Atelier « Outils pour l'exploration lexicale, morphosyntaxique et syntaxique de la Base de français médiéval »

Zeina Tmart, Alexei Lavrentiev,
Céline Guillot-Barbance et Sophie Prévost

IHRIM ENS de Lyon & CNRS

Colloque « Diachro X », Paris, 2 juin 2021

Plan

- 1. Lemmes et étiquettes morphosyntaxiques (requêtes CQL)
  - Base de français médiéval : **http://txm.bfm-corpus.org**
- 2. Requêtes syntaxiques (CQP)
  - Corpus SRCMF2022 en ligne : **https://txm-bfm.huma-num.fr**
- Pause
- 3. Requêtes syntaxiques (TIGERsearch)
  - Corpus SRCMF2022 avec le logiciel **TXM pour poste**
- 4. Annotation de corpus
  - Corpus ATELIER-DIACHRO avec le logiciel **TXM pour poste**

1. Lemmes et étiquettes morphosyntaxiques (CQL)

- Portail de la BFM : txm.bfm-corpus.org
- Corpus :
  - BFM2019
  - BFM2019 lemmatisé
- Jeu d'étiquettes Cattex2009
  - https://bfm.ens-lyon.fr/IMG/pdf/Cattex2009_2.0.pdf
  - VERinf
  - DETdem
  - PROdem

1. Lemmes et étiquettes morphosyntaxiques (CQL)

Corpus BFM2019 : formuler une requête simple à partir des propriétés de mots

- Commande **Index** :
  - Afficher tous les verbes à l'infinitif
    - Requête : [cattex-pos="VERinf"]
  - Cibler la requête sur le verbe "faire"
    - Requête : [cattex-pos="VERinf" & word="f..re"]
    - Afficher les propriétés lemma et lemma_src
  - Éliminer le bruit
    - Requête : [cattex-pos="VERinf" & word="ff?[ae]i?re"]

1. Lemmes et étiquettes morphosyntaxiques (CQL)

Corpus BFM2019 : vérifier un lemme

- Commande **Concordance** :
  - Double-clic
    - Masquer lemma-src
  - Requête
    - [lemma="\|faire\|" & word="ffaire"]
    - [lemma contains "faire" & word= "ffaire"]

1. Lemmes et étiquettes morphosyntaxiques (CQL)

Corpus BFM2019 lemmatisé : des résultats plus homogènes

- Commande **Index** :
  - Requête : [cattex-pos="VERinf" & word="ff?[ae]i?re"]

1. Lemmes et étiquettes morphosyntaxiques (CQL)

Corpus BFM2019 : Recherche sur une succession de deux propriétés

- Commande **Index** :
  - Chercher les déterminants et pronoms démonstratifs
    - Requête : [cattex-pos= ".*dem"]
  - Chercher quels mots suivent les déterminants démonstratifs
    - Requête : [cattex-pos="DETdem"][]
  - Préciser la requête pour viser les pronoms relatifs subséquents
    - Requête : [cattex-pos="PROdem"] [cattex-pos="PROrel"]

1. Lemmes et étiquettes morphosyntaxiques (CQL)

Corpus BFM2019 : rechercher les démonstratifs en discours direct

- Commande **Lexique :**
  - Afficher la propriété “q”
    - 0 : hors DD
    - 1 : DD de niveau 1
    - 2 : DD de niveau 2
    - 3 : DD de niveau 3
- Commande **Index :**
  - Répartition des démonstratifs en DD
    - Affichage de la propriété “q”
    - Requête : [lemma contains "(cil|cist)"]
- Commande **Concordance** :
  - Démonstratifs en DD de niveau 3
    - Requête : [q="3" & lemma contains "(cist|cil)"]

1. Lemmes et étiquettes morphosyntaxiques (CQL)

Corpus BFM2019 : chercher les démonstratifs en discours direct

- Commande **Concordance** :
  - Chercher un lemme “cil” , “cist” ou “ce” dans un DD de niveau 1
    - Trier les résultats sur le pivot
    - Trier les résultats sur le contexte gauche
  - Chercher les incises de type “ce dist X”
  - Chercher les incises de type “ce fait X”

2. Requêtes syntaxiques (CQL)

- Portail expérimental https://txm-bfm.huma-num.fr
  - nom d'utilisateur : diachro10
  - mot de passe : diachro10
- Corpus SRCMF2022
  - 11 textes, 200 519 mots et ponctuations, annotation syntaxique manuelle
- Modèle d'annotation Universal Dependencies https://universaldependencies.org
  - ud-deprel : étiquette syntaxique (relation de dépendance : (root), nsubj, obj, iobj, obl, nmod, det, conj, cc…)
  - ud-id : identifiant du mot (position dans la phrase)
  - ud-head : identifiant du gouverneur
  - ud-head-deprel : étiquette syntaxique du gouverneur
  - ud-dep-deprel : étiquettes syntaxiques du ou des dépendants directs
- Requêtes à copier-coller : https://bit.ly/3N1tFoX

2. Requêtes syntaxiques (CQL)

- Afficher l'étiquette syntaxique d'un pronom démonstratif
  - Commande Index
    - afficher les propriétés lemma et ud-deprel
    - requête [cattex-pos="PROdem"]
- Pour le déterminant démonstratif, on s'intéresse à sa « tête »
  - Commande Index
    - afficher les propriétés lemma et ud-head-deprel
    - requête [cattex-pos="DETdem"]

2. Requêtes syntaxiques (CQL)

- Recherche de phrases OSV
  - Construction de la requête
    - Objet **[ud-deprel="obj"]** ➔
      - \+ dépend du verbe principal **[ud-deprel="obj" & ud-head-deprel="root"]**
    - 0 ou plusieurs mots **[]*** ➔
      - \+ sans traverser la frontière de la phrase **[ud-id!="1"]***
    - Sujet **[ud-deprel="nsubj" & ud-head="root"]** ➔
      - \+ dans la même phrase ! **[ud-id!="1" & ud-deprel="nsubj" & ud-head="root"]**
    - 0 ou plusieurs mots dans la même phrase **[ud-id!="1"]***
    - Verbe de la même phrase **[ud-id!="1" & ud-deprel="root" & cattex-pos="VERcjg"]**
      - en UD le verbe auxiliaire dépend de la racine nominale ou participiale

[ud-deprel="obj" & ud-head-deprel="root"][ud-id!="1"]* [ud-id!="1" & ud-deprel="nsubj" & ud-head-deprel="root"] [ud-id!="1"]* [ud-id!="1" & ud-deprel="root" & cattex-pos="VERcjg"]

2. Requêtes syntaxiques (CQL)

- Recherche de phrases OSV
  - Pour retrouver les phrases avec un verbe auxiliaire, on modifie le dernier élément :
    - Auxiliare qui dépend de la racine de la phrase :
      - **[ud-id!="1" & ud-deprel="aux" & ud-head-deprel="root"]**

[ud-deprel="obj" & ud-head-deprel="root"][ud-id!="1"]*[ud-deprel="nsubj" & ud-head-deprel="root" & ud-id!="1"] [ud-id!="1"]* [ud-deprel="aux" & ud-id!="1" & ud-head-deprel="root"]

  - On peut combiner les deux conditions !
    - **[ud-id!="1" & ((ud-deprel="root" & cattex-pos="VERcjg") | (ud-deprel="aux" & ud-head-deprel="root"))]**

[ud-deprel="obj" & ud-head-deprel="root"] [ud-id!="1"]*
[ud-deprel="nsubj" & ud-head-deprel="root" & ud-id!="1"] [ud-id!="1"]*
[ud-id!="1" & ((ud-deprel="root" & cattex-pos="VERcjg") | (ud-deprel="aux" & ud-head-deprel="root"))]

3. Requêtes syntaxiques TIGERSearch

- Préparer TXM
  - passer au niveau de mise à jour ALPHA
    - Editer > Préférences > TXM > Avancé > Niveau de mise à jour > sélectionner ALPHA
      - Cliquer sur « Apply and close »
  - installer l'extension "TIGERSearch"
    - Fichier > Ajouter une extension > TIGERSearch
      - Accepter les choix proposés
  - mettre à jour TXM et l'extension
    - Fichier > Vérifier les mises à jour
    - Accepter les options proposées pour TXM et TIGERSearch
  - Ajouter le moteur TIGER aux concordances
    - Editer > Préférences > TXM > Avancé > Search engines
      - Cocher « Show available search engines »

3. Requêtes syntaxiques TIGERSearch

- (R)ouvrir le fichier requetes-syntaxiques.txt
  - https://bit.ly/3N1tFoX
- Télécharger le corpus SRCMF2022 : https://bit.ly/3MBJCIt
- Charger le corpus dans TXM
  - Fichier > Charger > SRCMF2022-2022-05-20.txm

3. Requêtes syntaxiques TIGERSearch

- Langage similaire à CQL
- Requêtes sur plusieurs lignes
  - définition de variables : #obj:[cat="obj"]
  - opérateur & : combiner des conditions
  - // : pour ajouter un commentaire
  - >D : relation de dépendance syntaxique
  - >L : expression lexicale
- Nœuds terminaux et non terminaux
- Le moteur retourne un ensemble de phrases qui correspondent à la requête
  - plusieurs “matchs” possibles dans une phrase

3. Requêtes syntaxiques TIGERSearch

- Exemple de requête : phrases OSV

```
#pivot:[pos="VERB"]
& #clause:[cat="root" & type="VFin"]
& #clause >L #pivot
& #clause >D #obj:[cat=("obj"|"ccomp"|"obj\:advneg"|"obj\:advmod")]
& #clause >D #suj:[cat=("nsubj"|"csubj")]
& #obj    >L #objhead:[]
& #suj    >L #sujhead:[]
& #objhead .* #sujhead & #sujhead .* #pivot //OSV//
```

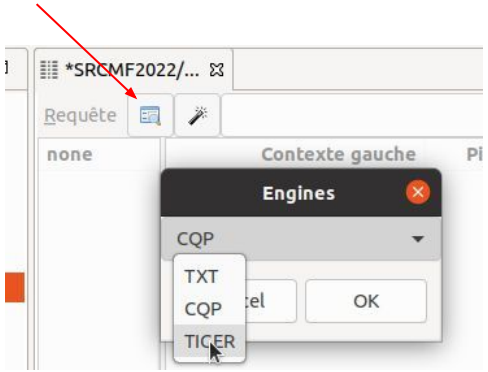
- Retrouvez cette requête dans le fichier requetes-syntaxiques.txt et copiez-la

3. Requêtes syntaxiques TIGERSearch

- Portail https://txm-bfm.huma-num.fr/txm/
  - cliquer sur le corpus SRCMF2022, puis sur l'icone « tête de tigre »
  - coller la requête, puis cliquer sur « Chercher »
- TXM 0.8.1
  - Commande « Arbres syntaxiques »
    - même manipulation que sur le portail

3. Requêtes syntaxiques TIGERSearch

- TXM 0.8.1
  - Concordance TIGER
    - sélectionner le corpus SRCMF2022
    - utiliser la commande « Concordances »
    - sélectionner le moteur TIGER
    - Copier-coller la Requête 5 depuis le fichier «requetes-syntaxiques.txt »



#pivot:[pos="VERB"] & #clause:[cat="root" & type="VFin"] & #clause >L #pivot & #clause >D #obj:[cat=("obj"|"ccomp"|"obj\:advneg"|"obj\:advmod")] & #clause >D #suj:[cat=("nsubj"|"csubj")] & #obj >L #objhead:[] & #suj >L #sujhead:[] & #objhead .* #sujhead & #sujhead .* #pivot //OSV//

- C’est la même requête sans les sauts de ligne !

3. Requêtes syntaxiques TIGERSearch

- Modifier la requête pour retrouver :
    - les phrases SOV
    - les phrases attributives, avec un sujet postposé au verbe auxiliaire

3. Requêtes syntaxiques TIGERSearch

- Conclusion
  - Les étiquettes syntaxiques ud-deprel, ud-head, etc. seront disponibles dans le corpus BFM2022
    - la qualité de l'annotation automatique n'est pas garantie
  - L'extension « Arbres syntaxiques » sera disponible en version « stable » avec TXM 0.8.2
  - Elle permettra
    - d'importer et d'exporter des annotations syntaxiques aux formats UD et TIGER XML
    - d'utiliser les moteurs de recherche TIGER et Universal Dependencies

4. Annoter un corpus sur TXM (pour poste)

- Télécharger le corpus de travail : https://bit.ly/3NxjA2C
  - Ce corpus contient le *Journal* de Nicolas de Baye et les *Mémoires* de Philippe de Commynes
  - ! La sauvegarde des annotations sur un grand corpus peut être longue (plusieurs minutes ou même plusieurs heures)
- Charger le corpus dans TXM
  - Fichier > Charger > ATELIER-DIACHRO-2022-05-23.txm

4. Annoter un corpus sur TXM (pour poste)

- Corriger des lemmes
  - Commande **Index** :
    - Requête : [lemma contains "f[ae]i?re"%cd]
      - Fère_|fère|_non vérifiée 1
      - ferè_|ferè|_non vérifiée 1
  - Commande **Concordance** :
    - Double-clic
    - Requête : [word="ferè" & lemma="\|ferè\|"]
  - Procédure d'annotation :
    - Insérer un "@" pour marquer le pivot de la requête
    - Sélectionner le crayon d'annotation
    - Choisir la propriété lemma
    - Corriger le lemme
    - Sauvegarder l'annotation

4. Annoter un corpus sur TXM (pour poste)

- Créer une nouvelle propriété
  - Temps => tpsV
    - “fist”
    - “face”
  - Personne => persV
    - “fait”