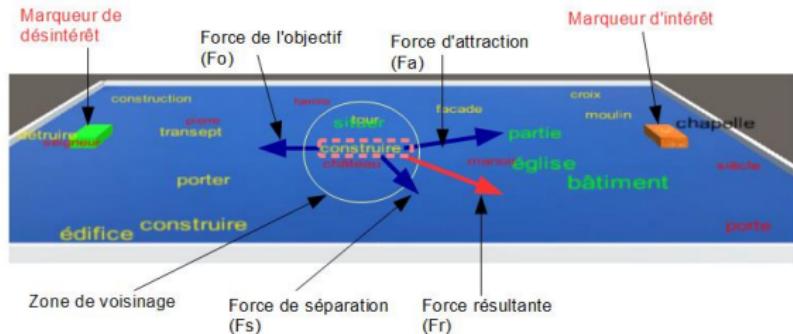
- $S_k$  : ensemble des mots les plus pertinents ( $N/3$  élts)
- $M_k$  : ensemble des mots moyennement pertinents ( $N/3$  élts)
- $W_k$  : ensemble des mots peu pertinents ( $N/3$  élts)

## Comportement

- $w \in S_k$  :  $w$  attiré par  $M^+$ , repoussé par les autres mots
- $w \in W_k$  :  $w$  attiré par  $M^-$ , repoussé par les autres mots
- $w \in M_k$  :  $w$  attiré par les mots qui l'ont produit

# Système auto-organisé énactif temps réel

Les mots : comportements



# Système auto-organisé énactif temps réel

Les mots : dynamique des pertinences

## Comportement asynchrone (sur sélection d'un objet $o$ )

- $V^+(o)$  : mots qualifiant  $o$
- **Contribution** :  $C = \sum_{w \in K} \alpha I(w)$
- **Redistribution** :  $R = \frac{C}{|w / w \in V^+(o)|}$

## Comportement synchrone (périodiquement)

- $\bar{I} = \frac{1}{|K|} \sum_{w \in K} I(w)$
- **Contribution** :  $C = \sum_{w / I(w) > \bar{I}} \beta(I(w) - \bar{I})$
- **Redistribution** :  $R = \frac{C}{|w / I(w) < \bar{I}|}$

# Anticipation de services dans un monde ouvert

# Anticipation de services dans un monde ouvert

## Objectif

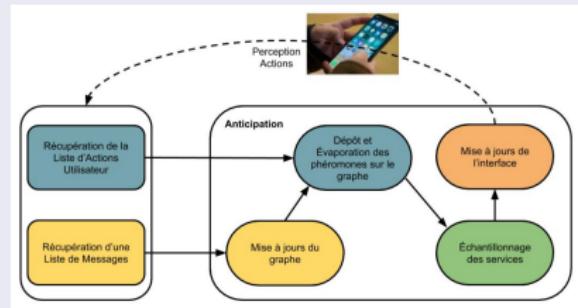
The image shows a mobile application interface with a light blue header and a white background. At the top left, there is a small icon labeled "Graph". Below the header, there are several service cards arranged in a grid-like structure:

- Click A Service**:
  - LISTE @BazinKaren
  - LISTE @AurelMa29
  - LISTE @LeTelegramme
  - LISTE @lemondefr
  - LISTE @JulienTocanier
  - LISTE @CMBretagne
  - LISTE @cmarkea
  - LISTE @OFCenterprises
  - LISTE @BazinKaren
  - LISTE @Marie22500
- CLOUD #Arkea17mai**:
  - @AurelMa29 @20217Gaby @LeTelegramme
  - #Arkeaindependant**
  - @JulienTocanier @Didden @KATHANGUYEN @cmarkea @JaimeCMArkea @olivier\_dream
- LISTE : #Arkeaindependant**:
  - @MarylineBilobro Une grande émission INCOLABLE Organisée par un collectif INCROYABLE Au milieu d'une marée rouge INCONTournable ... <https://t.co/LKzskgWvN>
  - @emilie\_bolleau VIDEO A la von IT Super montage by @HighSchoolMusic avec le son ♪ pour se redonner le rythme
  - @Pauline\_Freton @Arkeaindependant <https://t.co/mmeCRXrjK>
- CLOUD @cmarkea**:
  - @BazinKaren
  - @JulienTocanier
  - @OuestFrance @Didden
  - @LeTelegramme @Arkea17mai
  - @JaimeCMArkea @JaimeArkea
  - #Arkeaindependant**
  - @BonnetRougeBZH
  - @Paddy2411
- fake love #fakelovefriday**:
  - the (butcher) website AT3Denergy@fakelovefriday labo
  - Wysyłanie wiadomości i rekomendacji
  - praca z klientami
  - planowanie
- #Arkea17mai**:
  - @BonnetRougeBZH
  - @cmarkea
  - @JulienTocanier
  - #Arkeaindependant**
  - @JaimeCMArkea
  - @BazinKaren
- State your request**:
  - Arkea
  - 
  - Search

# Anticipation de services dans un monde ouvert

## Principe

### Schéma de principe



### Processus parallèles

- Construction des relations entre méta-données
- Estimation du profil utilisateur

&gt;

- Echantillonnage des services

# Anticipation de services dans un monde ouvert

## Structuration des mots-clés

### Principe

- Pouvoir avancer les tags les plus "proches" d'un tag
- Deux tags proches dénotent des concepts proches
- Comparer des tags afin de pouvoir calculer des "distances" entre eux

### Idée fondatrice

- Postulat de base pour trier les tags entre eux : plus ils ont un sens proche plus ils sont cités ensemble
- Approche statistique

### Inspiration

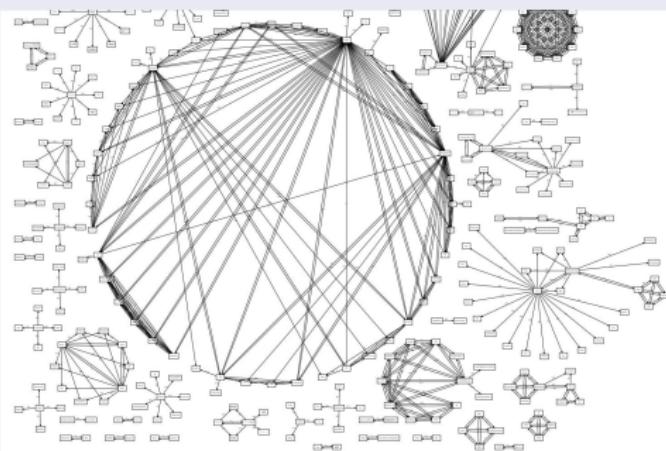
- Growing Neural Gaz : graphe pouvant se contracter et s'étirer afin de remplir un espace
- Application à un modèle de co-occurrences entre étiquettes

# Anticipation de services dans un monde ouvert

## Structuration des mots-clés

### Graphe de co-occurrence

- $G = (X, \Gamma, v)$  , avec
  - $X$  : ensemble des tags
  - $\Gamma$  : relation de co-occurrence
  - $v(x_1, x_2)$  : représentation de la co-occurrence entre  $x_1$  et  $x_2$



# Anticipation de services dans un monde ouvert

## Structuration des mots-clés

Mise à jour de  $v$

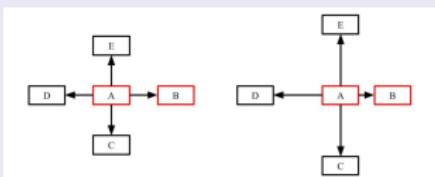
Réception du message  $m$  de tags  $T(m)$

Contraction (2 termes similaires se rapprochent)

$$x_1 \in T(m) \wedge x_2 \in C(m) \rightarrow v(x_1, x_2) := \min(1, v(x_1, x_2) + \alpha)$$

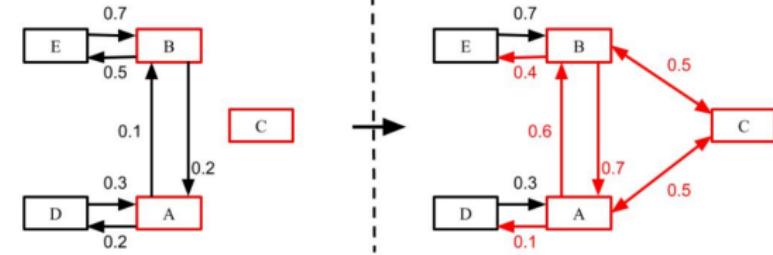
Dilatation (2 termes différents s'éloignent)

$$x_1 \in T(m) \wedge x_2 \notin C(m) \rightarrow v(x_1, x_2) := \beta v(x_1, x_2)$$



# Anticipation de services dans un monde ouvert

## Structuration des mots-clés

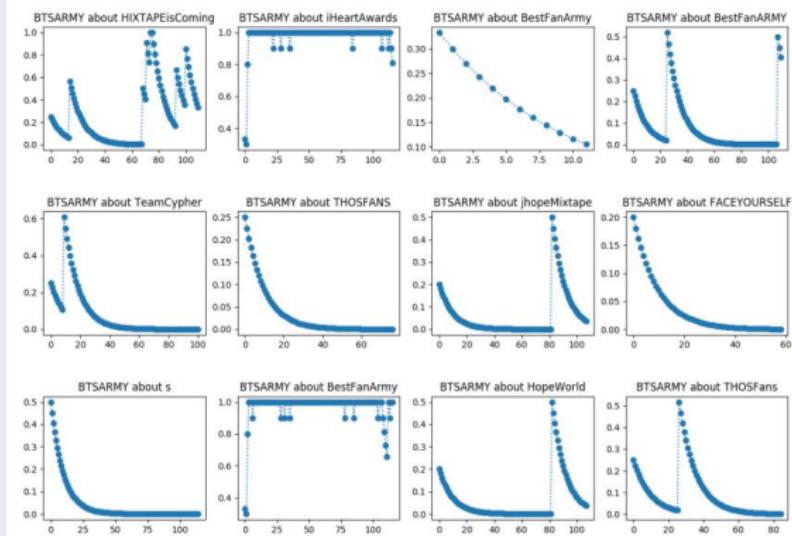


Modification des poids quand  $A$ ,  $B$  et  $C$  sont cités, avec

- $\alpha = 0.5$
- $\beta = 0.9$

# Anticipation de services dans un monde ouvert

## Structuration des mots-clés



Lien de corrélation entre un noeud et chacun de ses voisins tout au long de l'enrichissement du modèle à partir d'une base fixe de tweets

# Anticipation de services dans un monde ouvert

## Diffusion de phéromones

### Dépôt de phéromones

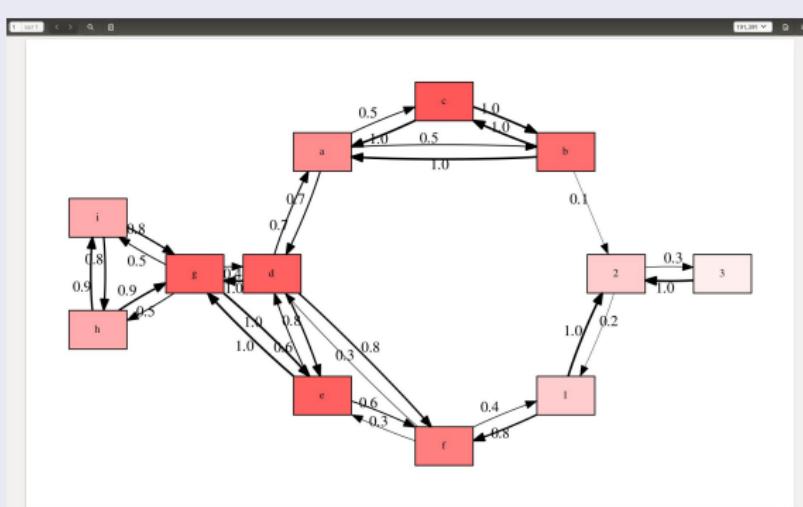
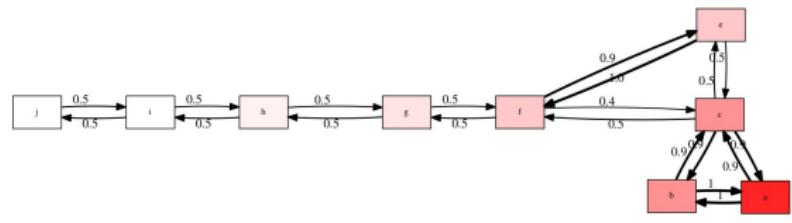
- $\Delta\Phi_0$  : quantité de phéromones injectée en  $n_0$
- $A$  : arbre couvrant  $G$  de racine  $n_0$
- $\Delta\Phi(m) = v(n, m)\Delta\Phi(n)$  avec  $n$  le père de  $m$  dans  $A$

### Evaporation de phéromones

- $\Phi_{t+1}(n) = \alpha\beta\Phi_t(n)$ ,  $\alpha \in [0, 1]$   $\beta \in [\sigma > 0, 1]$ , aléatoire

# Anticipation de services dans un monde ouvert

## Diffusion de phéromones



# Anticipation de services dans un monde ouvert

## Echantillonnage

### Services paramétrables

2 services paramétrables par des mots-clés :

- Liste : une dizaine de tweets avec une thématique commune (même mot-clé ou même auteur)
- Cloud : présente un nuage de mots-clés sélectionnés parmi les voisins les plus corrélés avec le mot-clé paramètre

### Echantillonnage

- $I(o) \sum_{n \in V(o)} \phi(n)$
- Echantillonnage : loterie basé sur l'intérêt

# Adaptation de règles pour une maison intelligente

# Adaptation de règles pour une maison intelligente

## Objectif

Maison autonome / intelligente

Pour

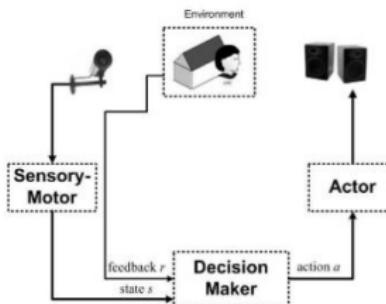
- Sécurité (dommage matériels, corporels, ...)
- Economie (consommation énergétique, ...)
- Confort (éclairage, son, température, média, ...)

Programmation

- a priori (impossible)
- par un technicien (ressource trop rare)
- par l'utilisateur final (trop complexe)
- par machine learning, approche supervisée (très difficile)
- par apprentissage interactif, en continu ( ... )

# Adaptation de règles pour une maison intelligente

## Principe



## Règles

Dans une situation  $s$   
engager une action  $a$   
afin de maximiser la satisfaction  $r$  de l'utilisateur

## Approche

- Adaptation du corpus de règles par renforcement (Q-learning)
- Interaction avec l'utilisateur final



# Adaptation de règles pour une maison intelligente

## Les actions (a)

- Vecteur de dim 6

- Music(genre, mood, volume)
- Lighting(pattern, color, brightness)

Service	Features	Values
Music	Genre	Blues, Christian, Classical, Country, Dance, Electronic, Folk, Hip_Hop, Holiday, Jazz, Latin, New_Age, Oldies, Pop, R_n_B, Reggae, Rock
	Mood	Calm, Exciting, Happy, Neutral, Sad
	Volume	Silence, Low, Medium, High
Lighting	Pattern	Cloudy, Starry, Fireworks, Solid
	Color	Purple, Blue, Gray, Green, Orange, Red, Yellow
	Brightness	Bright, Medium, Dark, Off

- vecteur action → musique, effets lumineux
- adaptation appliquée indpt par feature (réduction espace recherche)



# Adaptation de règles pour une maison intelligente

## Les situations (s)

### Etat / Situation

- State(temporal-slot, activity, location)
- location : kitchen, dining-table, tv-area
- activities : standing, sitting, lying, walking, eating, studying, watching
- time : intervalles de temps

# Adaptation de règles pour une maison intelligente

## Les retours utilisateur ( $r$ )



### Feedback

- direct ou indirect
- retour encourageant ( $r \geq 0$ ) ou pénalisant ( $r < 0$ )

### Mesure pour le feedback indirect

$$r = \frac{A}{B} - 1$$

- Action en cours : morceau de musique, effet lumineux
- $A$  = durée avant interruption de l'action
- $B$  = durée totale prévue de l'action

# Adaptation de règles pour une maison intelligente

## Adaptation

```
initialisation/normalisation de Q(s,a)
a = choixAleatoireAction()

when changementEtat ou interruptionAction :
    r = feedbackUtilisateur()
    s1 = nouvelEtatObservé()

    if random() < epsilon :
        a1 = choixAleatoireAction()
    else :
        a1 = argMax(Q(s1,_))

    Q(s,a) = Q(s,a) + alpha*[r + gamma*Q(s1,a1)- Q(s,a)]
    normalisationQ()

    a = a1, s = s1
```

# Adaptation de règles pour une maison intelligente

## Adaptation

- Pas de modèle utilisateur explicite a priori
  - Mécanisme de compromis exploitation / exploration
- 
- Choix des valeurs des paramètres (alpha : 0.2, epsilon : 0.002 avec une décroissance de 0.9 / slot)
  - Temps pour atteindre la convergence

