






## Activités du Lundi 1 avril au Vendredi 5 avril :

- **Amélioration des regex de nettoyage du corpus**
- **Choix de passer les fichiers annotés avec les « # » en entrée de Stanford Parser**
  - J'avais testé Stanford Parser sur les fichiers non annotés avec les « # » mais il en était ressorti la même annotation qu'avec les « # ». Donc pour la tâche de récupération des connecteurs, ce n'est pas gênant d'utiliser le corpus annoté avec « # ». Par contre pour l'analyse en dépendance et récupération du chemin syntaxique il faudra utiliser Stanford Parser sur le corpus non-annoté sinon il y aura des erreurs. (j'ai déjà appliqué Stanford Parser et addDiscourse sur tous les fichiers : annotés / non-annotés / AQUAINT / TimeBank, comme ça nous avons tout → fichiers sur github dans ressources)
- **addDiscourse plante sur les fichiers wsj\_0610 et wsj\_0736**
  - Annotation manuelle des connecteurs de ces deux fichiers
    - J'ai copié les fichiers wsj\_0610 et wsj\_0736 du dossier TimeBank\_StanfordParser pour le coller dans le dossier TimeBank\_AddDiscourse pour ensuite effectuer l'annotation manuelle.

 TimeBank	04/04/2019 09:21	Dossier de fichiers
 TimeBank_AddDiscourse	05/04/2019 10:20	Dossier de fichiers
 TimeBank_Connecteurs	05/04/2019 10:02	Dossier de fichiers
 TimeBank_NewInput	05/04/2019 10:24	Dossier de fichiers
 TimeBank_StanfordParser	22/03/2019 22:14	Dossier de fichiers

Le processus se fait dans cet ordre :

- 1- Passage des fichiers TimeBank dans Stanford Parser → TimeBank\_StanfordParser (fichiers annotés par Stanford Parser)
  - 2- Passage des fichiers annotés par Stanford Parser dans addDiscourse (plantage sur deux fichiers annotation manuelle) → TimeBank\_AddDiscourse (fichiers dont le texte a le même format que les fichiers Stanford Parser, c'est-à-dire qu'ils ont les parenthèses et les catégories grammaticales dans le corps du texte)
  - 3- Passage des fichiers annotés par addDiscourse dans la fonction de nettoyage → TimeBank\_Connecteurs (textes reconstruits, sans parenthèses, sans catégories grammaticales)
  - 4- Passage des fichiers de TimeBank\_Connecteurs dans la fonction d'assignation des nouveaux identifiants type #s1, #s2 → Timebank\_NewInput (qui servira de nouvel input pour traiter les fichiers .txt)
- **Amélioration des règles pour annoter les nouveaux connecteurs de TB**
  - **Mise-à-jour des CSV avec les nouveaux connecteurs**

- features\_signaux.csv,
  - dataframe\_contexts.csv,
  - dataframe\_id.csv,
  - + fichier fusionné signaux\_contexts\_id.csv).
- **Remarque après observation du corpus : addDiscourse n'annote pas les « as early as », « as early », « so far », etc.**
  - Rajout de ces expressions dans les listes d'addDiscourse pour qu'elles soient annotés avant un timex dans le corpus
    - Nous obtenons : as#0#0 early#0#0 as#0#0 / so#1#0 far#1#0... + timex
- **Diagrammes d'architecture de mes programmes**
  - Avec LucidChart
  - Améliorations à effectuer avant de push
- **En cours** : faire en sorte que les as#0#0 early#0#0 as#0#0 deviennent as#s1 early#s1 as#s1 puis as#s1>early#s1>as#s1 (pour respecter le même format que les timex type last#t1>week#t1)
- A la réunion du mardi 2 avril, Gaëtan m'a demandé d'ajouter dans les contextes des events/timex/signaux le contexte en POS (**fait**) et en lemmes (**à faire**)