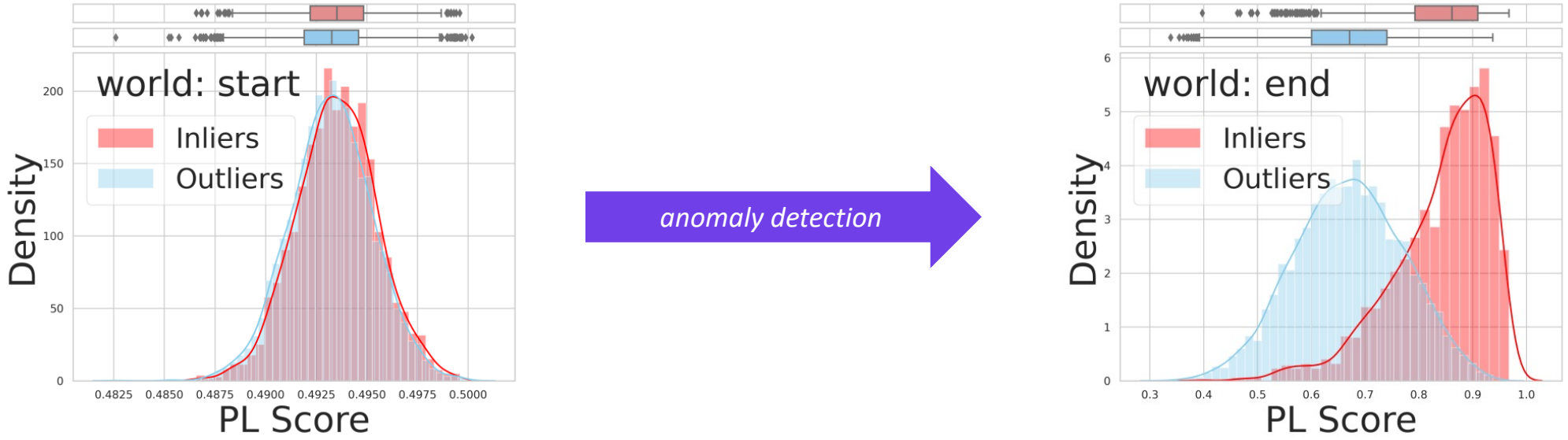
Quelques pistes pour la détection d’anomalies factuelles par intégration de connaissance externe

Séminaire TALia du 10/03/2023

Aurélien Renault



Plan

1. Quelques types d’anomalies textuelles
2. Zoom sur la détection d’anomalies par substitution (DATE)
3. Enrichissement d’un LM avec pré-entraînement faiblement supervisé
4. Enrichissement d’un LM avec base de connaissance externe
5. Détection de substitution d’entités avec LM enrichi
6. Expériences
7. Difficultés restantes à surmonter

Quelques types d'anomalies textuelles

Anomalies textuelles :

- Texte qui ne se décompose pas bien via un sous-ensemble de thèmes identifié dans le training set (NMF, voir séminaire n°33)
- Texte (ou autre) pour lequel un modèle reconstruit mal le signal (VAE, voir séminaire n°56)
- Texte au sein duquel un modèle doute sur des substitutions de tokens (DATE, voir séminaire n°28)
- Texte dans lequel une ou plusieurs entités entrent en contradiction avec une base de connaissance
- Argumentation fallacieuse et syllogismes

Motivations :

- Détection de propos violents/haineux sur les réseaux sociaux
- Détection de fake news/désinformation, incohérence factuelle
- Détection de panne en analysant des séries de logs

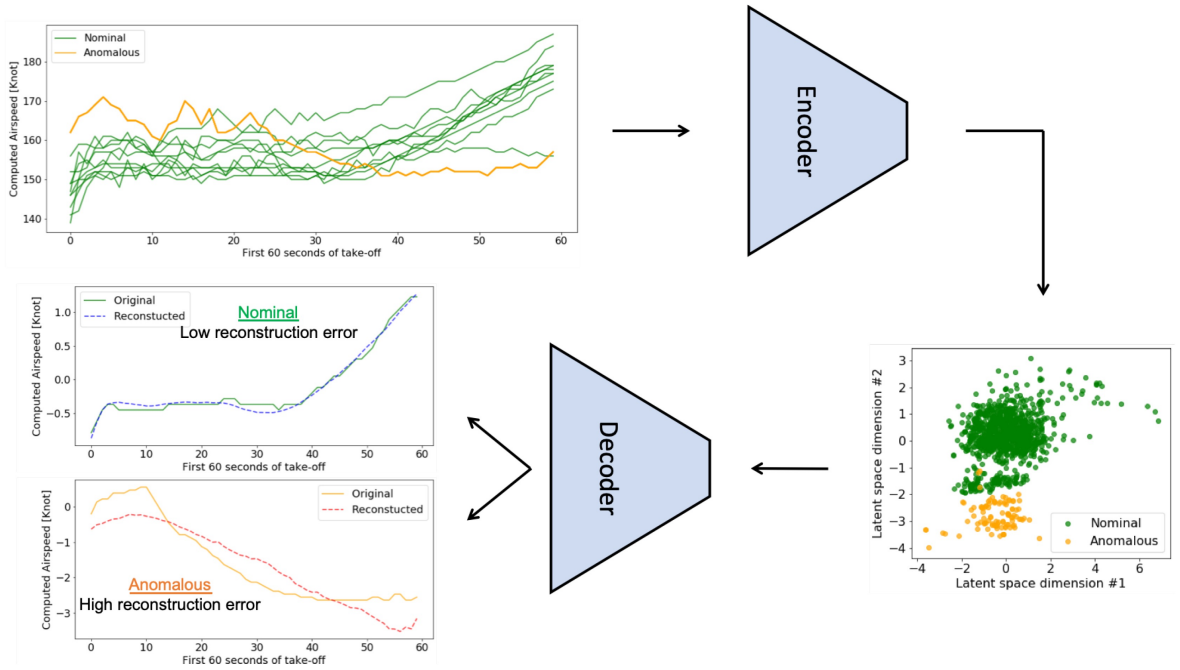
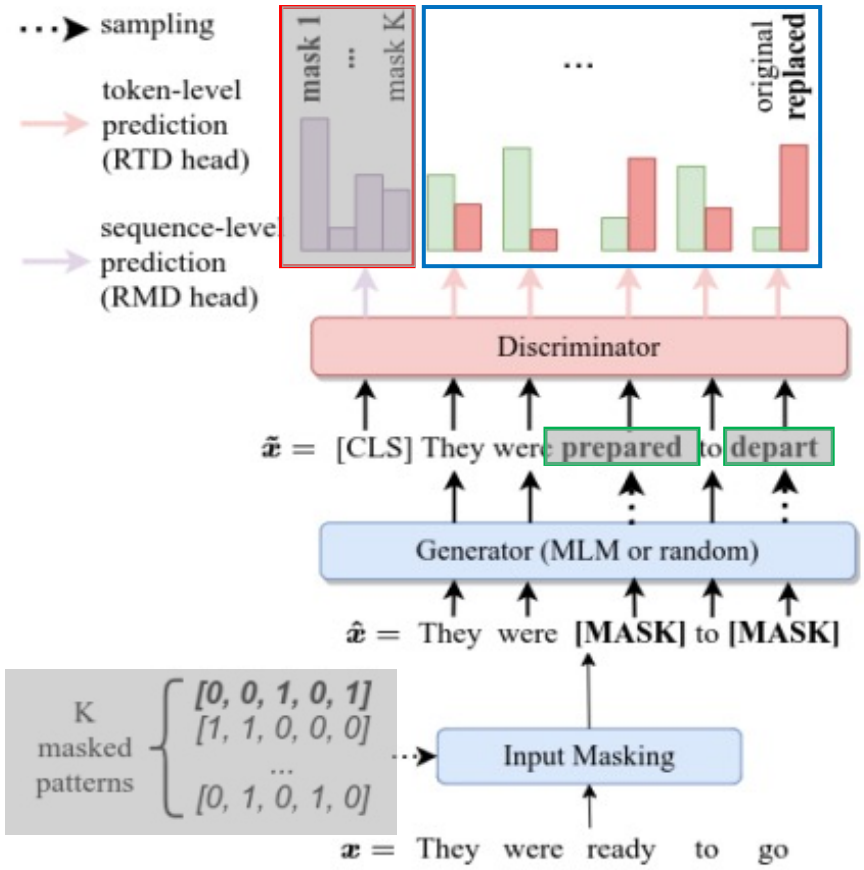
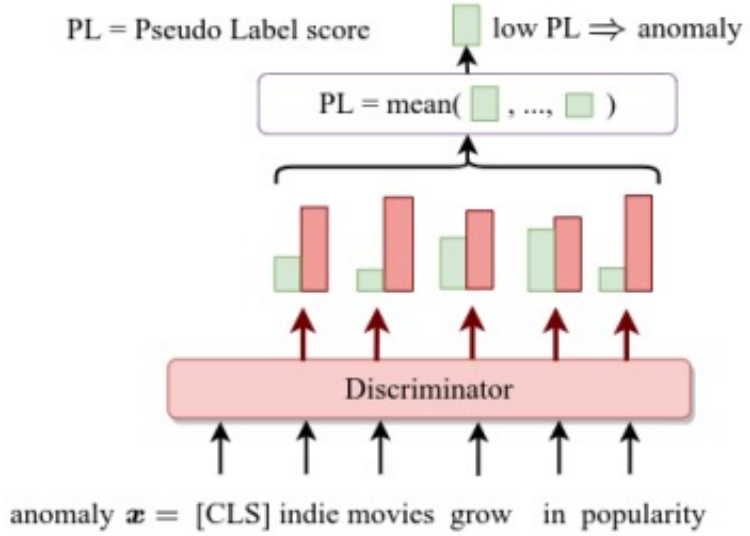


Figure : VAE for anomaly detection

Zoom sur la détection d'anomalies par substitution (DATE)



Inference



$$PL_{RTD}(x) = \frac{1}{T}\sum_{i=1}^{T} P_D(m_i = 0 | \tilde{\boldsymbol{x}}(\boldsymbol{m}^{(0)}); \theta_D)$$

Pendant l'inférence, on utilise uniquement la tête RTD, par soucis de temps de calcul

$$\mathcal{L}_{DATE}(\theta_D, \theta_G; \boldsymbol{x}) = \mu\mathcal{L}_{RMD}(\theta_D; \boldsymbol{x}) + \mathcal{L}_{MLM}(\theta_G; \boldsymbol{x}) + \lambda\mathcal{L}_{RTD}(\theta_D; \boldsymbol{x})$$

Enrichissement (implicite) d'un LM avec pré-entraînement faiblement supervisé – WKLM [Xiong et al., ICLR 2020]

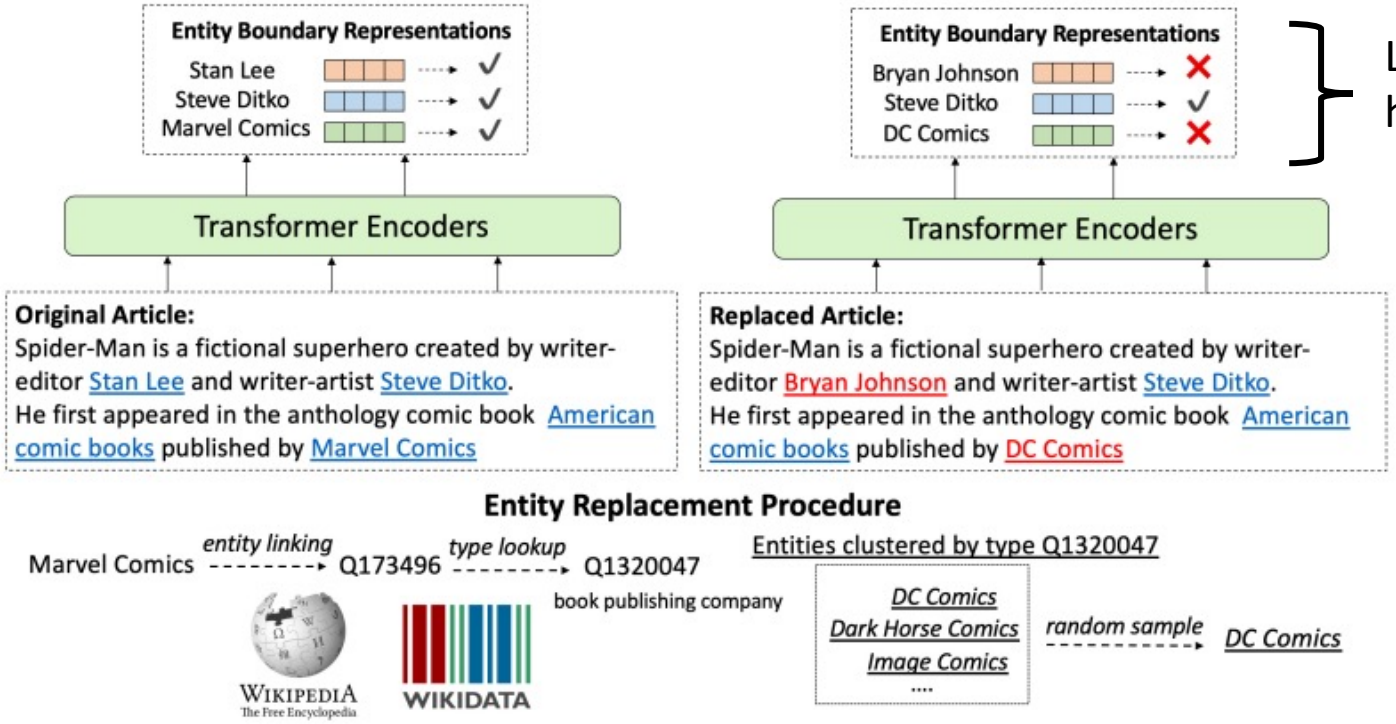


Figure 1: Type-Constrained Entity Replacements for Knowledge Learning.

Les localisations des entités sont **données** (via hyperliens wikipedia)

On minimise la loss suivante :

$$\mathcal{L}_{WKLM} = \mathcal{L}_{MLM} + \mathcal{L}_{RED}$$

RED : Replaced Entity Detection

et

$$\mathcal{L}_{RED} = \mathbb{1}_{e \in E^+} \log P(e|C) + (1 - \mathbb{1}_{e \in E^+}) \log(1 - P(e|C))$$

avec e une entité, C son contexte et E^+ les vraies mentions d'entités

WKLM : Expériences [Xiong et al., ICLR 2020]

Résultats Entity Typing :

Table 5: Fine-grained Entity Typing Results on the FIGER dataset.

Model	Acc	Ma-F1	Mi-F1
LSTM + Hand-crafted (Inui et al., 2017)	57.02	76.98	73.94
Attentive + Hand-crafted (Inui et al., 2017)	59.68	78.97	75.36
BERT baseline (Zhang et al., 2019)	52.04	75.16	71.63
ERNIE (Zhang et al., 2019)	57.19	75.61	73.39
Our BERT	54.53	79.57	74.74
WKLM	**60.21**	**81.99**	**77.00**

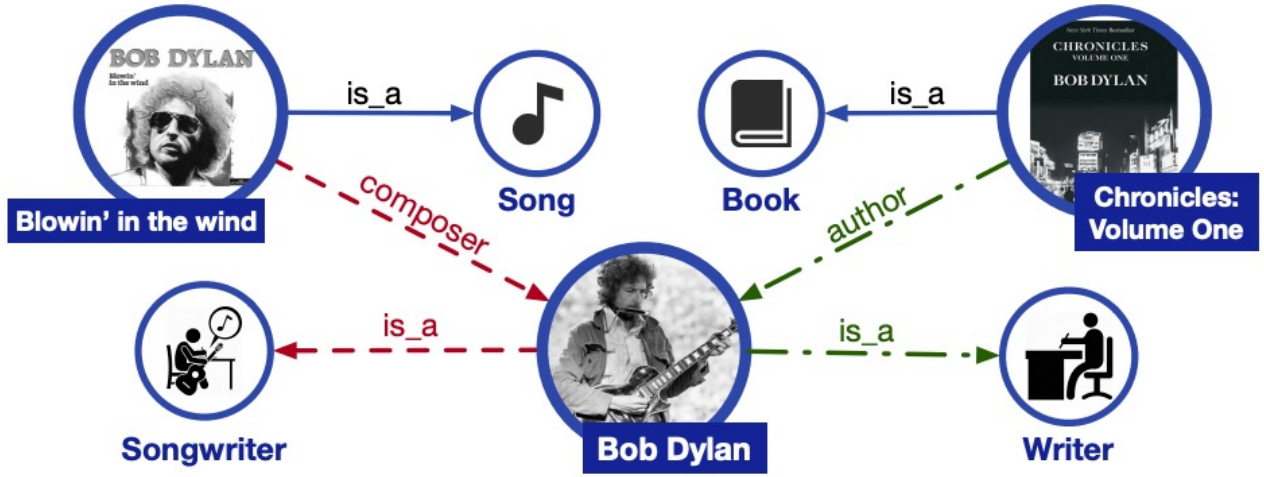
Avantages :

- Conceptuellement simple
- Extraction de connaissance directement à partir du texte de pré-entraînement (données non structurées)
- Pas besoin de modifier l'architecture du modèle
- Meilleur que BERT et ERNIE sur un ensemble de downstream tasks (e.g. entity typing)

Enrichissement (explicite) d'un LM avec base de connaissance externe – ERNIE [Zhang et al., ACL 2019]

But :
Tirer partie de connaissance structurée (graphe de connaissance) pour identifier les occupations de **Bob Dylan** à partir de la phrase ci-contre, i.e. *compositeur* et *écrivain*

Comment ? :
1) Encoder la connaissance et sa structure
2) Fusionner les informations lexicales, syntaxiques et factuelles dans un même espace



Bob Dylan wrote ***Blowin' in the Wind*** in 1962, and wrote ***Chronicles: Volume One*** in 2004.

ERNIE : Représentations d'entités via TransE [Bordes et al., NIPS 2013]

Algorithm 1 Learning TransE

input Training set $S = \{(h, \ell, t)\}$, entities and rel. sets E and L, margin γ, embeddings dim. k.

initialize $\boldsymbol{\ell} \leftarrow \text{uniform}(-\frac{6}{\sqrt{k}}, \frac{6}{\sqrt{k}})$ for each $\ell \in L$

$\boldsymbol{\ell} \leftarrow \boldsymbol{\ell} / \|\boldsymbol{\ell}\|$ for each $\ell \in L$

$\mathbf{e} \leftarrow \text{uniform}(-\frac{6}{\sqrt{k}}, \frac{6}{\sqrt{k}})$ for each entity $e \in E$

loop

  $\mathbf{e} \leftarrow \mathbf{e} / \|\mathbf{e}\|$ for each entity $e \in E$

  $S_{batch} \leftarrow$ sample(S, b) // sample a minibatch of size b

  $T_{batch} \leftarrow \emptyset$ // initialize the set of pairs of triplets

  for $(h, \ell, t) \in S_{batch}$ **do**

    $(h', \ell, t') \leftarrow$ sample$(S'_{(h,\ell,t)})$ // sample a corrupted triplet

    $T_{batch} \leftarrow T_{batch} \cup \{((h, \ell, t), (h', \ell, t'))\}$

  end for

  Update embeddings w.r.t. $\sum_{((h,\ell,t),(h',\ell,t')) \in T_{batch}} \nabla \left[\gamma + d(\boldsymbol{h} + \boldsymbol{\ell}, \boldsymbol{t}) - d(\boldsymbol{h'} + \boldsymbol{\ell}, \boldsymbol{t'})\right]_+$

end loop

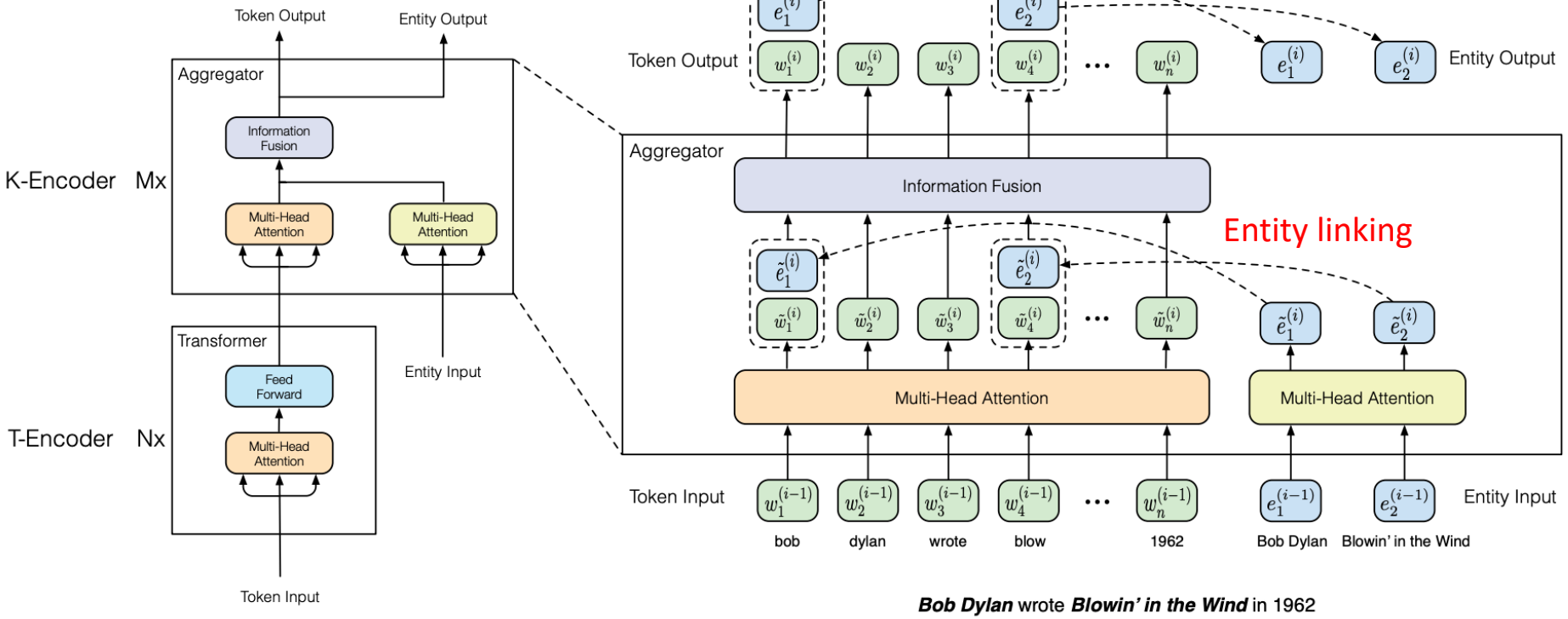
Avantages :

- Peu de paramètres
- Passe bien à l'échelle
  (Wikidata : 5M entités, 24M triplets)

Limitations :

- Gère mal les relations multiples du type *1-to-N, N-to-1, N-to-N*

ERNIE : Fusion des représentations du texte et des entités



(a) Model Achitecture

(b) Aggregator

$$\boldsymbol{h}_j = \sigma(\tilde{\boldsymbol{W}}_t^{(i)}\tilde{\boldsymbol{w}}_j^{(i)} + \tilde{\boldsymbol{W}}_e^{(i)}\tilde{\boldsymbol{e}}_k^{(i)} + \tilde{\boldsymbol{b}}^{(i)}),$$
$$\boldsymbol{w}_j^{(i)} = \sigma(\boldsymbol{W}_t^{(i)}\boldsymbol{h}_j + \boldsymbol{b}_t^{(i)}),$$
$$\boldsymbol{e}_k^{(i)} = \sigma(\boldsymbol{W}_e^{(i)}\boldsymbol{h}_j + \boldsymbol{b}_e^{(i)}).$$

Embeddings initiales de TransE

Tâche de pré-entrainement (***dEA*** : denoising Entity Autoencoder) : Masquer aléatoirement des alignements token-entité pour prédire quelle entité devrait être liée à quel token dans la séquence

$$p(e_j|w_i) = \frac{\exp(\texttt{linear}(\boldsymbol{w}_i^o)\cdot\boldsymbol{e}_j)}{\sum_{k=1}^{m}\exp(\texttt{linear}(\boldsymbol{w}_i^o)\cdot\boldsymbol{e}_k)}$$

ERNIE : Expériences [Zhang et al., ACL 2019]

Résultats Entity Typing :

Model	Acc.	Macro	Micro
NFGEC (Attentive)	54.53	74.76	71.58
NFGEC (LSTM)	55.60	75.15	71.73
BERT	52.04	75.16	71.63
ERNIE	**57.19**	**76.51**	**73.39**

Table 2: Results of various models on FIGER (%).

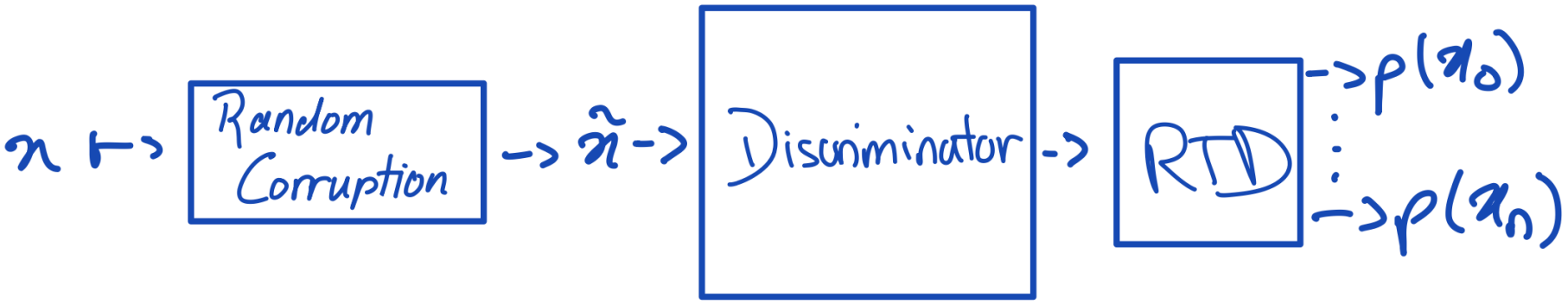
Avantages :

- Meilleures performances sur des « knowledge intensive » tasks (e.g. entity typing, relation classification)

Limitations :

- Représentations de TransE (statiques)
- Nécessite un alignement entre les mentions d'entités dans le texte et leur ID correspondant dans le graph

Détection de substitution d'entités avec LM enrichi – DATE

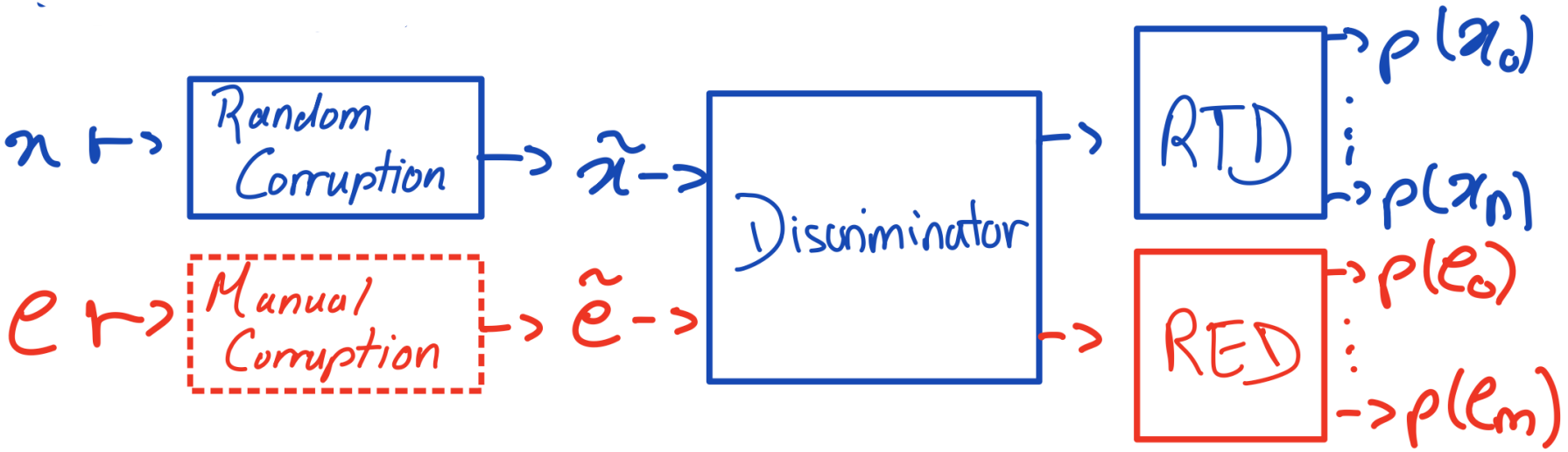


x : input text of length n

Type d'anomalies ciblé : Texte pour lequel le discriminant, en moyenne, doute sur la substitution de token ; e.g. Articles de politique VS articles de philosophie / religion

Score de normalité : $$s(x) = \frac{1}{n}\sum_{i=1}^{n} p_{RTD}(x_i)$$

Détection de substitution d'entités avec LM enrichi – DATE&WKLM

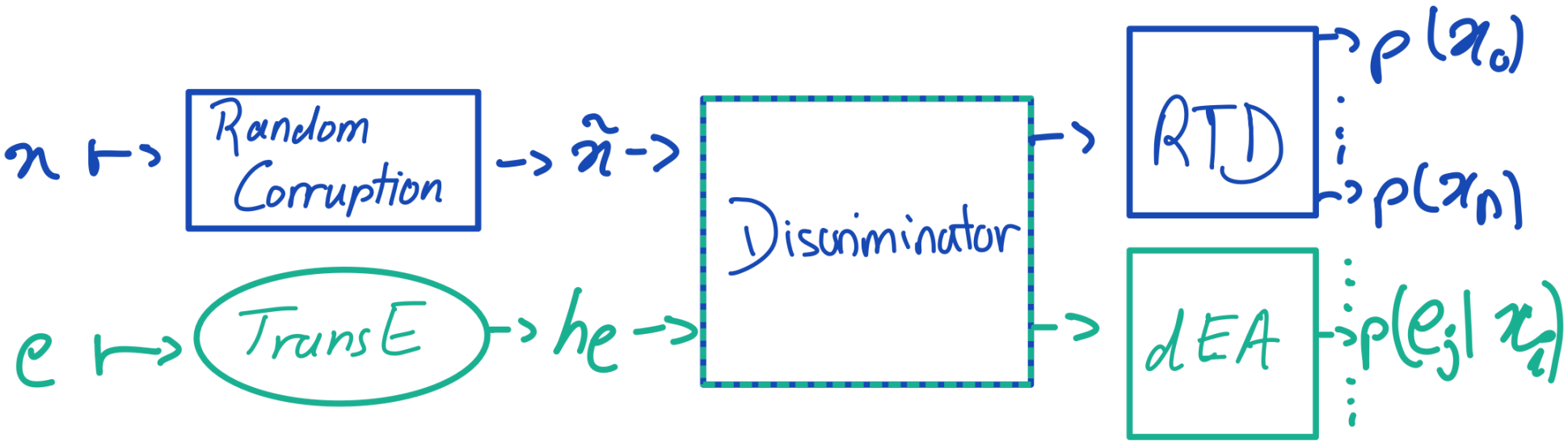


x : input text of length n
e : input entities sets containing m entities

Type d'anomalies ciblé : Texte pour lequel le discriminant, en moyenne, doute sur la substitution d'entités ; e.g. remplacer Emmanuel Macron par Mao Tse-Tong

Score de normalité : $$s(x) = \frac{1}{n}\sum_{i=1}^{n} p_{RTD}(x_i) + \frac{\lambda}{m}\sum_{j=1}^{m} p_{RED}(e_j)$$

Détection de substitution d'entités avec LM enrichi – DATE&ERNIE



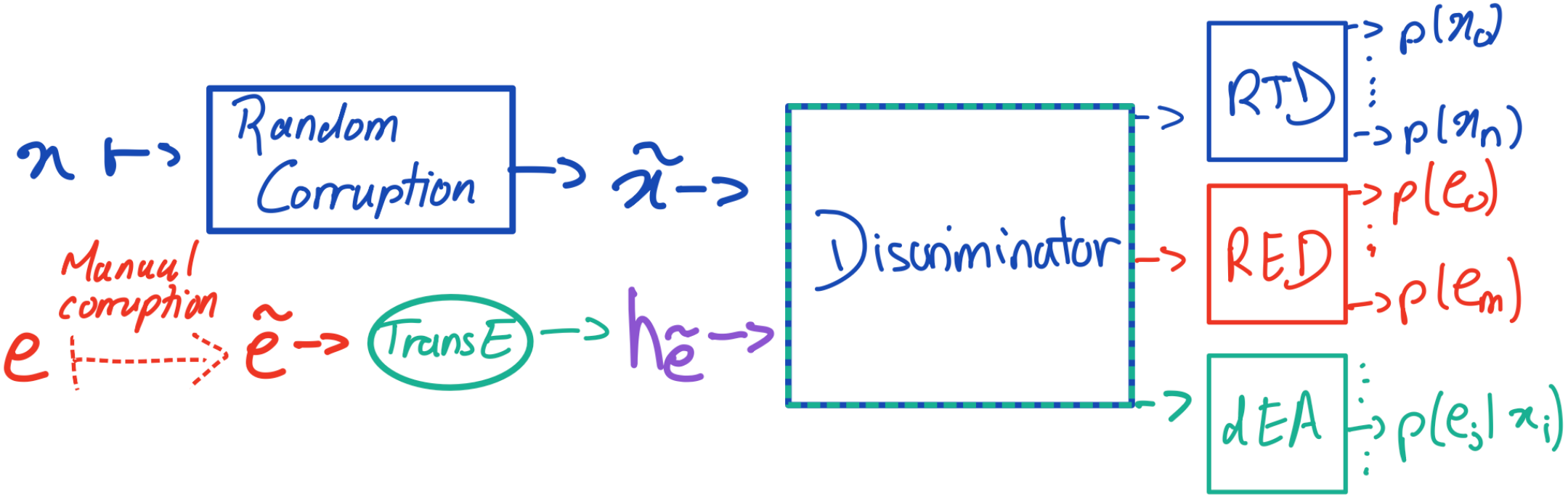
x : input text of length n
e : input entities sets containing m entities

Type d'anomalies ciblé : Texte pour lequel le discriminant prédit une entité, associée à un token, qui n'est pas celle observée, i.e. une violation de relation dans le graphe de connaissance

Score de normalité :

$$s(x) = \frac{1}{n}\sum_{i=1}^{n} p_{RTD}(x_i) + \frac{\lambda}{m}\sum_{j=1}^{m} p_{dEA}(e_j|x_{e_j})$$

Détection de substitution d'entités avec LM enrichi – DATE&WKLM&ERNIE



Score de normalité : $$s(x) = \frac{1}{n}\sum_{i=1}^{n} p_{RTD}(x_i) + \frac{\lambda}{m}\sum_{j=1}^{m} p_{RED}(e_j)$$

Question : dEA aide RED de manière soft ?

Expériences

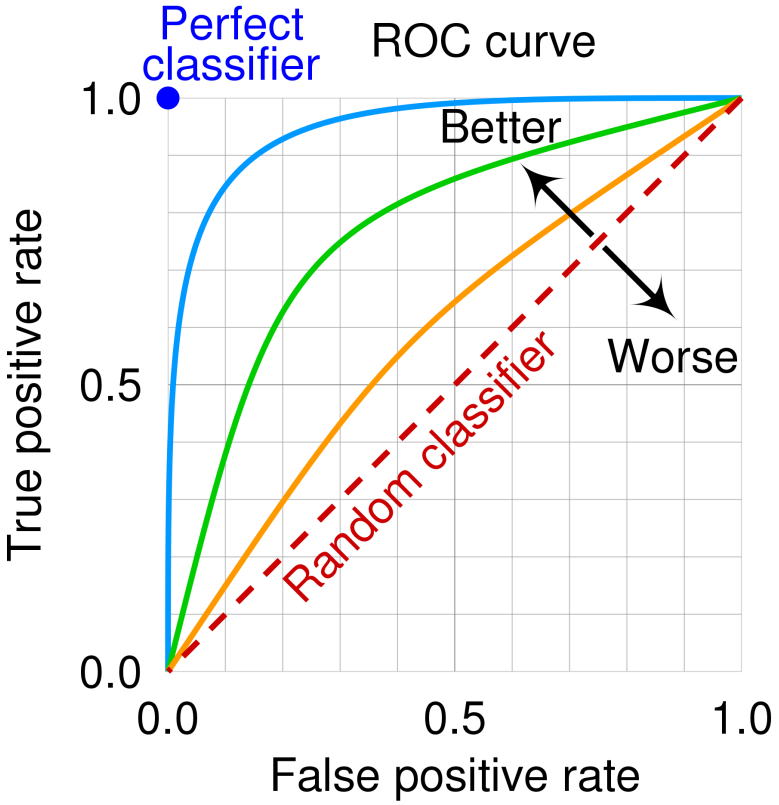
Données : Abstracts Wikivitals avec
~10 entités par texte
N_train = 16706
N_test_in = 4178
N_test_out = 3866 - 4956

	AD_semantic	AD_entities
inliers classes	"people", "geography"	"people", "geography"
outliers classes	"mathematics", "philosopy", "religion"	"people", "geography"
outliers entities	no corruption	50% of corrupted entities

Table : Evaluatior

Method	Pre-training heads	AD_semantic	AD_en
DATE	RTD	93,5	66,8
DATE_wklm	RTD + RED	86,6	69,9
DATE_ernie	RTD + dEA	90.4	61.6
	10 RTD + dEA	87,0	69,5
DATE_ernie&wklm	RTD + RED + dEA	87,1	71,3
	10 RTD +10 RED + dEA	84,8	72,5

Table : AUROC results on both *AD_semantic* and *AD_entities* tasks
RTD : *Replaced Token Detection*
RED : *Replaced Entity Detection*
dEA : *Denoising Auto-Encoder* (ERNIE)



Difficultés restantes à surmonter

Court terme :

- Caractériser proprement les différents types d'anomalies que ciblent WKLM et ERNIE indépendamment, et ensemble (exemples qualitatifs) ?
- Evaluations sur des anomalies factuelles plus complexes (conservation du type)
- Score d'anomalies utilisé ?

Moyen/long terme :

- Directement utiliser le graphe de connaissance pour apprendre nos représentations d'entités (CokeBERT) en lieu et place de TransE
- Désambiguïsation d'entités ?
- Peut-être serait-il préférable de garder indépendant les modules gérant le texte et la connaissance (pas besoin de re-pré-entraîné à chaque évolution du graphe) ?