Une synthèse du rendu

1.Explication de la méthode :

Notre problème est un problème d'extraction d'information qui se fait en 3 étapes :

a. **NER (Named entity recognition) :** ma démarche consistait dans un premier temps à charger un modèle pré entrainé de NER (Named entity recognition) entrainé par CNN (réseau de neurones convolutif) formé sur le corpus en français (Sequoia et WikiNER). Affecte un context à des vecteurs-tokens, fait du POS-tagging, et prend en charge l'identification des entités PER, LOC, ORG et MISC.

b. **Word-sense disambiguation :** ceci n'a pas été fait pour mon rendu car c'est une étape qui prend trop de temps

c. **Relation extraction :** grâce à notre modèle pré entraîné, nous avons fait une recherche de nos deux mots clés (salaire, ancienneté) et ensuite chercher dans l'arborescence des phrases où ces mots ont été évoqués pour chercher le fils qui vient après (resp. avant) l'article de (resp. le mot euros) pour retrouver l'ancienneté (resp. le salaire)

2.Problèmes rencontrés :

Ma méthode a marché pour les 3 arrets (1, 2 et 4) cependant elle comporte plusieurs failles :

a. **Une connaissance préalable du contexte où chaque mots sera évoqué :** si l'utilisateur (le développeur) n'est pas au courant des mots précédents ou suivants nos mots clés, la tâche de retrouver le salaire et l'ancienneté sera impossible (par exemple : je savais que le chiffre arrivant après ancienneté et de est le bon chiffre)

b. **Impossible de faire autrement que par une recherche par mots clés :** puisque mon modèle de REN est un modèle pré-entrainé sur un corpus différent de celui des arrêts, les labels et les noms des entités spécifique au jargon judiciaire ne pourront pas y faire partie, donc une reconnaissance personnalisée des entités nommées à l'aide de spaCy est impossible.

c. **Désambiguïsation du sens du mot:** dans l'arrêt 3 qui pose problème, les mots recherchés sont évoqués de manière indirectes, donc il a fallu avoir un modèle

d’embedding capable de mesurer le contexte et détecter les connexité entre les mots (par exemple : prendre la date d’entrée et de sortie puis calculer l’ancienneté)

3.Axes d’amélioration :

Pour les 3 étapes :

a.NER (Named entity recognition) : réaliser son propre NER, par exemple travailler sur une architecture LSTM-CRF standard, où les étiquettes des entités en sont modélisées comme des étiquettes multiples correspondant au produit cartésien des étiquettes imbriquées dans notre notre architecture de LSTM-CRF.

b.Word-sense disambiguation : en exploitant les relations sémantiques entre les sens des mots tels que la synonymie et l’hyponymie

c.Relation extraction : Mieux représenter les spans d’un texte, masquer les spans aléatoires contiguës, plutôt que les tokens aléatoires. Construire des représentations de relations uniquement à partir de texte lié à une entité.