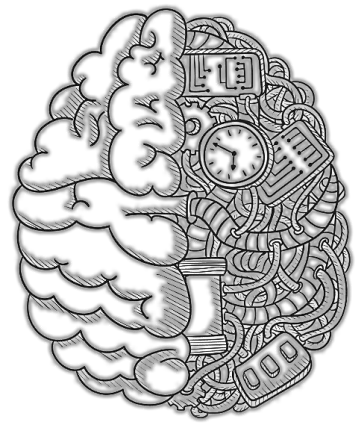
Réalisé par : yahya akli

Page de garde

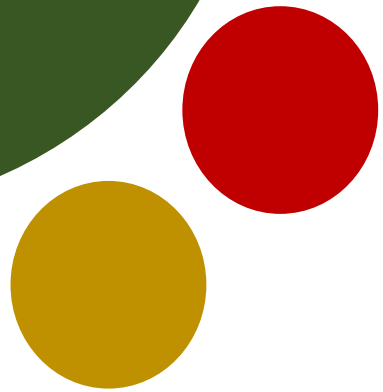


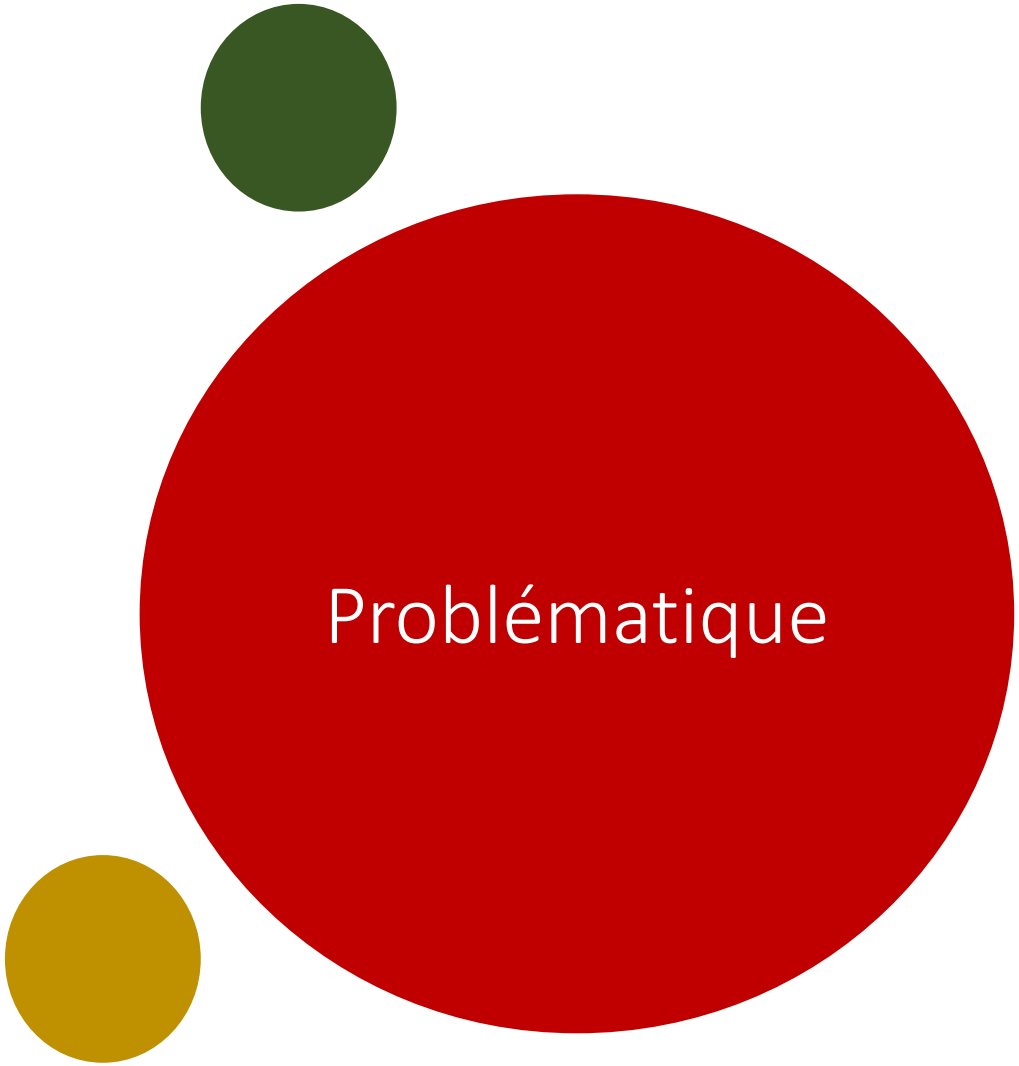
interface homme-machine

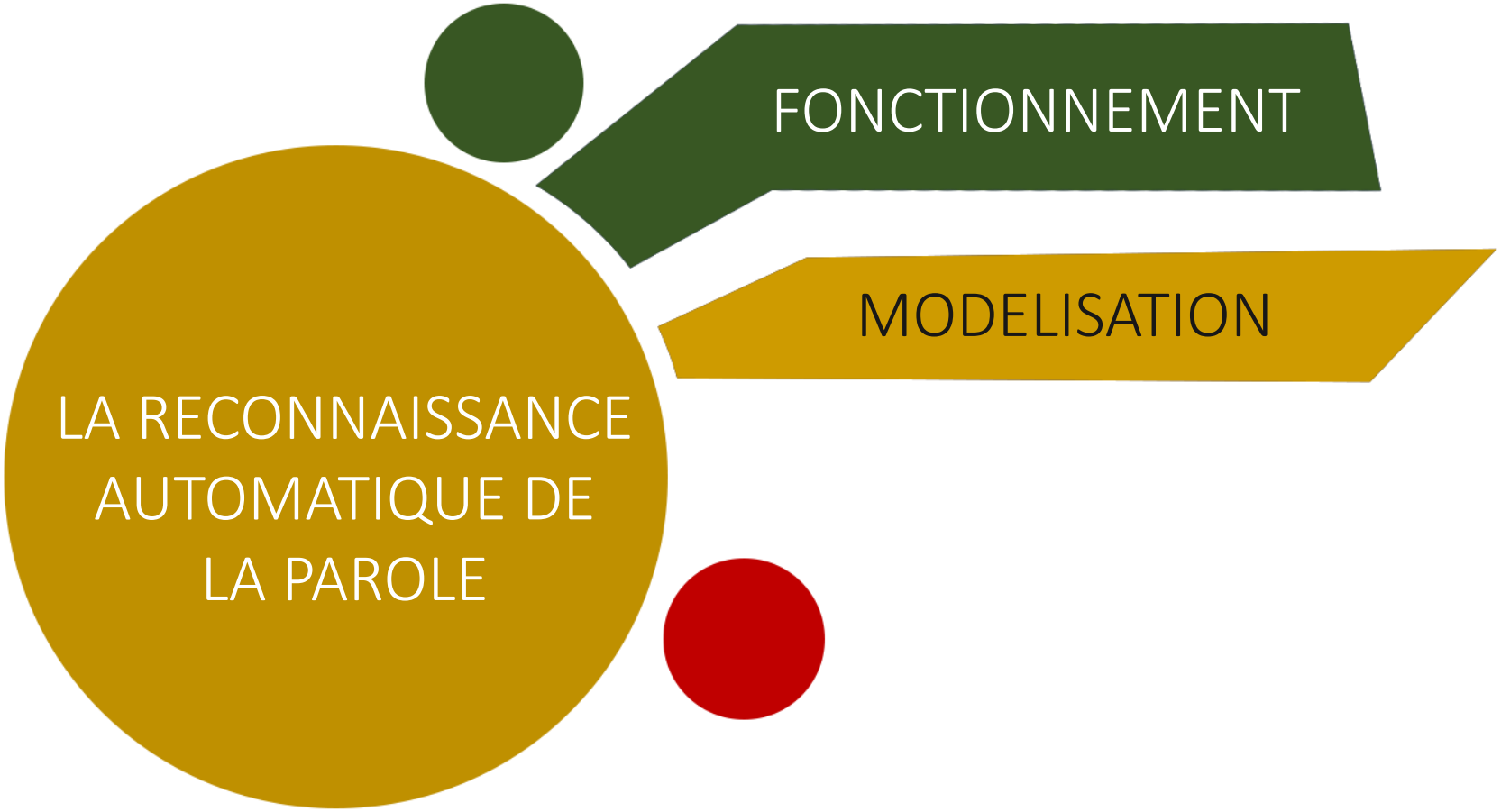
Améliorer l'interface homme machine à travers la reconnaissance vocale

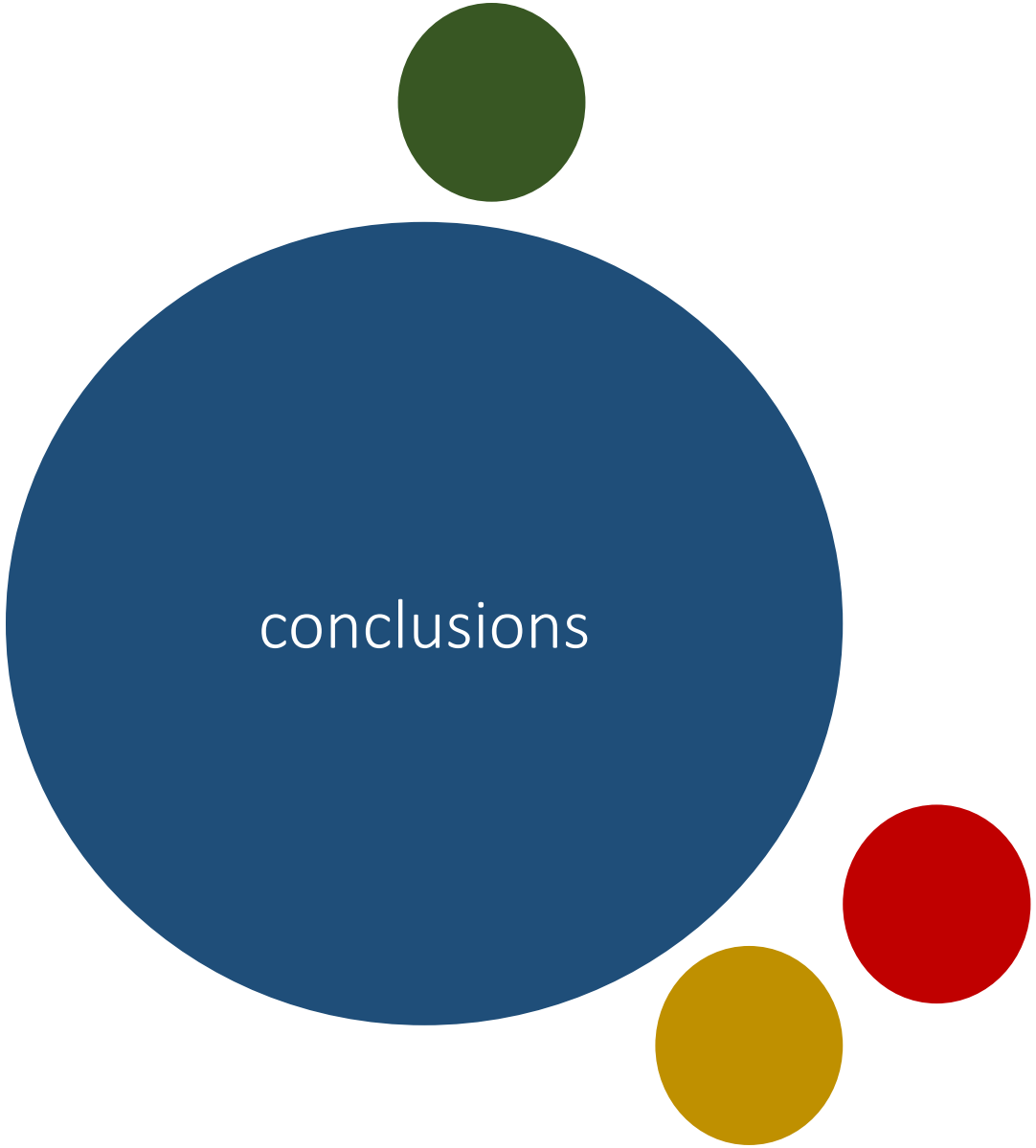
SOMMAIRE

REPRESENTATION GENERAL




Problématique


FONCTIONNEMENT
MODELISATION
LA RECONNAISSANCE AUTOMATIQUE DE LA PAROLE


conclusions

Représentation générale de l'étude

Aussi vielle que l'informatique, l'histoire de l'interaction homme-machine est marquée d'une péripétie d'évènements résultants de la volonté d'optimiser et faciliter l'utilisation de ses interfaces

Modes d’interaction

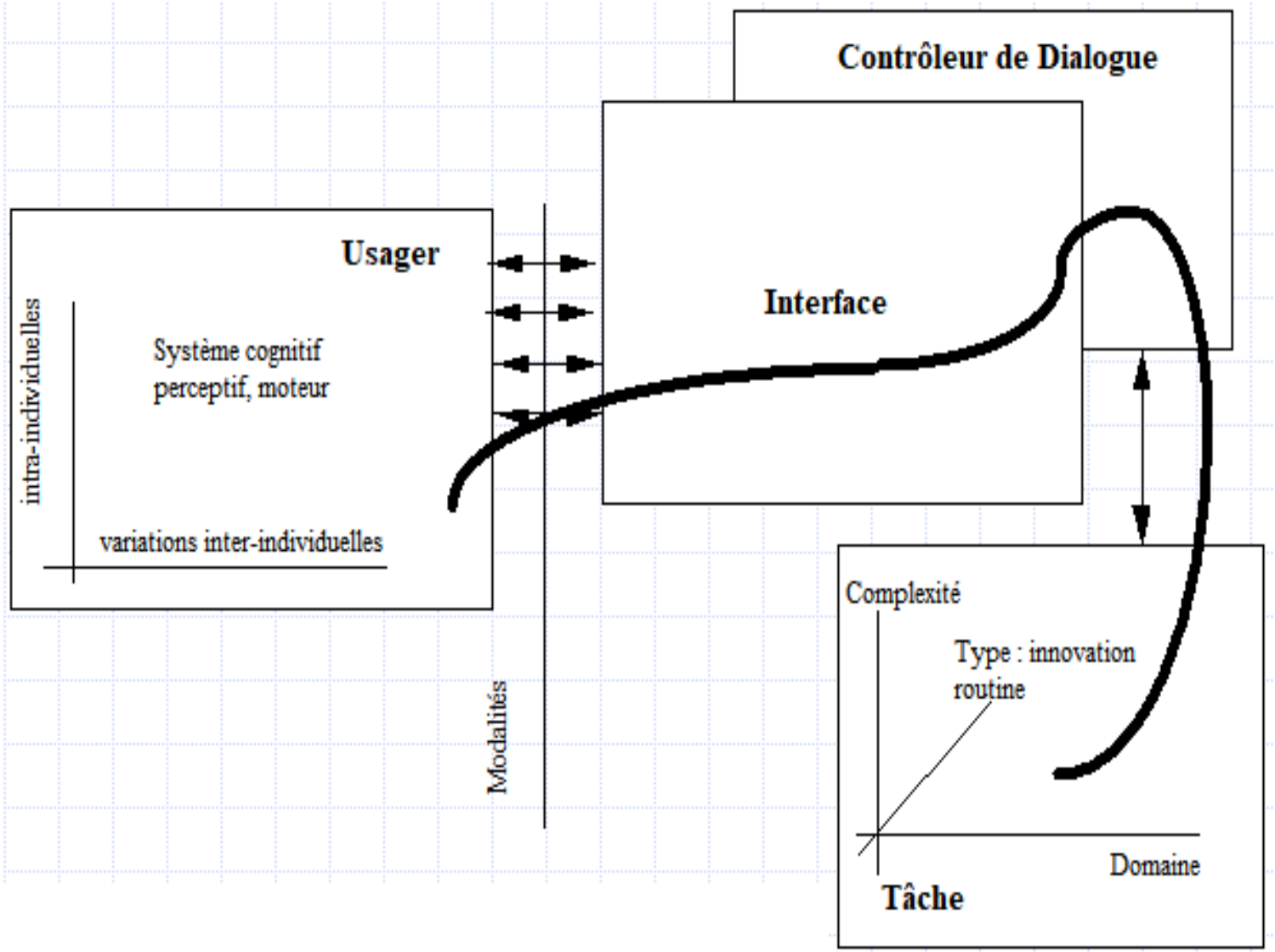
Mode parlé

Mode écrit

Mode gestuel

Mode visuel

Fig. 1 - Mode de fonctionnement de l'IHM



Problématique

La reconnaissance automatique de la parole, facilite-t-elle l'interaction avec la machine?

La reconnaissance automatique de la parole

Introduction

le traitement automatique des langues est un vaste domaine et champs de manœuvre économiquement porteur, dont les applications sont multiples

bureautique

l'aide aux handicapés

enseignement

Définition

technique informatique qui permet d'analyser la voix humaine captée au moyen d'un microphone pour la transcrire sous la forme d'un texte exploitable par une machine.

Historique

1950 - début

1952 - reconnaissance des 10 chiffres

1960 - utilisation des méthodes numériques

1971 - lancement du projet ARPA

Historique

1972 - premier appareil commercialisé de reconnaissance de mots

1983 - première mondiale de commande vocale à bord d'un avion de chasse en France

2008 - Google lance une application de recherche sur Internet

2011 - Apple propose l'application Siri sur ses téléphone

Historique

2017
Microsoft annonce égaler les performances de reconnaissance vocale des êtres humain

Marché

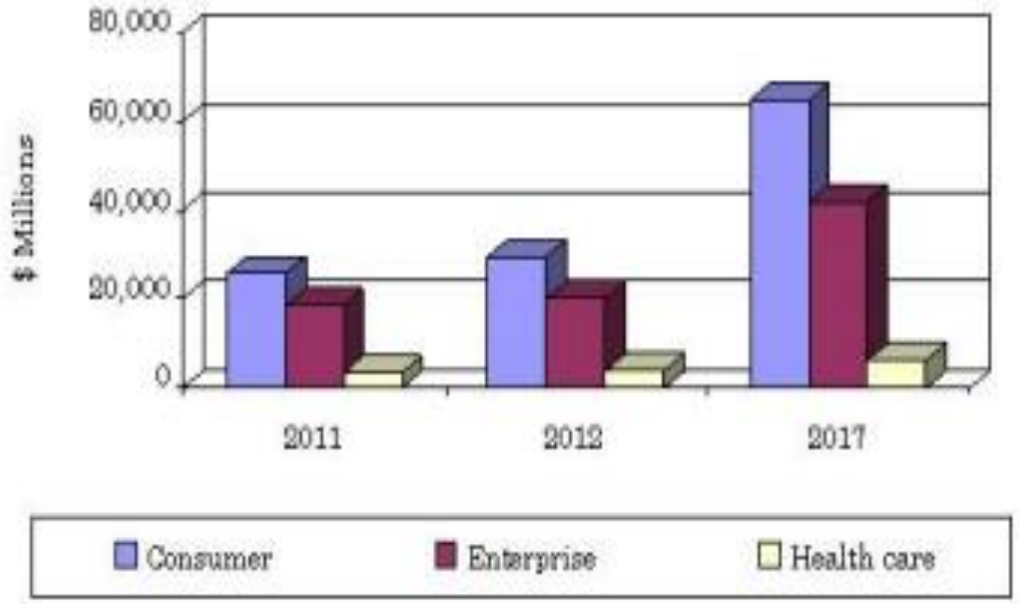
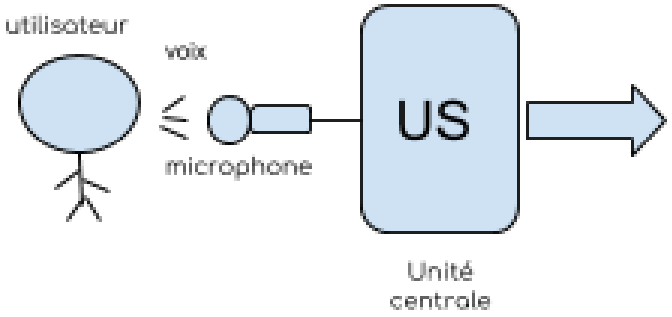
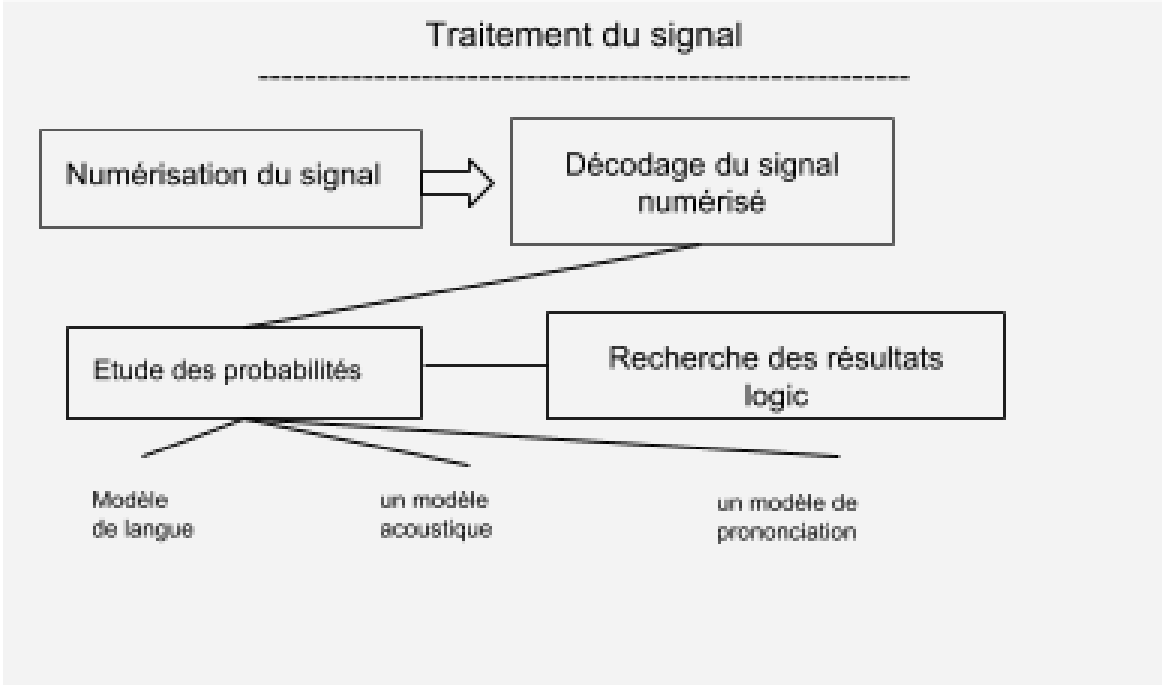


Fig.2 – la distribution du marché de la reconnaissance vocale en 2011, 2012 et 2017

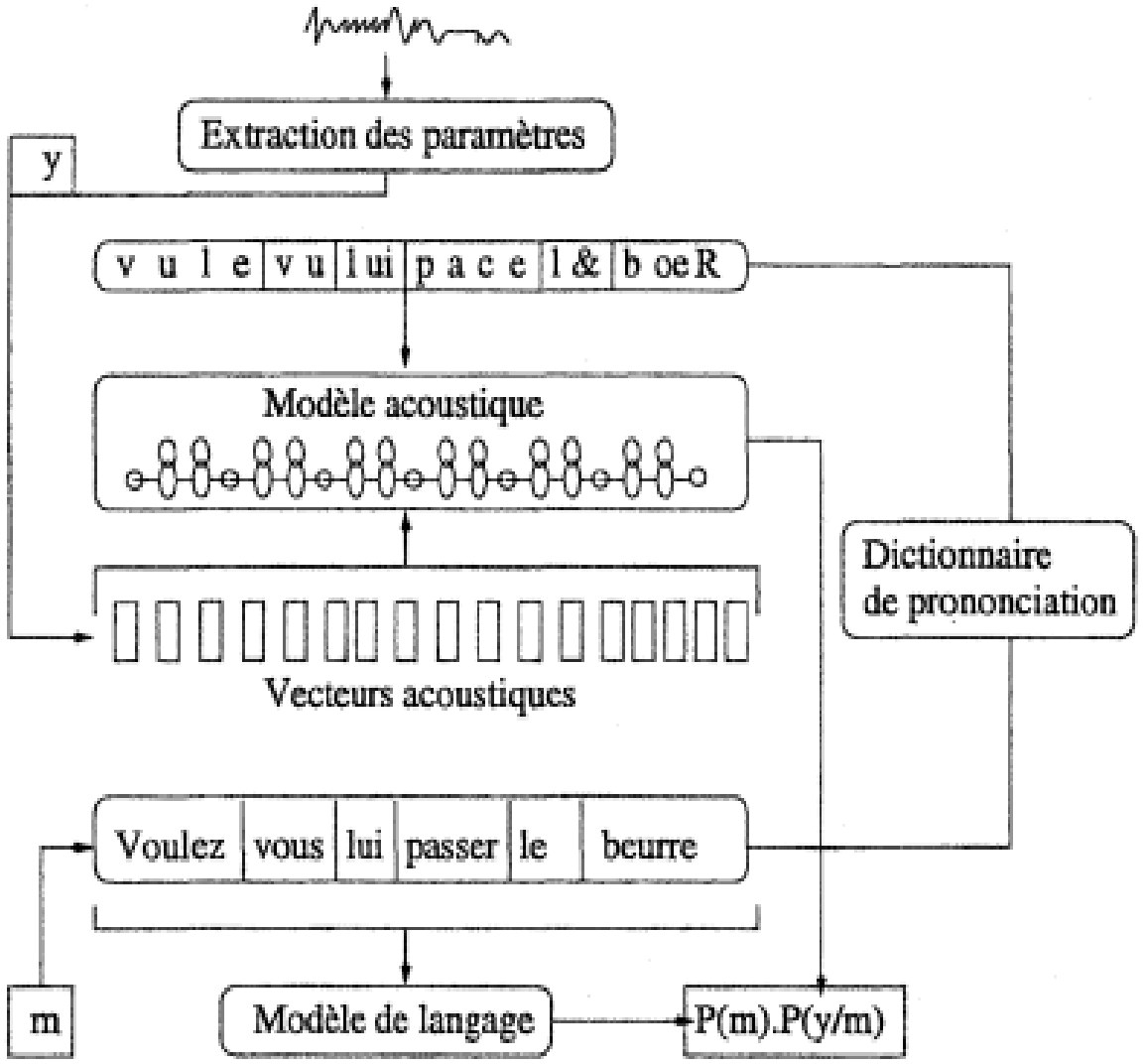
Etude du fonctionnement de la reconnaissance automatique de la parole

chaine de fonctionnement du système de reconnaissance vocale





La reconnaissance de la parole, vue comme un problème de la théorie de la communication, a pour but de reconstruire un message **m** à partir d'une séquence d'observations **y**



L'étape de la reconnaissance consiste donc à déterminer la suite de mots m qui maximise le produit des deux termes **P(m)** et **P(y/m)**

Les modèle de langue

L'objectif des modèles de langage probabilistes est d'estimer la probabilité $P(W_1^T)$ d'une séquence de mots

$$\boldsymbol{W_1^T = \omega_1, \omega_2, \ldots, \omega_T}$$

Les modèles n-grammes

Ils reposent sur l'application de la règle des probabilités conditionnelles, définissent la vraisemblance d'une suite de mots $W_1^T = \omega_1, \omega_2, \ldots, \omega_T$ comme suit

$$P(w_1, w_2, \ldots w_N) = \prod_{i=1}^{N} p(w_i / w_1, \ldots w_{i-1})$$

où $p(w_i/w_1, \ldots w_{i-1})$ est la probabilité du $i^{ème}$ mot de la suite W_1^N, sachant tous les mots précédemment émis.

Les modèles n-grammes

Dans la version la plus simple du modèle, la probabilité du mot Wi sachant son contexte $\omega_{i-n+1},\ldots \omega_{i-1}$ est estimée suivant le principe du maximum de vraisemblance sur un corpus d'apprentissage W représentant la langue

$$p(w_i/w_{i-n+1},\ldots w_{i-1}) = \frac{N(w_{i-n+1},\ldots w_{i-1},w_i)}{\sum_{j\in W} N(w_{i-n+1},\ldots w_{i-1},w_j)}$$

où $N(.)$ est le nombre d'occurrences de la suite de mots en argument dans le corpus W

Les modèles n-classes

Si n suppose qu'un mot ne peut appartenir qu'à une seule classe, la probabilité d'un mot sachant son contexte est définie comme suit

$$p(w_i/w_{i-n+1},\dots w_{i-1}) = p(w_i/C(w_i))p(C(w_i)/C(w_{i-n+1}),\dots C(w_{i-1}))$$

où $C(.)$ est la classe à laquelle appartient le mot en argument.

Les modèles n-classes

La probabilité d'appartenance d'un mot à une classe, p($\omega_i/C(\omega_i)$), est calculée comme suit

La probabilité d'appartenance d'un mot à une classe, $p(w_i/C(w_i))$, est calculée comme suit :

$$p(w_i/C(w_i)) = \frac{N(w_i)}{N(C(w_i))}$$

où $N(.)$ est le nombre d'occurrences de l'argument dans le corpus.

Les modèles POS

le modèle POS calcule la somme des probabilités n-classes sur toutes les classes associées au mot à prédire ω_t. Un modèle POS tri grammes (3-grammes), estimant la probabilité d'un mot ω_t sachant un historique de classes, est généralement défini comme suit :

$$p(w_t/g_{t-2},g_{t-1}) \approx \sum_{g_t \in G_{w_t}} p(w_t/g_t)p(g_t/g_{t-2},g_{t-1})$$

où $g(w_t) = g_t$ est la classe (POS) du mot w_t au temps t et G_{w_t} est l'ensemble de toutes les classes associées au mot w_t, $p(w_t/g_t)$ est la probabilité d'appartenance du mot w_t à la classe g_t (peut être estimée par la formule 3.5) et $p(g_t/g_{t-2},g_{t-1})$ est la probabilité de la suite de classes g_{t-2},g_{t-1},g_t qui peut être estimée par la formule

$$p(w_i/w_{i-n+1},\dots w_{i-1}) = \frac{N(w_{i-n+1},\dots w_{i-1},w_i)}{\sum_{j \in W} N(w_{i-n+1},\dots w_{i-1},w_j)}$$

Les modèles POS

En gardant les mêmes notations utilisées, un modèle POS bigrammes (2-grammes) estime donc la probabilité d'un mot ω_t sachant un historique ω_{t-1} comme suit :

$$p(w_t/w_{t-1}) = \sum_{g_t \in G_{w_t}} p(w_t/g_t) p(g_t/w_{t-1})$$

$$= \sum_{g_t \in G_{w_t}} p(w_t/g_t) \sum_{g_{t-1} \in G_{w_{t-1}}} p(g_t/g_{t-1}) p(g_{t-1}/w_{t-1})$$

avec

$$p(g_{t-1}/w_{t-1}) = \frac{N(w_{t-1}, g_{t-1})}{N(w_{t-1})},$$

où $N(.)$ est l'occurrence de la suite de mots en argument.

Afin de modéliser cette approche probabiliste, on utilise le langage de programmation *python* pour modéliser le fonctionnement, dans le domaine pratique, de la RAP.

on utilise **Google Web Speech API,** et le module *python* **speech_recognition**

```python
import speech_recognition as sr

# create a recognizer
r = sr.Recognizer()
mic = sr.Microphone()

L=''
while L=='':
    with mic as source:
        audio = r.listen(source)
    try:
        L=r.recognize_google(audio)
    except sr.UnknownValueError:
        print('Sorry, could you repeat your message?')
print(L)
```

On utilise par suite, la commande

r.recognize_google(audio, show_all=True)

qui renvoie les possibilité prise par le système afin d’aboutir au message finale

Expérience

On lance le programme python, puis on dit la phrase suivante :

Input >> ‘*one of the joys of being a geologist is fieldwork*’

Le résultat de cette expérience :

Output >> ‘*one of the joys of being a geologist is fieldwork*’

Résultat

>> **r.recognize_google(audio, show_all=True)**

{'alternative': [{'transcript': 'one of the joys of being a geologist is fieldwork', 'confidence': 0.87437057}, {'transcript': 'one of the joys of being a geologist his field work'}, {'transcript': 'one of the joys of being a geologist his fieldwork'}, {'transcript': 'one of the joys of being a geologist is field work'}, {'transcript': 'one of the joys of being a geologist use fieldwork'}], 'final': True}

Modélisation

RoboDoc

Description

Pour modéliser l’efficacité de la communication avec la machine à travers la reconnaissance vocale, j’ai programmé le système ROBODOC qui interagit avec l’utilisateur d’une manière délicate, afin de repérer sa maladie probable, en lui présentant une liste de symptômes. L’utilisateur choisit une possibilité à chaque essai, et avec chaque réponse le système fait ses calcules pour trouver la maladie la plus probable.

Le système ROBODOC fonctionne suivant le processus suivant :

Le système repère les données sur les maladies d’après une base de données SQL ‘ **Myhealth.db’**

```
import sqlite3
conn = sqlite3.connect('Myhealth.db')
c = conn.cursor()
```

TABLE DISEASE

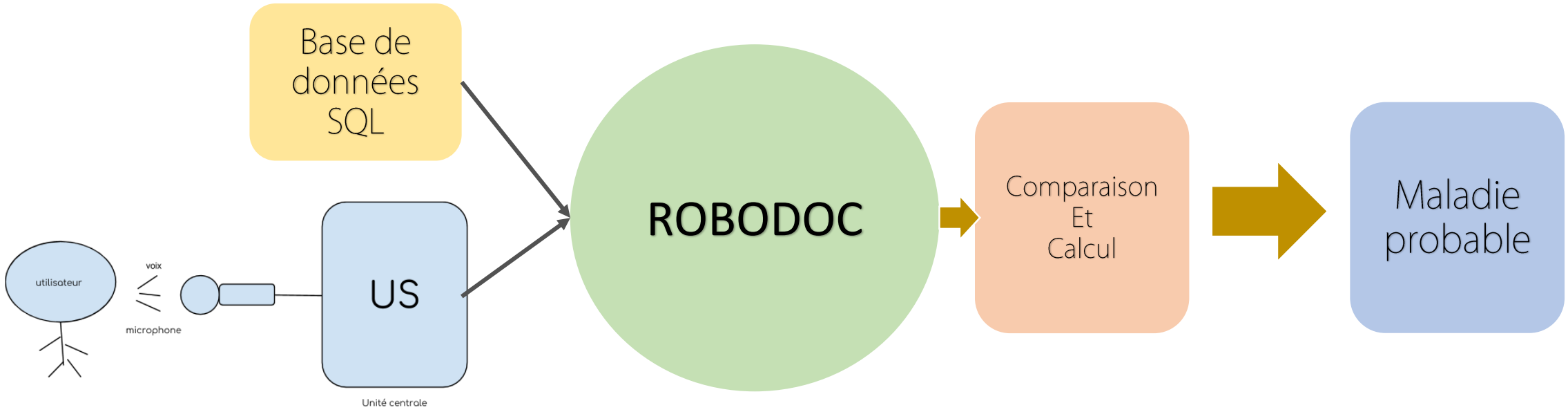
	cancer	migraine	depression	nauxestomac	allergie	colonirritable	maladiecardiovasculaire
	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	fatigue	fatigue	tristesse	vomissem...	oeil rouge	constipation	malaise dans la poitrine
2	vomissement	mal à la tête	perte d'intérêt	fatigue	vomissement	crampes au ventre	vertiges
3	difficulté à rés...	vomissement	baisse de libido	ballonnem...	larmoiement	diarrhée	vomissements
4	troubles de la...	sensibilté à la...	fatigue	crampes a...	difficultés à respirer	fatigue	difficulté à réspirer
5	douleur	sueurs froides	insomnie	crampes a...	douleurs abdominal...	besoin urgent d'aller...	manque d'énergie
6	perte d'appétit	troubles de vi...	troubles digestif	douleur au...	peau rouge	des flatulences	rythme cardiaque plus rapide
7	constipation	sensibilité au ...	perte d'appétit	brûlures d'...	peau séche	trouble digestif	palpitation cardiaque
8	trouble digestif	sensation d'él...	trouble de co...	troubles di...	constipation	douleur	fatigue

On utilise le module python ***speech_recognition*** pour recevoir les messages de l'utilisateur, ainsi que **Google Web Speech API**

Et pour envoyer des messages vocaux aux utilisateurs, on utilise les modules ***pygame*** et ***gtts***

L’algorithme

Résultats de la modélisation


Base de données SQL
ROBODOC
Comparaison Et Calcul
Maladie probable
utilisateur
voix
microphone
US
Unité centrale

Conclusions

SOURCES

[1] https://fr.wikipedia.org/wiki/Interactions_homme-machine
[2] https://fr.wikipedia.org/wiki/Reconnaissance_automatique_de_la_parole
[3] Fang Chen and Kristiina Jokinen, Speech Technology, Theory and Applications. New York, USA: Springer, 2010.
[4] BCC Research. Global Voice Recognition Market To Reach $113 Billion In 2017. [Online]. http://www.bccresearch.com/pressroom/ift/global-voice-recognition-market-reach-$113- billion-2017
[5] Jardino (M.) et Adda (G.). - Language modeling for csr of large corpus using automatic classification of words. In: Proceeding of the European Conference On Speech Communication and Technologie, pp. 1191-1194. - September 1993.
[6] Witschel (P.). - Constructing lingcistic oriented language models for large vocabulary speech recognition. In: Proceeding of the European Conference On Speech Communication and Technologie, pp. 1199-1202. - September 1993.
[7] Cerf-Danon (H.) et El-Bèze (M.). - Three different probabilistic language models: Comparison and combination. In: Proceeding of the International Conference on Acoustics, Speech and Signal Processing, pp. 297-300. - Toronto, Canada, 1991.