



Recherches sur les algorithmes de matching

Description du document :

Titre	Recherches sur les algorithmes de matching
Objet	Recherches et documentation sur les algorithmes de matching
Auteurs	Hégron Stanislas
E-mail	psynderFR@gmail.com
Mots-clés	Recherches, documentation, algorithmes, matching
Promotion	2022
Date de mise à jour	03 mars 2021
Version du modèle	1.3

Tableau des révision :

Date	Version	Auteur(s)	Sections(s)	Commentaire
05/02/2021	1.0	Hégon Stanislas	I - Introduction et II - Premières idées	Création du fichier, renseignements sur le but du documents et mes premières idées sur le fonctionnement des algorithmes de matching ainsi que l'application pour mon projet
20/03/2021	1.2	Hégon Stanislas	III - Recherches et IV - Annexes	Recherches sur les algorithmes de matching et rédaction du fruit de mes recherches
23/03/2021	1.3	Hégon Stanislas	Description du document et tableau des révisions	Mise à jour du document avec une cartouche et un tableau de révisions au début de celui-ci

Table of content

I - Introduction	5
II - Premières idées	6
III - Recherches	7
Qu'est-ce qu'un algorithme de matching ?	7
Un algorithme de matching est-il 100% fiable ?	7
Comment ça marche ?	7
Comment appliquer cela à notre projet ?	9
IV - Annexes	11

I - Introduction

Dans le but de déceler trouver avec quels thérapeutes seraient aptes à soigner les potentiels troubles décelés chez un utilisateur en fonction de leurs spécialités, il nous faut réfléchir au fonctionnement de l'algorithme afin d'être le plus efficient possible.

Ce document a pour but de garder un trace écrite des recherches sur les algorithmes de matching dans le but de pouvoir construire un algorithme efficace étant capable de dire (avec un pourcentage) quel thérapeute serait potentiellement le plus apte à soigner un utilisateur, atteint de troubles dépressifs, qui souhaite être aidé.

II - Premières idées

Si on décèle plusieurs troubles chez un utilisateur, on regarde quelle(s) spécialité(s) traite chaque trouble. De là, on récupère tous les thérapeutes qui ont au moins une de ces spécialités et on définit le matching à 50% (Afin d'avoir une base).

Puis pour chaque trouble en plus que le thérapeute est capable de traiter on ajoute 5% de matching.

Si le thérapeute gère aussi un trouble dont l'utilisateur a entre 30 et 49% du score d'un questionnaire lié à un trouble, on ajoute 3% de matching. On peut considérer que nous nous sommes trompés et qu'il y a une faible chance mais non négligeable que la personne soit potentiellement atteinte du trouble.

De la même manière, si le thérapeute a déjà **soigné** (dans le sens guérir) un ou des patients atteints d'un même trouble, on ajoute 1-2%. Mais s'il a soigné (sans guérir) un ou des patients, on diminue de 1-2%.

Cependant, il faudra, sûrement dans un premier temps, prendre en compte la localisation de l'utilisateur souhaitant se faire aider et les thérapeutes susceptibles de l'aider.

C'est-à-dire, que pour un rayon donné, tous les thérapeutes en faisant partie (avec une limite supérieure de 5%?) devront être considérés comme potentiels aidants et formeront la base des données à traiter. Ceux (bien) au-delà de la zone, n'auront pas à être pris en compte puisque trop éloignés de la personne souhaitant se faire aider.

III - Recherches

- Qu'est-ce qu'un algorithme de matching ?

“Les algorithmes de matching sont des algorithmes utilisés pour résoudre des problèmes de matching de graphiques, en théorie des graphes” (traduction).

Source: <https://brilliant.org/wiki/matching/>

“[...] En fonction de la manière dont l'algorithme est configuré, vous pouvez aussi bien récupérer une shortlist de 15 candidats qu'une shortlist de 3 candidats. Cela dépend de vos besoins et de votre sélectivité.

[...]

*Ensuite, l'algorithme de matching va déduire un “**score de correspondance**” entre le candidat et l'offre d'emploi. Si ce score est élevé, cela signifie que le candidat correspond très bien aux principaux critères de l'offre d'emploi. À l'inverse, un score très faible signifie que le candidat est bien trop éloigné de vos critères de choix.*

*Attention, il faut bien noter que **ce score de correspondance peut varier en fonction de l'importance donnée à chaque critère**. Si une compétence particulière est absolument obligatoire, il faudrait **pondérer de manière plus forte le critère “compétences”**”.*

Source: <https://www.kicklox.com/blog-client/matching-recrutement-a-savoir>

- Un algorithme de matching est-il 100% fiable ?

*“La réponse est plus compliquée qu'il n'y paraît. En réalité, **tout dépend de la manière dont il a été mis en place** [...]”*

On ne peut malheureusement pas encore se reposer à 100% sur un algorithme de matching pour trouver le meilleur candidat [...]

*La meilleure approche reste de combiner l'algorithme de matching avec un double filtre “humain”. Vérifiez toujours par vous-même, **n'accordez pas une confiance aveugle à un algorithme**”.*

Source: <https://www.kicklox.com/blog-client/matching-recrutement-a-savoir>

- Comment ça marche ?

Au plus simple, on regarde est-ce qu'une entrée A a une ou des caractéristiques similaires à une entrée B. Si oui, il y a match.

Cependant, on peut avoir une dimension, une précision supérieure si nous cherchons un critère en particulier, si nous avons un ordre de préférence ou que l'on souhaite privilégier certains points avant d'autres.

Dans ce cas, l'entrée A et l'entrée B doivent classer par ordre d'importance les caractéristiques qu'ils veulent mettre en avant.

Par exemple, un élève qui fait ses vœux pour des universités classe par ordre de préférence les universités qu'il choisit. Les universités font de même avec les élèves qui souhaitent s'inscrire dans leur établissement mais les classent en fonction de leur moyenne générale (par exemple). De ce fait, les deux ont des caractéristiques similaires triées selon un critère permettant alors de lancer l'algorithme de matching.

En gardant l'exemple précédent et en admettant que chaque université ne peut accueillir qu'un seul élève, nous avons:

	Université A	Université B	Université C
Veut les élèves	Elève A	Elève B	Elève A
	Elève B	Elève A	Elève C
	Elève C	Elève C	Elève B

	Elève A	Elève B	Elève C
Veut les universités	Université C	Université C	Université B
	Université A	Université B	Université C
	Université B	Université A	Université A

L'élève A veut l'université C en priorité et comme celle-ci désire aussi cet élève fortement, le match est direct.

L'élève B souhaitait lui aussi l'université C mais comme celle-ci a déjà été prise par l'élève A, qui est mieux classé que lui, il doit se rabattre sur un autre choix. Donc son deuxième choix, qui est l'université B.

Enfin, l'élève C souhaitait l'université B mais comme celle-ci a déjà été prise par l'élève B qui est mieux classé que lui, il doit se rabattre sur un autre choix (donc deuxième vœu). Mais comme celui-ci a aussi été pris par un élève mieux classé, il doit de nouveau se rabattre sur le vœu suivant.

Ce qui nous donne

	Université A	Université B	Université C
Veut les élèves	Elève A	Elève B	Elève A
	Elève B	Elève A	Elève C
	Elève C	Elève C	Elève B

	Elève A	Elève B	Elève C
Veut les universités	Université C	Université A	Université B
	Université A	Université B	Université C
	Université B	Université C	Université A

Ici, tous nos élèves se retrouvent dans une université. Mais il aurait pu arriver qu'un des élève ne se retrouve dans aucune pour diverses raisons:

- Il n'est pas dans le classement des élèves souhaités par l'université;
- Il n'a pas choisi d'université qui aurait pu vouloir de lui;
- Il n'y avait plus de place dans toutes les autres universités qu'il aurait choisi;

Alors, dans le cas de l'élève C, il était par chance premier du classement de cette université et est donc prioritaire sur cette sélection. Cependant, si un autre étudiant avait "déjà été choisi", mais qu'il était plus bas dans le classement, l'élève C aurait pris sa place puisque mieux classé. On aurait alors parlé de **choix provisoire** pour parler de l'autre élève.

Sources: <https://www.carms.ca/the-match/how-it-works/> (https://youtu.be/28CvXJ_DcGo)
<https://www.nrmp.org/matching-algorithm/> (<https://youtu.be/kvgfgGmemdA>)

- Comment appliquer cela à notre projet ?

Premièrement, à la différence de l'exemple ci-dessus, notre entrée B (qui serait les thérapeutes) ne ferait pas de classement des personnes qu'il souhaite traiter. Ce serait plutôt un classement de la ou des spécialités qu'il/elle maîtrise et trouble qui ont déjà été soignés par le passé (ou non).

On pourrait alors imaginer quelque chose comme:

Thérapeute Th:

Spécialisé dans les troubles	À déjà traité	À déjà guéri
T1	T2	Oui
T3	T4	Non
	T5	Non

Patient *Pa*:

Trouble traités	A le trouble	Score au questionnaire
T1	Oui	73%
T2	Non	48%
T3	Oui	51%
T4	Non	5%
T5	Oui	65%

En appliquant le raisonnement (et les valeurs) exprimés dans la partie [II - Premières idées](#), on pourrait alors obtenir les résultats suivants:

T1: patient a le trouble + est spécialité du thérapeute => +50%

T2: patient n'a pas le trouble mais a un score entre 30 et 49% à ce trouble + thérapeute a déjà traité **avec** guérison => +3% + (1-2%)

T3: patient a le trouble + est spécialité du thérapeute **mais** déjà une spécialité du thérapeute est comptée => +5%

T5: patient a le trouble + thérapeute a déjà traité **sans** guérison => -(1-2%)

Total :

min: 58%

max: 58%

/!\ Les +1-2% s'annulent puisqu'en même quantité (+1-1 = +2-2 = 0).

Donc pour *Pa* et *Th* nous aurions **théoriquement** un matching de 58%.

IV - Annexes

Différentes sources utiles et/ou utilisées dans ce document:

<https://www.kicklox.com/blog-client/matching-recrutement-a-savoir>

<https://brilliant.org/wiki/matching-algorithms/>

<https://brilliant.org/wiki/matching/>

<https://www.sciencedirect.com/topics/computer-science/matching-algorithm>

<https://www.carms.ca/the-match/how-it-works/> (https://youtu.be/28CvXJ_DcGo)

<https://www.nrmp.org/matching-algorithm/> (<https://youtu.be/kvgfgGmemdA>)