Projet TAL : Evaluation du moteur de traduction neuronale OpenNMT sur un corpus parallèle anglais-français

RAHOUAL Wassim – FORT Adrien

I. Introduction

La traduction automatique a toujours été un défi pour les linguistes et les informaticiens, car la compréhension de la langue est complexe et contextuelle. Les approches traditionnelles de traduction automatique basées sur des règles et des dictionnaires ne parviennent pas à capturer les nuances et les variations dans les langues et les contextes. La traduction automatique neuronale (NMT) est une approche plus récente qui utilise des réseaux de neurones pour apprendre à traduire des phrases d'une langue à une autre.

Cependant, la traduction automatique neuronale pose également des défis. Tout d'abord, il nécessite des quantités massives de données d'entraînement pour entraîner le réseau de neurones. De plus, la qualité de la traduction dépend de la qualité et de la diversité des données d'entraînement. Les erreurs de traduction peuvent également se produire en raison de l'ambiguïté des langues et des contextes, ainsi que des différences dans la syntaxe et la grammaire entre les langues source et cible.

En outre, la traduction automatique neuronale peut poser des problèmes de confidentialité et de sécurité, car les données sensibles peuvent être traitées par des réseaux de neurones et stockées dans des systèmes externes. Enfin, les erreurs de traduction peuvent avoir des conséquences graves dans certaines situations, telles que les domaines médical et juridique.

Malgré ces défis, la traduction automatique neuronale continue de s'améliorer grâce aux progrès de la technologie et à l'augmentation des données d'entraînement.

II. Présentation du moteur de traduction neuronale OpenNMT

OpenNMT est une plate-forme open source pour la traduction neuronale automatique (NMT) qui utilise des réseaux de neurones pour apprendre à traduire des phrases d'une langue à une autre. Le fonctionnement d'OpenNMT est basé sur l'architecture de réseaux de neurones récurrents (RNN) qui permet à la plate-forme d'apprendre à traduire des phrases de manière contextuelle, en utilisant les informations contenues dans les phrases précédentes pour améliorer la traduction.

La plate-forme OpenNMT fonctionne en deux étapes principales : l'entraînement et la traduction. Pendant la phase d'entraînement, les réseaux de neurones sont entraînés à partir d'un ensemble de données de paires de phrases dans les langues source et cible. Les données d'entraînement sont utilisées pour ajuster les poids des connexions neuronales du réseau de neurones, afin de minimiser l'erreur de traduction.

Une fois que le réseau de neurones a été entraîné, il peut être utilisé pour la traduction. Lors de la traduction, la phrase source est entrée dans le réseau de neurones, qui génère une séquence de mots dans la langue cible. La qualité de la traduction dépend de la qualité de l'entraînement initial et de la quantité de données d'entraînement. Les utilisateurs peuvent également ajuster les paramètres du modèle de traduction pour améliorer la qualité de la traduction en fonction de leurs besoins spécifiques.

En somme, OpenNMT est une plate-forme de traduction neuronale automatique puissante et flexible qui utilise des réseaux de neurones pour apprendre à traduire des phrases d'une langue à une autre. Grâce à son architecture RNN et à sa capacité à s'adapter à différents ensembles de données, OpenNMT est capable de fournir des traductions précises et de haute qualité pour une variété de langues et de contextes.

III. Objectif du projet

L'objectif de ce projet est d'analyser et d'évaluer un système de traduction neuronale OpenNMT. Dans un premier temps, nous allons prendre en main l'outil OpenNMT sur deux corpus, toy-ende qui propose un des corpus de Train, Dev et Test en anglais et son équivalent en allemand. Ensuite, nous testerons l'outil pour une traduction de l'anglais vers le français sur le corpus Europarl avec une période d'entrainement sur 10k phrases, un dev sur 1k phrase et un test sur 500 phrases. Enfin, nous utiliserons le score BLEU, un algorithme d'évaluation de la qualité du texte qui a été traduit d'une langue naturelle à une autre, pour évaluer les résultats sur ces corpus.

Score BLEU	Interprétation
< 10	Traductions presque inutiles
10 à 19	L'idée générale est difficilement compréhensible
20 à 29	L'idée générale apparaît clairement, mais le texte comporte de nombreuses erreurs grammaticales
30 à 40	Résultats compréhensibles à traductions correctes
40 à 50	Traductions de haute qualité
50 à 60	Traductions de très haute qualité, adéquates et fluides
> 60	Qualité souvent meilleure que celle d'une traduction humaine

La deuxième partie de ce projet est dédié à la comparaison du modèle sur des corpus parallèles en **formes fléchies** à large échelle, avec un modèle sur des corpus parallèles **en lemmes** à large échelle, à l'aide des corpus Europarl et EMEA.

Pour cela, nous utiliserons différents outils à savoir Python et la librairie PyTorch qui sera utilisée dans OpenNMT-py. Nous utiliserons un script perl pour calculer le score BLEU des modèles.

OpenNMT-py est une version PyTorch du projet OpenNMT, un framework de traduction neuronale open source. C'est un outil qui a été conceptualisé pour être facile d'utilisation et d'expérimentation afin de comprendre la traduction neuronale.

IV. Evaluation du moteur de traduction neuronale OpenNMT sur un corpus en formes fléchies

La première étape lorsque l'on utilise OpenNMT est de préparer les données. Pour cela, nous allons nous aider d'un fichier YAML, qui contiendra les chemins des fichiers utiles, considérons-le comme un fichier de configuration.
Le dataset toy-ende possède des données parallèles source (src) et target (tgt) qui contiennent chacune une phrase par ligne avec des jeton séparés par des espaces pour un total de 10k phrases. Les fichiers de validations sont utilisés afin d'évaluer la convergence de l'entrainement. Il contient dans notre cas 5k phrases.

À partir de cette configuration, nous pouvons construire le ou les vocabulaires qui seront nécessaires pour entraîner le modèle à l'aide de la commande suivante :

```
onmt_build_vocab -config toy_en_de.yaml -n_sample 10000
```

-n_sample représente ici le nombre de lignes échantillonnées de chaque corpus pour construire le vocabulaire.

L'étape suivante est l'entrainement du modèle. Pour cela, nous précisons dans le fichier de configuration le chemin des fichiers de vocabulaires créées, si l'on compte utiliser un processeur graphique pour entrainer le modèle (rendu plus rapide), et la précisions des étapes d'entrainement et de validation. Cette configuration consiste en un LSTM à 2 couches avec 500 unités cachées sur le codeur et le décodeur. Nous lançons ensuite la commande, ce qui nous donneras notre modèle :

```
onmt_train -config toy_en_de.yaml
```

La dernière étape est la traduction, nous lançons simplement la commande suivante :

```
onmt_translate -model toy-ende/run/model_step_1000.pt -src toy-ende/src-
test.txt -output toy-ende/pred_1000.txt -gpu 0 -verbose
```

model_stepp_1000.pt est le modèle précédemment créé.

Voici les résultats de notre traduction ainsi que le score BLEU correspondant.

Texte source de test :

```
Orlando Bloom and Miranda Kerr still love each other
Actors Orlando Bloom and Model Miranda Kerr want to go their separate ways .
However , in an interview , Bloom has said that he and Kerr still love each other .
Miranda Kerr and Orlando Bloom are parents to two-year-old Flynn .
Actor Orlando Bloom announced his separation from his wife , supermodel Miranda Kerr .
In an interview with US journalist Katie Couric , which is to be broadcast on Friday (local time), Bloom said , &quot; sometimes life doesn &apos;t
He and Kerr still love each other , emphasised the 36-year-old .
&quot; We &apos;re going to support one another and love each other as parents to Flynn &quot; .
Kerr and Bloom have been married since 2010 . Their son Flynn was born in 2011 .
Jet makers feud over seat width with big orders at stake
```

Texte traduit avec notre modèle :

```
<unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <un
<unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <un
Es gibt ein , dass die Kommission die Europäischen Union , die die Mitgliedstaaten , die die Mitgliedstaaten , die die Mitgliedstaaten , die die Mi
<unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <un
<unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <unk> , <un
In diesem Parlaments , die wir der Europäischen Union , die die Europäischen Union , die in dem Europäischen Union , die in dem Europäischen Union
```

Score BLEU :

```
BLEU = 0.00, 0.6/0.0/0.0/0.0 (BP=1.000, ratio=9.573, hyp_len=587523, ref_len=61376)
```

Les résultats sont terribles, ce qui est plutôt normal avec des corpus de cette taille-là. De plus, le corpus utilisé provient d'un cadre très professionnel à savoir une session du parlement européen. Les données du texte de test portent sur les nouvelles de deux acteurs américains. La différence de contexte des deux corpus ne joue pas en la faveur de notre modèle.

Les résultats sont similaires pour l'utilisation du moteur OpenNMT sur le corpus Europarl :

```
BLEU = 0.00, 0.0/0.0/0.0/0.0 (BP=1.000, ratio=4.125, hyp_len=47125, ref_len=11424)
```

```
Il n'est pas de la Commission et de la Commission et de la Commission.
Il faut la Commission de la Commission de la Commission et de la Commission et de la Commission et de la Commission et
La Commission ne peut pas le cas de la Commission.
Nous avons besoin de la Commission de la Commission et de la Commission de la Commission et de la Commission et de la C
Il faut la Commission de la Commission et de la Commission de la Commission et de la Commission.
Nous avons besoin de la Commission de la Commission et de la Commission et de la Commission.
Nous avons besoin de la Commission de la Commission et de la Commission et de la Commission.
C' est de la Commission de la Commission de la Commission et de la Commission et de la Commission et de la Commission.
```

Les deux traductions (allemands et français) comportent une répétition du mot Commission, mot pourtant répété que 1557 fois sur le corpus d'entrainement français sur 10k phrases, nous n'arrivons pas à expliquer ce résultat.

V. Evaluation sur des corpus parallèles en formes fléchies à large échelle

Nous allons maintenant expérimenter de la façon suivante, découpé en deux runs :

N° du run	Apprentissage (nombre de phrases)	Tuning (nombre de phrases)
1	100K (Europarl)	3,75K (Europarl)
2	100K+10K (Europarl+Emea)	3,75K (Europarl)

Ci-dessous, les résultats pour les corpus Europarl_train_100k, Europarl_dev_3750 et Europarl_test_500.

Le score bleu est encore une fois catastrophique.

```
BLEU = 0.00, 0.1/0.0/0.0/0.0 (BP=1.000, ratio=1.005, hyp_len=14625, ref_len=14552)
```

```
The vote will take place tomorrow at 12.30 p.m.
Financial assistance for Member States' balances of payments
The next item is the report (A5-0277/2000) by Mrs Sartori, on behalf of the Committee on Economic and Monetary Affairs, on the Commission report to
Madam President, ladies and gentlemen, today, on behalf of the Committee on Economic and Monetary Affairs, I am tabling a proposal for a resolution
The financial assistance is currently governed by Council Regulation No 1969 of 1988.
This mechanism can be activated either at the request of a Member State or under Article 119 of the Treaty establishing the European Community, a pr
The regulation and the proposal for a regulation have merged two mechanisms (medium-term financial assistance and the Community loan mechanism inter
This proposal changes the mechanism' s ceiling.
The fact that, at the moment, only three countries can benefit from it suggests that the amount hitherto available - EUR 16 billion - can be reduced
Therefore, although the increase in convergence of States reduces the likelihood that the mechanism will be used, it must still have sufficient reso
This is why we propose to reduce the amount available from EUR 16 billion to EUR 12 billion.
This resolution also envisages the possibility of considering creating a special mechanism providing financial assistance to the balance of payments
Clearly, this is a possibility for the future and therefore does not have a bearing on the present situation, but it is a possibility which I consid
Mr President, I would like to thank Mrs Sartori for her report, which we have found very valuable.
```

```
La Commission ne peut pas être <unk>
Il est important que nous ne pouvons pas.
Le Conseil est lieu que la Commission ne peut pas être <unk> à ce que nous sommes en train de l' Union européenne de l' Union européenne de l' Union
Madame la Présidente, chers collègues, je voudrais vous dire que vous avez dit à cette question de l' Assemblée de la commission de la commission de
Le Parlement est lieu que la Commission ne peut pas être <unk>
Il est important que nous ne pouvons pas être <unk> à ce que nous ne pouvons pas nous en être de l' Union européenne en matière de l' Union européen
Le Conseil est lieu que les États membres de l' Union européenne de l' Union européenne de l' Union européenne de l' Union européenne de l' Union eu
Il est important que nous ne pouvons pas.
```

Certaines phrases sont reconnues comme Madam President et Madame la Présidente par exemple, très proche syntaxiquement. Jusqu'ici nous avons trois hypothèses que nous n'avons pas pu vérifier pour le moment en attente des résultats de nos camarades, et une erreur de notre part nous semble impossible car il s'agit seulement de recopier les lignes de commandes et les instructions d'un tutoriel créée par la communauté de OpenNMT-py. De plus, le corpus de test correspond cette fois-ci au cadre du corpus de train.

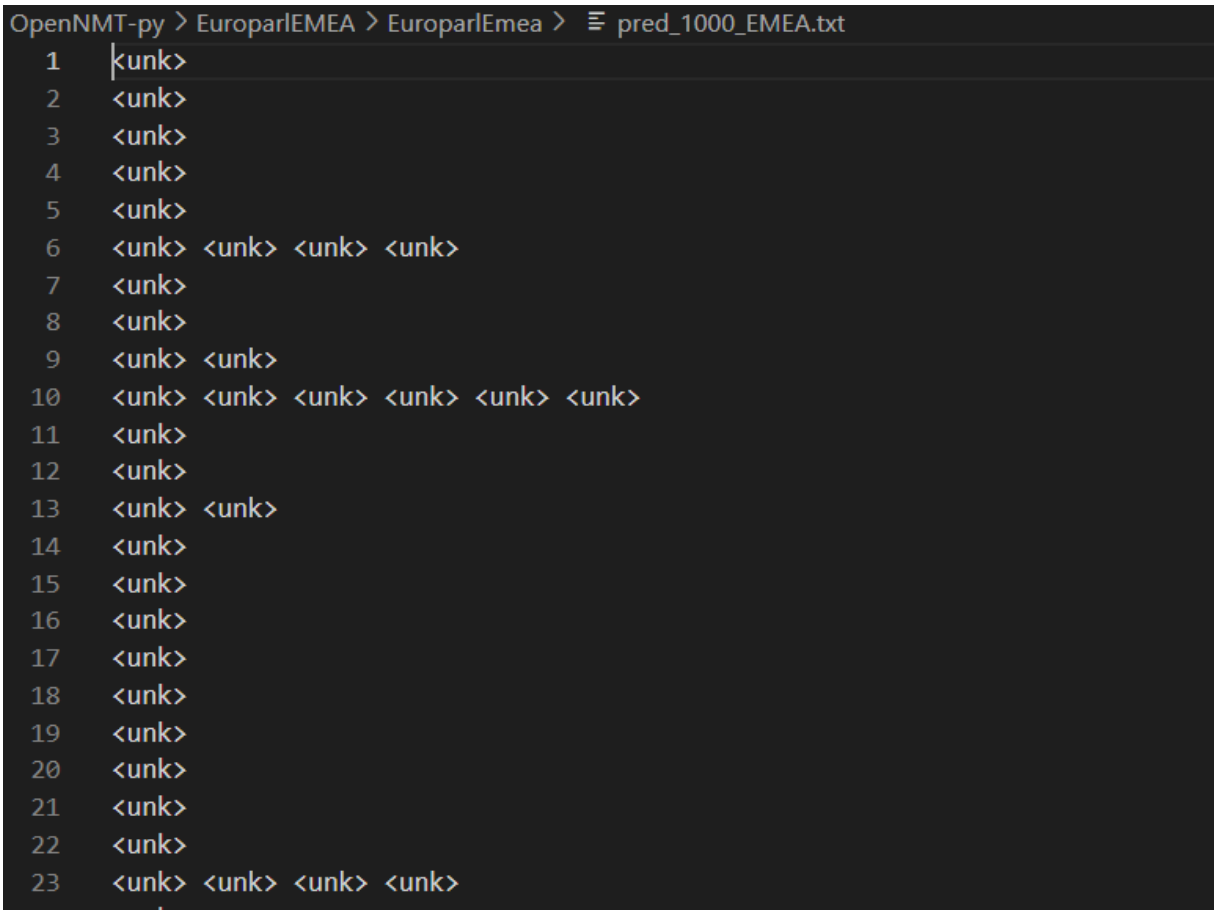
Après l'ajout du corpus EMEA, les résultats sont encore du même ordre, à savoir un score BLEU d'environ 0.00 sur le corpus de test Europarl

```
BLEU = 0.00, 0.3/0.0/0.0/0.0 (BP=0.878, ratio=0.885, hyp_len=12873, ref_len=14552)
```

Et de 0.00 sur le corpus de test EMEA

```
BLEU = 0.00, 0.0/0.0/0.0/0.0 (BP=0.416, ratio=0.533, hyp_len=1130, ref_len=2120)
```

Le fichier étant rempli de unknown



Maintenant, nous allons effectuer une autre approche avec des données d'entrainements préparées. Nous allons passer par plusieurs traitements qui sont les suivants :

La tokenisation soit le processus de séparation d'une chaîne de caractères en une suite de symboles discrets, des mots ou des tokens. L'ensemble des symboles observés dans le dataset constitue le vocabulaire. Des espaces peuvent être insérés entre certaines lettres pour minimiser la diversité des formes du vocabulaire. La transformation des majuscules en minuscules et enfin le nettoyage du corpus en limitant la longueur des phrases à 80 caractères. Nous aurions pu utiliser d'autres prétraitements comme une sérialisation ou encore une dé-accentuation mais ce n'était pas l'objectif dans ce projet.

Les résultats sont nettement mieux, avec pour la run numéro 1 correspondant au corpus Europarl un score bleu de :

```
BLEU = 18.01, 44.8/22.6/13.2/7.9 (BP=1.000, ratio=1.100, hyp_len=17098, ref_len=15547)
```

pour un corpus appartenant au même domaine et un score bleu de 0.00 pour un corpus hors domaine. Le modèle de traduction est convenable sur des corpus du même domaine mais catastrophique sur des corpus de domaines différents.

Pour la run numéro 2, on obtient un score BLEU de 20.12 pour un corpus du même domaine et un score BLEU de 81.19 pour un corpus hors domaine. Le modèle de traduction s'est amélioré sur un corpus du même domaine avec l'ajout du corpus d'entrainement EMEA mais n'est pas encore totalement au point. Cependant sur un corps hors-domaine, celui-ci est très performant.

Un tableau qui résume les différents résultats

	Score BLEU sans pré-traitement	Score BLEU après pré-traitement même domaine	Score BLEU après pré-traitement même hors domaine
Run n°1	0.00	18.01	20.12
Run n°2	0.00	0.00	81.19

VI. Evaluation sur des corpus parallèles en lemmes à large échelle

L'objectif de cette partie était de tester OpenNMT sur des corpus en lemmes à large échelle. Afin de lemmatiser les corpus, nous avons écrit deux scripts python, l'un pour les corpus en français et l'autre pour les corpus en anglais.

Ci-dessous, le script pour la lemmatization d'un corpus anglais.

```python
import sys
import nltk
nltk.download('punkt')
from nltk.stem import WordNetLemmatizer
from nltk.tokenize import word_tokenize

nltk.download('wordnet')

lemmatizer = WordNetLemmatizer()

if (len(sys.argv) != 3):
    raise ValueError('Mauvaise utilisation du script : python lemmatize_english_text.py input_file output_file')

input_file = sys.argv[1] # fichier d'entrée
output_file = sys.argv[2] # fichier de sortie

def lemmatize_file(input_file, output_file):
    with open(input_file, 'r', encoding="utf-8") as f_in, open(output_file, 'w') as f_out:
        for line in f_in:
            tokens = word_tokenize(line) # tokenisation du corpus
            lemmatized_words = [lemmatizer.lemmatize(token) for token in tokens]  # lemmatisation des mots
            lemmatized_line = ' '.join(lemmatized_words)  # Ajout des mots lemmatize
            f_out.write(lemmatized_line + '\n')  # écriture de la ligne complète lemmatisée dans le fichier de sortie

lemmatize_file(input_file, output_file)
```

On obtient les résultats suivants :

	Score BLEU lemmatisation même domaine	Score BLEU lemmatisation hors domaine
Run n°1	24.49	12.72
Run n°2	0.21	49.52

On s'aperçoit que la lemmatisation apporte un intérêt pour des corpus du même domaine mais présente une baisse d'efficacité pour des corpus de domaines différents.

VII. Points forts, limitations et difficultés rencontrées

On s'aperçoit que sur des données non préparés au préalable, les résultats de OpenNMT sont catastrophiques. Nous avons rencontré des difficultés sur cette partie car nous partions en vacances

et nous avons effectué le projet avant de recevoir le mail précisant les étapes de pré-traitement à suivre, il a donc fallu recommencer intégralement le projet et le rapport une fois de retour de vacances.

VIII. Organisation

Pour ce projet, nous avons dans un premier temps comme expliqué lors du rapport chacun fait tout une première version du projet et comparé nos résultats à la fin. Enfin, lors de la deuxième version, Wassim RAHOUAL s'est occupé de la préparation des données et de la partie en mode fléchie et Adrien FORT s'est occupé du script python de lemmatization ainsi que la partie en mode lemme. Nous avons ensuite rédigé ce rapport ensemble et Adrien s'est occupé de mettre le projet sur Github : https://github.com/Adrien2209/TAL_OpenNMT_Project.git