Rapport final du projet

Cours de Traduction Automatique et Assistée

HUL Maria
SKRZYNIARZ Agata
INALCO

1. Introduction

Le domaine de la traduction automatique a considérablement évolué ces dernières années, grâce au développement des outils de traduction assistée par ordinateur. Ces outils exploitent différents types de systèmes de traduction automatique pour réaliser des traductions de haute qualité de manière efficace. Parmi eux, la traduction automatique neuronale se distingue en s'appuyant sur les principes de l'apprentissage automatique et des réseaux de neurones pour améliorer la qualité des traductions. Contrairement aux modèles basés sur des règles ou des approches statistiques, elle utilise des techniques de deep learning pour effectuer des traductions plus nuancées qui respectent le contexte et la signification originale des textes.

Les systèmes de traduction automatique neuronale se basent principalement sur une architecture de type encodeur-décodeur. L'encodeur traite le texte source pour créer une représentation intermédiaire, tandis que le décodeur utilise cette représentation pour générer le texte cible. Un aspect crucial de ces systèmes est l'intégration des mécanismes d'attention, qui permettent au modèle de se concentrer sur des parties spécifiques du texte source durant la traduction, ce qui améliore la pertinence de la traduction effectuée. De plus, ces systèmes utilisent des embeddings de mots (Word Embeddings), qui représentent les mots sous forme de vecteurs dans un espace vectoriel où les distances illustrent les similarités sémantiques entre eux. Cette approche aide le système à mieux comprendre les nuances sémantiques de la langue.

Suivant les progrès constants, la traduction automatique neuronale a dépassé les systèmes traditionnels en termes d'efficacité et de précision, et elle continue son développement en intégrant des technologies avancées telles que les RNN, LSTM, et les Transformers, qui contribuent également à l'amélioration de la qualité de traduction.

2. Présentation du moteur de traduction neuronale OpenNMT

OpenNMT est un écosystème open source pour la traduction automatique neuronale et l'apprentissage séquentiel neuronal. Lancé en décembre 2016 par le groupe NLP de Harvard et SYSTRAN, le projet a depuis été utilisé dans plusieurs applications de recherche et d'industrie. Il est basé sur des réseaux de neurones récurrents (RNN) et des transformateurs, permettant des traductions de haute qualité.

Le moteur de traduction neuronal d'OpenNMT est basé sur des réseaux neuronaux profonds, principalement des RNN et des transformateurs. Les RNN, tels que LSTM et GRU, traitent des séquences de données en préservant les relations à long terme, ce qui est crucial pour la traduction de textes. Les transformateurs utilisent un mécanisme d'attention qui permet au modèle de se concentrer sur les parties pertinentes de la séquence d'entrée, améliorant ainsi la qualité de la traduction. OpenNMT forme des modèles sur de vastes ensembles de paires de phrases, puis les applique pour générer des traductions de nouveaux textes.

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

Pour évaluer le moteur de traduction neuronale OpenNMT, nous avons procédé à deux types d'évaluations : une utilisant le corpus en formes fléchies et une autre avec des lemmes. La première évaluation a consisté à traiter le corpus en extrayant les formes originales des textes, sans lemmatisation. Pour cette évaluation, nous avons utilisé des corpus parallèles en français et en anglais, notamment Europarl et Emea, téléchargés depuis leurs sites officiels. Pour l'entraînement, nous avons préparé deux ensembles : le premier comprenait 100 000 phrases du domaine général extraites d'Europarl, et le second ajoutait 10 000 phrases du domaine médical issues d'Emea pour diversifier les données d'entraînement. La validation du modèle a été réalisée avec 3750 phrases d'Europarl, débutant à partir de la phrase ayant le numéro 100 001.
Afin de tester le moteur et de réaliser l'évaluation complète, nous avons sélectionné deux ensembles de données : 500 phrases tirées d'Europarl, débutant au rang 103 751 pour représenter les données de domaine général, et 500 phrases issues d'Emea, avec un début au rang 10 001, pour représenter les données hors domaine. Cela nous a permis d'évaluer la capacité du modèle à s'adapter à de nouveaux contextes. Nous avons développé le script Python *extraire_phrases.py* pour faciliter l'extraction de ces données, qui est disponible sur le dépôt GitHub de notre projet.
Pour évaluer les deux modèles que nous avons entraînés, nous avons employé le score BLEU, une métrique essentielle dans l'évaluation de la traduction automatique. Cette métrique a facilité la comparaison entre les textes traduits par nos modèles et les versions traduites par des humains, nous permettant ainsi de distinguer les points forts et les limitations du système de traduction OpenNMT. Le score BLEU est exprimé en pourcentage de 0 à 100, où un score plus élevé indique une meilleure correspondance avec les traductions de référence.
L'évaluation a été divisée en quatre parties : tout d'abord, nous avons testé le premier modèle entraîné avec des phrases issues du domaine général du corpus Europarl. Ensuite, ce même modèle a été évalué en utilisant des données hors domaine du corpus Emea. Puis, un second modèle a été testé avec des phrases du domaine général extraites également du corpus Europarl. Enfin, ce modèle a également été évalué sur des données hors domaine du corpus Emea. Les résultats de ces évaluations indiquant les scores BLEU pour chaque partie sont présentés dans le Tableau 1. ci-dessous :

	Corpus de domaine	Corpus hors-domaine
Premier modèle entraîné	18.0	0.3
Deuxième modèle entraîné	18.8	15.3

Tableau 1.

Comme nous pouvons le voir dans le Tableau 1, les scores BLEU sont plus élevés pour le corpus en domaine pour les deux modèles, ce qui indique que les textes en domaine traduits par les modèles sont plus pertinents par rapport aux textes de référence que les textes hors-domaine. Ce qui est intéressant, c'est que le score du deuxième modèle pour le corpus hors-domaine est beaucoup plus élevé que celui du premier modèle, même si les données d'entraînement étaient enrichies de seulement 10 000 phrases provenant du corpus hors-domaine pour le deuxième entraînement.

4. Evaluation du moteur de traduction neuronale OpenNMT sur un corpus en lemmes

Nous avons créé un programme qui lemmatise des corpus en anglais et en français. Pour la partie anglaise, nous avons utilisé le lemmatiseur WordNetLemmatizer du paquet NLTK. Le programme lit les fichiers portant l'extension ".en", divise le texte en tokens, les lemmatise, puis enregistre les tokens lemmatisés dans de nouveaux fichiers, en changeant l'extension en "_lemmatised.en".

Pour la section en français, nous avons utilisé FrenchLefffLemmatizer. Le programme lit les fichiers avec l'extension ".fr", divise le texte en tokens, les lemmatise, et écrit ensuite les tokens lemmatisés dans de nouveaux fichiers, en changeant l'extension en "_lemmatised.fr". Si le dossier de sortie n'existe pas, le programme le crée.

Malheureusement, mon ordinateur s'est avéré trop faible pour fonctionner efficacement avec ce programme. Il bégayait souvent et même le terminal s'éteignait de lui-même. J'ai donc décidé d'essayer d'utiliser le programme sur Google Colab, mais malgré mes efforts, je n'ai pas réussi à le faire fonctionner correctement, principalement en raison d'un manque d'instructions suffisantes.

5. Points forts, limitations et difficultés rencontrées

Pour les deux types d'évaluations, en formes fléchies et en lemmes, le moteur de traduction automatique neuronale OpenNMT a démontré plusieurs points forts ainsi que certaines limitations. Voici un résumé pour un corpus en formes fléchies :

- Points forts :
    - Malgré l'absence de GPU sur la machine, le moteur a réussi à effectuer correctement les deux entraînements.

    - Concernant la traduction de textes en domaine et hors-domaine, le moteur a montré une performance rapide.
    - L'utilisation de corpus en domaine a considérablement amélioré la performance du moteur, comme nous pouvons voir dans le Tableau 1.
- Limitations :
    - Malgré l'utilisation de deux grands corpus pour l'entraînement, les scores d'évaluation du moteur peuvent être considérés comme bas.
    - En prenant en compte que les textes de référence ont été traduits par des traducteurs professionnels, il apparaît que les scores obtenus par nos modèles sont inférieurs.
    - Le moteur ne traduit pas de manière pertinente les textes hors-domaine, comme le confirme un score très bas pour le premier modèle entraîné.
    - Bien que l'entraînement ait pu être effectué sur la machine, il a nécessité plusieurs heures pour chaque modèle, ce qui a considérablement réduit l'efficacité du processus.

Après avoir effectué l'évaluation des deux modèles sur le corpus en formes fléchies, il est évident que les scores sont considérablement bas. Néanmoins, il serait possible d'améliorer ces résultats en exploitant des fonctionnalités plus précises du moteur. Dans notre projet, nous nous sommes concentrés sur l'utilisation générale du moteur plutôt que sur des optimisations spécifiques.
Concernant les difficultés lors de l'entraînement, la plus grande était que le modèle a pris beaucoup de temps à s'entraîner. Cela nous a poussés à limiter la taille du modèle utilisé. De plus, malgré plusieurs essais, nous n'avons pas réussi à traiter les données tokenisées. Nous avons entraîné les deux modèles avec les données tokenisées et non tokenisées, et ce que nous avons pu observer, c'était que le score BLEU était plus élevé avec les données non tokenisées. Lors de l'évaluation des modèles utilisant les données pré-tokenisées, le moteur affichait l'erreur : « *It looks like you forgot to detokenize your test data, which may hurt your score* ». Nous n'avons pas réussi à résoudre ce problème.

6. Organisation

Pour bien mener notre projet, nous avons créé un dépôt GitHub, dont la responsable était Maria Hul. Elle a structuré ce dépôt en créant des dossiers organisés selon les différents aspects de notre projet. De plus, elle a développé et mis à disposition sur notre dépôt le code permettant d'extraire les phrases des deux corpus pour les ensembles d'entraînement, de test et de développement.
L'introduction du rapport a été écrite par Maria Hul et la présentation du moteur OpenNMT par Agata Skrzyniarz. Concernant l'évaluation du moteur de traduction neuronale OpenNMT, Maria Hul a pris en charge l'évaluation sur le corpus en formes fléchies, tandis qu'Agata Skrzyniarz s'est occupée de l'évaluation sur un corpus en lemmes. Nous avons rédigé les parties 3 et 4 de notre rapport, dédiées à ces évaluations. Enfin, chacune de nous a exposé les points forts, les limitations et les difficultés rencontrées lors de son évaluation dans la section 5 de notre rapport.