# Wikipedia Recommender System with serendipity

Barszezak Yoann,Bricout Raphael, David Nicolas, Dupre Remi,
Fouche Aziz, Gourdel Garance, Kaddar Younesse, Mallem Maher,

Department of Computer science, ENS Paris-Saclay

November 2017

**TO DO : Abstract**

## 1 Introduction

## 2 Problem Overview

## 3 Retrivial of Candidate articles

### 3.1 Neighbourhood

Let's try to understand the idea of proximity between two articles. One way to do it is to introduce a potential function telling us the proximity between two articles.

#### 3.1.1 Naive Similarity

Our attempt to define this potential function uses the set of categories linked to an article, it is called the similarity.

**Definition 3.1.** Let $A_1$ and $A_2$ be two articles. We call $C_1$ (resp. $C_2$) the set of categories linked to $A_1$ (resp. $A_2$). We define the similarity of $A_1$ and $A_2$ the following quantity:

$$S_W(A_1, A_2) = \frac{Card(C_1 \cap C_2)}{min(Card(C_1), Card(C_2))}$$

*Remark.* $\forall A_1\ A_2,\ S_W(A_1, A_2) \in [0, 1]$

With this definition in mind, let's seek for an output of the "Retrivial of Candidate Articles".

Let's call $A_c$ the current article. Given an subinterval $I$ of $[0, 1]$ (define in the serendipity subsection), one way to find candidate articles will be to pick randomly $N$ articles $(A_i)_{1 \leq i \leq N}$ such that:

$$\forall i\ S_W(A_c, A_i) \in I$$

This approach should work as long as the similarity is precise (in the case of wikipedia, it means as long as an article have a significant number of categories). Unfortunately, some articles are poorly cateorised. Thus, it is not always accurate to use this object.
Therefore, an other approach is needed.

#### 3.1.2 Ontology basis

-Modele general -Utilisation -YAGO

### 3.2 serendipity

#### 3.2.1 Similarity

-choix de I

#### 3.2.2 Ontology

-parcours de l'Ontology

## 4 Candidate Ranking

### 4.1 possible input in neural network

#### 4.1.1 Category vector

#### 4.1.2 Word2Vect and Wikidata

### 4.2 Wide and deep Neural Network

#### 4.2.1 Wide : memorization/focus

#### 4.2.2 Deep : generalization/serendipity

## 5 System architecture

## 6 Exprimentation

## 7 State of the Art