Actes de CORIA-TALN-RJCRI-RECITAL 2025
Marseille, France, 30 juin au 4 juillet 2025

Améliorer la Traduction Neuronale par Exemple avec des Données Monolingues

Maxime Bouthors[1] Josep Crego[1] François Yvon[2]
(1) SYSTRAN by ChapsVision, 5 rue Feydeau, 75002 Paris, France
(2) Sorbonne Université, CNRS, ISIR, 75005 Paris, France
mbouthors@chapsvision.com, jcrego@chapsvision.com, francois.yvon@cnrs.fr

RÉSUMÉ

Les systèmes de traduction neuronale augmentée par des exemples (RANMT) utilisent des corpus bilingues dits mémoires de traduction (TM). Pourtant, dans de nombreux cas, des corpus monolingues du domaine d'intérêt dans la langue cible sont disponibles. Nos travaux s'intéressent à l'exploitation de telles ressources, en recherchant les segments pertinents directement dans la langue cible, conditionnellement à une phrase source en requête. À cet effet, nous proposons d'améliorer les systèmes de recherche cross-lingue, en les entraînant à réaliser des association lexicales. Nos expériences avec deux architectures neuronales montrent l'avantage de notre méthode dans un cas contrôlé, conduisant à des performances de traduction qui peuvent surpasser les méthodes basées sur une mémoire de traduction. Enfin, nous évaluons notre méthode dans une configuration réaliste pour laquelle la quantité de données monolingues excède celle des données parallèles. Cette approche résulte en une nette amélioration des performances par rapport à des modèles de base ainsi que des encodeurs pré-entraînés.

ABSTRACT

Improving Retrieval-Augmented Neural Machine Translation with Monolingual Data.

Conventional retrieval-augmented neural machine translation (RANMT) systems leverage bilingual corpora, e.g., translation memories (TMs). Yet, in many settings, in-domain monolingual target-side corpora are often available. This work explores ways to take advantage of such resources by retrieving relevant segments directly in the target language, based on a source-side query. For this, we design improved cross-lingual retrieval systems, trained with both sentence level and word-level matching objectives. In our experiments with two RANMT architectures, we first demonstrate the benefits of such cross-lingual objectives in a controlled setting, obtaining translation performances that surpass standard TM-based models. We then showcase our method on a real-world set-up, where the target monolingual resources far exceed the amount of parallel data and observe large improvements of our new techniques, which outperform both the baseline setting, and general-purpose cross-lingual retrievers.

MOTS-CLÉS : traduction neuronale, recherche d'information, recherche cross-lingue, traduction à base d'exemples.

KEYWORDS: neural machine translation, information retrieval, cross-lingual retrieval, retrieval augmented translation.

ARTICLE : **Contribution originelle.**

Actes des 32ème Conférence sur le Traitement Automatique des Langues Naturelles (TALN), volume 1 : articles scientifiques originaux, pages 117–133.

1 Introduction

L'utilisation de modèles génératifs augmentés par des exemples (Li *et al.*, 2022) se généralise rapidement. Cela est dû à leur capacité intrinsèque à conditionner la génération à des exemples illustratifs issus d'une mémoire. L'utilisation de segments recherchés automatiquement est un élément clé de la Traduction Assistée par Ordinateur (TAO) (Arthern, 1978; Kay, 1997; Bowker, 2002) : des segments similaires sont récupérés d'une Mémoire de Traduction (TM), fournissant au/à la traductaire des suggestions pertinentes pour produire une nouvelle traduction. De telles techniques ont été transposées en traduction automatique statistique (Koehn & Senellart, 2010), et, plus récemment, dans la traduction neuronal à base d'exemples (RANMT) (Gu *et al.*, 2018), rendant le processus de traduction entièrement automatisé. Des travaux ont continué à utiliser des données parallèles sous forme de TM, en effectuant la recherche dans la langue source, se servant ensuite de la traduction côté cible pour guider la traduction du modèle. L'utilisation d'exemples est un élément important de la traduction par des Grands Modèles de Langue (LLM) dans laquel les exemples de traduction (sources et cibles) sont insérés dans l'amorce (Brown *et al.*, 2020).

Les approches utilisant une TM utilisent des techniques de **fuzzy-matching** (FM) avec BM25 et/ou la distance de Leveshtein (DL) en tant que filtre ou score d'ordonnancement (Bouthors *et al.*, 2024). Les techniques FM sont connues pour leur efficacité calculatoire, et surpassent les systèmes de recherche continus (Xu *et al.*, 2020). En effet, les phrases récupérés via des méthodes lexicales contiennent les termes appropriés. Cependant, le fait de s'appuyer sur le côté source soulève deux problèmes pouvant causer un défaut de pertinence du côté cible des exemples : (a) le désalignement entre les phrases sources et cibles de la TM ; (b) les divergences morphosyntaxiques entre les deux langues (Dorr, 1994).

Ces problèmes peuvent être évacués en effectuant une recherche directement dans **la partie cible** des exemples de la TM, grâce à des méthodes de recherche cross-lingue (CLIR). Un autre avantage important est le fait de pouvoir se dispenser de données parallèles pour la recherche d'exemples. Non seulement les corpus monolingues sont plus faciles à récupérer, mais évitent également le phénomène des traductionais (*translationese*) (Bogoychev & Sennrich, 2020). Pour récupérer de tels exemples, des systèmes de recherche cross-lingue sont disponibles, comme **LASER** (Artetxe & Schwenk, 2019b) ou **LaBSE** (Feng *et al.*, 2022). Cependant, il leur manque la capacité cruciale d'assurer une couverture lexicale forte entre les segments jugés similaires.

Dans cet article, nous proposons une nouvelle méthode pour renforcer la capacité des modèles alingues (*language agnostic*) à trouver des phrases exemples **lexicalement proches d'une potentielle traduction**. Nous explorons plusieurs stratégies d'ajustement d'encodeurs alingues pré-entraînés dans l'optique de reproduire dans un contexte cross-lingue le comportement des systèmes FM.

Nos expériences concernent deux paires de langue (anglais-français et allemand-anglais), et de multiples domaines de taille variable. Nous considérons deux architectures RANMT : `NFA`, un modèle autorégressif dont la source est augmentée d'un exemple (Bulte & Tezcan, 2019) ; $\texttt{TM}^k\texttt{-LevT}$, un modèle d'édition non-autorégressif (Bouthors *et al.*, 2023).

Dans un premier temps, nous appliquons nos méthodes aux modèles `NFA` et $\texttt{TM}^k\texttt{-LevT}$, montrant l'avantage de nos ajustements dans un cas contrôlé sur des données de TM. Dans un second temps, nous exposons notre méthode à un cadre véritablement monolingue dans lequel la quantité de phrases disponibles excède largement la première configuration qui, elle, ne contient que des phrases parallèles. Il convient de noter que notre méthode requiert toujours des phrases parallèles pour l'étape

d'ajustement, mais peut exploiter des données monolingues lors de l'inférence.

2 Travaux Connexes

Nos travaux sont liés aux méthodes dites RAG (Retrieval Augmented Generation), utilisant des méthodes lexicales (BM25 (Robertson & Walker, 1994) et distance de Levenshtein (Levenshtein, 1965)) et sémantiques pour la recherche d'information. Les systèmes sémantiques se basent généralement sur des encodeurs duals entraînés sur des données parallèles (Gillick *et al.*, 2018) avec une fonction de perte contrastive (Sohn, 2016). Notre intérêt se porte sur des encodeurs multilingues (comme LASER (Artetxe & Schwenk, 2019b) et LaBSE (Feng *et al.*, 2022)) qui peuvent permettre de rechercher des traductions mutelles avec des méthodes kNN.

Ces dernières années, des travaux ont cherché à intégrer les exemples dans des systèmes de TA. Cela permet plus de transparence sur les choix de traduction des modèles (Rudin, 2019), et permet de déléguer une partie des connaissances (lexicales, stylistiques, etc.) du modèle à une base de données externe. Une implémentation simple consiste à encoder le côté cible des exemples pour fournir de l'information contextuelle (Bulte & Tezcan, 2019; Gu *et al.*, 2018; Xia *et al.*, 2019; He *et al.*, 2021; Cheng *et al.*, 2022; Agrawal *et al.*, 2023). D'autres travaux utilisent à la fois les côtés source et cible (Pham *et al.*, 2020; Reheman *et al.*, 2023).

Au lieu de regénérer un texte complet, les modèles d'édition proposent d'effectuer des opérations minimales pour transformer les exemples en une traduction. De telles techniques sont étudiées dans (Niwa *et al.*, 2022; Xu *et al.*, 2023; Zheng *et al.*, 2023; Bouthors *et al.*, 2023), adaptant le Transformer de Levenshtein (Gu *et al.*, 2019).

Les système de TA via des LLM utilisent généralement des exemples via l'apprentissage en contexte (ICL), dans lequel l'amorce est enrichie de multiples démonstrations de la tâche de traduction (Radford *et al.*, 2019). De nombreuses études ont cherché à optimiser les performances des telles architectures en modifiant les critères de sélection des exemples (Moslem *et al.*, 2023; Vilar *et al.*, 2023; Zhang *et al.*, 2023; Hendy *et al.*, 2023; Bawden & Yvon, 2023; Zebaze *et al.*, 2024).

Notre alignement lexical correspond à une forme d'apprentissage de métrique neuronal, dans laquelle un espace de grande dimension est réduit à un plus petit (Suárez *et al.*, 2021). Ce champ étudie l'apprentissage de distances dans les données (Kulis, 2013; Bellet *et al.*, 2015), ce qui est utile pour des algorithmes de recherche par similarité (Cakir *et al.*, 2019).

3 Méthode

3.1 Formulation du Problème

Étant donné un corpus monolingue $\mathcal{D} = (\mathbf{y}_1, \ldots, \mathbf{y}_N)$ et une phrase source $\mathbf{x}$, l'objectif des méthodes CLIR pour la TA est de retrouver un sous-ensemble de k segments pertinents dans $\mathcal{D}$: $(\tilde{\mathbf{y}}_1, \ldots, \tilde{\mathbf{y}}_k)$. Notre approche correspond à la recherche des k plus proches voisins de l'encodage de la source $\mathbf{x}$ parmi les exemples de $\mathcal{D}$. Le score de similarité utilisé est la similarité cosinus.

Généralement, les encodeurs alingues sont entraînés avec des méthodes d'apprentissage de représen-

tation ou d'apprentissage contrastif. De telles méthodes encodent principalement des informations sémantiques, de sorte que deux paraphrases avec des termes totalement différents correspondent généralement à des représentations proches. Pourtant, il est désirable que les $\tilde{\mathbf{y}}_i$ ainsi récupérés partagent une similarité lexicale avec ce qui pourrait être une potentielle traduction $\mathbf{y}$ de la source $\mathbf{x}$. Dans les domaines spécialisés (comme le droit ou la médecine), le respect de la terminologie propre au domaine prime sur les critères de similarité sémantique, motivant le recours à une fonction de perte lexicale. Nous proposons plusieurs fonctions de perte basées sur la similarité de Levenshtein DL définie comme :

$$\mathrm{DL}(\mathbf{y}, \tilde{\mathbf{y}}) = 1 - \frac{\Delta(\mathbf{y}, \tilde{\mathbf{y}})}{\max(|\mathbf{y}|, |\tilde{\mathbf{y}}|)}, \tag{1}$$

Elles définissent un objectif précisément construit pour augmenter la quantité de fragments des phrases exemples alignés à la phrase source. L'espace d'encodage est alors transformé à l'apprentissage de sorte que la propriété suivante soit respectée :

$$f(\mathrm{sim}(\mathbf{x}, \tilde{\mathbf{y}})) \approx \mathrm{DL}(\tilde{\mathbf{y}}, \mathbf{y}), \tag{2}$$

où f est une fonction associant le codomaine de la similarité cosinus sim ($[-1,\ 1]$) à celui de DL ($[0,\ 1]$). Autrement dit, le respect de l'équation (2) entend que la similarité cosinus doive correspondre de manière sous-jacente à une similarité de Levenshtein.

3.2 Apprentissage

Nous considérons trois fonctions de perte. Les deux premières consistent à apprendre à respecter l'équation (2) selon la formulation d'un problème de régression. La fonction de perte est de la forme :

$$\mathcal{L}(\mathbf{x}, \mathbf{y}, \tilde{\mathbf{y}}) = \mathrm{Err}(f(\mathrm{sim}(\mathbf{x}, \tilde{\mathbf{y}})), \mathrm{DL}(\tilde{\mathbf{y}}, \mathbf{y})), \tag{3}$$

avec Err la fonction d'erreur (ici MSE ou MAE). Quant à la fonction f, nous choisissons une fonction bijective croissante paramétrisée :

$$f : \begin{array}{l} [-1,\ 1] \to [0,\ 1] \\ t \mapsto \sigma(a \times \mathrm{arctanh}(t) + b), \end{array} \tag{4}$$

avec a (la pente) et b (la position) des paramètres appris ; et σ la fonction sigmoïde.

Le troisième objectif s'inscrit dans le cadre de l'apprentissage d'ordonnancement. Étant donné un ensemble de k segments, ordonnés par similarité de Levenshtein décroissante vis-à-vis de la référence $\mathbf{y}$, la fonction de perte est calculée ainsi :

$$\mathcal{L}(\mathbf{x}, \mathbf{y}, \tilde{\mathbf{y}}_{[1:k]}) = \sum_{i>j} \max(0, \mathrm{sim}_\theta(\mathbf{x}, \tilde{\mathbf{y}}_i) - \mathrm{sim}_\theta(\mathbf{x}, \tilde{\mathbf{y}}_j) + m \times |\,\mathrm{DL}(\mathbf{y}, \tilde{\mathbf{y}}_j) - \mathrm{DL}(\mathbf{y}, \tilde{\mathbf{y}}_j)|), \tag{5}$$

avec m un facteur d'échelle. L'équation (5) correspond à une agrégation de fonctions de pertes à marge, elles-mêmes inspirées du *triplet loss* (Schultz & Joachims, 2003). La marge n'est cependant pas fixe : elle est proportionnelle à la différence absolue entre la similarité de Levenshtein des deux exemples comparés $\tilde{\mathbf{y}}_i$, $\tilde{\mathbf{y}}_j$ avec la référence $\mathbf{y}$.

4 Données et Métriques

Données Nous considérons deux tâches de traduction : anglais vers français (en-fr) et allemand vers anglais (de-en). Les données contiennent une grande variété de domaines textuels (16) tous issus d'OPUS (Tiedemann, 2012). Les détails sont indiqués dans l'annexe A. Nous préparons également un grand corpus Wikipédia français [1]. Le texte est segmenté en pseudo phrases via des expréssions rationnelles simples. Les duplicats et les fortes correspondances FM [2] avec les partition de validation et de test sont retirés du corpus. La taille finale est de 45M de phrases.

Métriques Le succès de systèmes cross-lingues est évalué via la similarité de Levenshtein DL entre la meilleure correspondance et la référence dans la partition de validation, pour la recherche effectuée dans le corpus d'apprentissage. Nous reportons également le taux d'erreur `xsim`, mesurant la capacité à retrouver des phrases parallèles dans des corpus bi-parallèles (Artetxe & Schwenk, 2019a), ici le corpus d'entraînement. La qualité de traduction est évaluée avec `BLEU` (Papineni *et al.*, 2002) selon SacreBLEU (Post, 2018), [3] et `COMET` [4] (Rei *et al.*, 2022a).

5 Configuration Expérimentale

Configuration de Recherche Dans nos expériences contrôlées dans des corpus parallèles, nous pouvons effectuer une comparaison avec les méthodes de recherche FM usuelles. Pour chaque configuration, nous récupérons jusqu'à k=3 exemples. Nous considérons les systèmes de recherche avec les configurations suivantes :

- **`fuzzy-src`** : FM avec DL sans filtre ;
- **`fuzzy-gold`** : FM avec DL sans filtre côté cible, c'est-à-dire en utilisant la référence. Il s'agit d'une configuration oracle, correspondant à une borne supérieure empirique de ce que pourrait faire une fonction de recherche idéale ;
- **`fuzzy-bt`** : les phrases dans la langue cible de la TM sont rétrotraduites dans la langue source. La méthode de recherche appliquée est ensuite identique à **`fuzzy-src`** à la différence que la recherche se fait sur les pseudo-sources. Nous simulons ainsi une alternative à la recherche CLIR dans le cas où un corpus monolingue cible d'exemples est disponible ;
- **`dense`** : un modèle BERT (Devlin *et al.*, 2019) est entraîné depuis zéro avec une fonction de perte contrastive InfoNCE (Sohn, 2016) sur des données parallèles (pour la recherche CLIR) ;
- **`dense+bow`** : nous ajoutons une fonction de perte sac-de-mots au modèle précédant de manière identique aux travaux de Cai *et al.* (2021) afin de favoriser l'encodage d'informations lexicales cross-lingues ;
- **`LASER/LaBSE`** : LASER (Artetxe & Schwenk, 2019c) et LaBSE (Feng *et al.*, 2022) utilisés en tant que modèles pré-entraînés pour la recherche CLIR ;
- **`ft-[modèle]-[Err]`** : nous ajustons l'un des encodeurs précédents avec l'une des trois fonctions de perte {MSE, MAE, Rank} (e.g. **`ft-LaBSE-MSE`**).

Pour chaque méthode, un seuil est appliqué pour filtrer les phrases avec un score de similarité faible. Pour chaque système de recherche, nous ajustons ce seuil pour que la partition de validation obtienne

1. https://huggingface.co/datasets/Plim/fr_wikipedia_processed.
2. Lorsque DL > 0.9.
3. signature : `nrefs:1|case:mixed|eff:no|tok:13a| smooth:exp|version:2.1.0`;
4. `Unbabel/wmt22-comet-da`, avec l'implémentation officielle.

un taux d'exemples [5] de 50%. Ce taux constant permet de s'assurer que chaque résultat est comparable, en retirant le taux d'exemples des variables de confusion : **les résultats reflètent uniquement une différence de qualité des exemples**.

Architectures RANMT Nous considérons deux architectures RANMT. La première est une implémentation locale de `NFA` (Bulte & Tezcan, 2019), qui étend l'encodeur-décodeur standard en concaténant le côté cible des exemples à la source à l'entrée de l'encodeur. Au plus un seul exemple est utilisé, le rendant capable de produire une traduction même lorsqu'aucun exemple n'est présenté. La seconde architecture est une implémentation d'un encodeur-décodeur non-autorégressif d'édition qui peut co-éditer jusqu'à k exemples pour générer une traduction : le Transformer multi-Levenshtein (`TM`^{k}`-LevT`) (Bouthors *et al.*, 2023). Les exemples utilisés à l'entraînement sont également sélectionnés selon la méthode **`fuzzy-src`**. Dans nos expériences, k=3.

6 Résultats

Dans cette section, nous présentons les résultats de différentes fonctions de recherche dans une TM, à savoir les corpus d'apprentissage des données *en-fr* et *de-en*. Nous pouvons ainsi comparer la méthode **`fuzzy-src`** aux méthodes cross-lingues, recherchant directement les segments dans la langue cible.

Scores de Recherche Nous évaluons le taux d'erreur `xsim` (dans la partition d'entraînement) et la similarité de Levenshtein moyenne (dans la partition de validation) du premier exemple récupéré. Nous reportons les résultats dans le tableau 1.

L'utilisation de la rétrotraduction (**`fuzzy-bt`**) dégrade la proximité lexicale des exemples par rapport à la *baseline* **`fuzzy-src`**. L'usage d'un objectif sac-de-mot (**`dense+bow`**) permet d'augmenter les performances de recherche sur les deux métriques. Les ajustements **MSE** et **MAE** appliqués au modèle **`dense+bow`** améliorent les résultats sur les deux métriques. Lorsqu'ils sont appliqués à **`LaBSE`**, ils améliorent les résultats sur les données *en-fr*, mais conduisent à des résultats inchangés ou dégradés sur les données *de-en*. L'ajustement **Rank** obtient des résultats plus nuancés, entraînant une dégradation des performances du modèle **`dense+bow`**, tout en ayant un effet relativement neutre sur les scores de **`LaBSE`**. **`fuzzy-gold`** met en évidence la présence d'une marge de progression additionnelle de la proximité lexicale des exemples par rapport à la configuration **`ft-LaBSE-MAE`**, obtenant les plus hauts scores (hors **`fuzzy-gold`**). la recherche d'exemples côté source (allemand) produit des phrases côté cible (anglais) moins proches lexicalement que n'importe quel système CLIR sur les données *de-en*. Il semble plus difficile de récupérer des exemples de qualité sur les données anglais-allemand que sur les données anglais-français puisque les scores de taux d'erreur `xsim` sont plus importants ($\sim$ 15 points de taux d'erreur `xsim` en plus) et que la DL moyenne est également plus faible (de 2 points pour les méthodes CLIR et de 10 points pour les méthodes FM).

Absence d'exemples : nous utilisons les modèles `NLLB` et `NFA` sans le moindre exemple en tant que *baselines*. Leur scores `BLEU`, moyennés sur tous les domaines, sont respectivement de 39,0 (en-fr) et 33,8 (de-en) pour `NLLB` et 48,0 (en-fr) et 38,2 (de-en) pour `NFA`. Les détails complets par domaine sont présentés dans le tableau 3.

5. Proportion de phrases avec au moins un exemple récupéré.

					TM^3-LevT				NFA			
	test en-fr		test de-en		test en-fr		test de-en		test en-fr		test de-en	
	DL moy. ↑	xsim ↓	DL moy. ↑	xsim ↓	BLEU	COMET	BLEU	COMET	BLEU	COMET	BLEU	COMET
					-	-	-	-	48.0	87.1	38.2	79.8
fuzzy-gold (oracle)	50,0	-	53,4	-	45,3	51,7	31,3	-8,6	52,2	87,8	44,7	82,2
fuzzy-src	36,0	-	21,9	-	**43,8**	**48,4**	28,8	-10,3	**51,3**	**87,5**	41,9	**81,7**
fuzzy-bt	31,1	-	21,8	-	40,5	43,8	22,0	-22,9	49,9	87,2	38,9	81,3
dense	30,4	10,8	-	-	41,5	42,2	-	-	49,9	87,1	-	-
dense+bow	32,7	8,0	30,9	25,2	42,4	44,1	28,3	-11,6	50,5	87,3	40,0	80,9
ft-dense+bow-MSE	34,9	7,6	33,6	25,2	43,1	46,0	29,0	-12,7	51,0	**87,5**	41,6	81,5
ft-dense+bow-MAE	35,2	8,0	32,9	25,3	43,2	46,3	29,1	-12,6	51,0	87,4	41,8	81,4
ft-dense+bow-Rank	29,9	11,9	25,9	41,5	42,1	44,3	26,0	-17,9	50,7	87,3	41,8	80,9
LASER	32,8	10,0	33,2	26,2	42,5	44,3	**30,4**	**-7,7**	49,9	87,3	41,1	81,3
LaBSE	33,8	7,4	34,4	**21,0**	43,1	45,5	29,8	-9,5	50,5	87,3	41,5	81,4
ft-LaBSE-MSE	36,7	7,6	34,0	24,6	**43,7**	**47,8**	27,7	-12,5	**51,2**	**87,5**	41,6	**81,6**
ft-LaBSE-MAE	**37,0**	**6,4**	**34,5**	23,5	43,6	47,7	29,1	-9,2	**51,2**	**87,5**	**42,0**	81,5
ft-LaBSE-Rank	36,0	7,3	34,2	23,3	43,3	47,1	28,4	-12,2	51,1	**87,5**	41,8	**81,6**

TABLE 1 – Taux d'erreur xsim et similarité de Levenshtein (DL) moyenne pour les différents systèmes de recherche (gauche) et scores BLEU et COMET des modèles TM^3-LevT et NFA (droite).

TM^3-LevT : Les performances du modèle TM^3-LevT sont reportées dans le tableau 1. L'écart entre l'oracle **fuzzy-gold** et **fuzzy-src** met en évidence deux phénomènes : (a) Les divergences morpho-syntaxiques entre la langue source et la langue cible. Plus cette divergence est élevée, plus l'écart l'est également ; Aussi, (b) la qualité de l'alignement des phrases du corpus parallèle. On peut précisément observer que l'écart est plus faible sur le corpus *en-fr* (1,7 BLEU) que nous avons nettoyé, comparé à celui de *de-en* (2,5 BLEU). La rétrotraduction (**fuzzy-bt**) produit une dégradation sur toutes les métriques, davantage que ce que le tableau 1 laissait présager. Pour les deux paires de langue, l'utilisation de systèmes CLIR est plus efficace que **fuzzy-bt**. Les meilleurs systèmes obtiennent des résultats similaires à **fuzzy-src**. Concernant *en-fr*, les meilleures performances CLIR sont obtenues avec **LaBSE** ajusté avec MSE ou MAE. De manière similaire à ce que le tableau 1 permettait de projeter, nous n'obtenons pas de tel gain avec nos stratégies d'ajustement sur les données *de-en*. Néanmoins, l'encodeur **LASER** permet à TM^3-LevT d'obtenir une augmentation de 1,6 BLEU par rapport à **fuzzy-src**. Notons que le modèle TM^3-LevT anglais-allemand a été entraîné sur moins de données et que le corpus d'entraînement n'a pas été nettoyé. Par conséquent, le comportement de TM^3-LevT est plus instable que ne peut l'être NFA sur *de-en*.

NFA : Les résultats pour NFA sont présentés dans le tableau 1. Ce modèle obtient des performances nettement supérieures à TM^3-LevT. La marge de progression (jusqu'à **fuzzy-gold**) est plus faible. Pourtant, nous pouvons remarquer que le fait de récupérer les exemples directement côté cible améliore la *baseline* sans exemple d'environ 3 points BLEU dans le meilleur des cas. Les techniques d'ajustement avec MSE et MAE permettent de rattraper complètement tous les scores de **fuzzy-src**. En comparaison, **fuzzy-bt** permet d'augmenter les scores de la version sans exemple, mais obtient les pires scores de traduction avec exemple.

Analyse par domaine : Les scores BLEU par domaine sont reportés dans les tableaux 2 (pour TM^3-LevT) et 3 (pour NFA). D'une part, nous observons la difficulté d'améliorer certains domaines (à savoir News-Commentary et TED2013), mis en évidence par l'absence d'écart entre **fuzzy-gold** et

	English-Français											Allemand-English				
	ECB	EME	Epp	GNO	JRC	KDE	News	PHP	TED	Ubu	Wiki	KDE	Kor	JRC	EME	Sub
fuzzy-gold	58,5	64,8	32,5	61,3	60,5	44,3	27,0	41,0	31,2	40,5	36,9	32,1	8,2	48,7	47,6	19,8
fuzzy-src	56,7	**62,2**	**31,9**	**58,9**	58,8	41,5	**27,2**	**40,3**	**30,7**	**38,9**	34,6	29,2	8,7	45,8	43,3	16,9
fuzzy-bt	51,0	54,7	**31,9**	51,3	55,1	35,2	27,0	38,7	**30,7**	36,2	33,3	21,3	4,5	27,5	39,2	**17,5**
LASER	56,5	60,5	31,7	56,0	58,6	36,6	26,9	39,7	**30,6**	36,4	33,9	**29,3**	11,3	48,9	**46,6**	16,2
LaBSE	**57,0**	59,8	31,3	57,8	**59,4**	40,8	26,9	39,8	30,5	37,0	33,9	29,0	9,3	**49,2**	45,4	16,2
ft-LaBSE-MSE	**57,0**	**62,0**	**31,8**	58,6	58,6	**42,5**	**27,0**	39,9	30,5	**38,1**	**34,9**	28,1	10,9	41,4	41,1	17,0
ft-LaBSE-MAE	56,9	61,4	31,6	**58,7**	58,7	**42,5**	**27,0**	39,9	30,4	37,7	34,5	28,7	**11,7**	44,4	43,7	17,0
ft-LaBSE-Rank	56,6	61,1	31,7	57,7	58,0	41,7	26,9	**40,1**	30,6	37,6	34,1	28,7	9,6	44,6	42,5	16,6

TABLE 2 – Scores BLEU moyen par domaine de TM^3−LevT. Les domaines contaminés de NLLB (rétrotraduction) sont soulignées.

	Anglais-Français											Allemand-Anglais				
	ECB	EME	Epp	GNO	JRC	KDE	News	PHP	TED	Ubu	Wiki	KDE	Kor	JRC	EME	Sub
taux de contamin.	33,9	58,3	8,9	0	2,1	6,8	100	6,8	29,8	0	0	47,5	0	54,0	17,3	8,9
NLLB	44,5	40,3	35,9	39,6	50,7	34,2	32,1	39,0	41,0	35,9	35,4	29,1	23,5	46,4	38,8	31,4
no example	58,9	55,9	39,9	48,8	62,3	42,8	50,1	45,6	43,9	43,5	36,4	39,5	13,2	55,5	51,6	31,0
fuzzy-gold	65,2	65,1	40,3	59,7	68,1	46,6	50,1	48,1	44,1	47,8	39,0	45,6	21,9	63,9	60,5	31,7
fuzzy-src	**64,5**	**63,0**	**40,0**	58,4	**66,6**	45,2	50,0	47,3	44,0	**46,8**	**38,3**	**44,3**	14,2	**62,3**	57,4	31,1
fuzzy-bt	62,4	60,4	39,8	54,7	64,9	44,5	**50,1**	46,8	43,7	44,7	37,0	40,1	13,2	55,6	54,8	31,1
LASER	63,7	61,6	39,8	56,8	65,4	44,0	50,0	46,8	**44,1**	46,1	37,1	41,4	16,1	60,7	56,0	31,1
LaBSE	63,8	62,5	39,0	57,7	65,0	45,3	50,0	46,9	**44,1**	46,0	37,7	43,0	15,6	60,6	56,9	31,1
ft-LaBSE-MSE	**64,3**	**63,0**	**40,0**	58,2	**66,6**	45,3	**50,1**	**47,5**	44,0	**46,8**	37,9	43,1	16,8	60,3	56,9	**31,2**
ft-LaBSE-MAE	64,2	**63,0**	39,9	**58,5**	**66,6**	**45,5**	**50,1**	47,4	43,9	46,3	37,8	43,3	**17,1**	**61,4**	**57,5**	30,9
ft-LaBSE-Rank	64,1	62,6	39,8	58,1	66,3	45,3	50,0	**47,5**	44,0	46,0	**38,1**	**43,7**	15,9	61,0	57,2	**31,2**

TABLE 3 – Scores BLEU par domaine de NFA (et NLLB). Les domaines contaminés par NLLB (pour la rétrotraduction) sont soulignés. Les taux de contaminations du modèle NFA sont également reportés.

la *baseline* sans exemples. D'autre part, certains domaines sont fortement améliorés par nos méthodes (CLIR et ajustements). Pour TM^3−LevT, il s'agit des domaines ECB, EMEA, Gnome, KDE, Ubuntu et Wikipedia. Quant à NFA, il s'agit de Europarl, Gnome, Law et Koran. Nous pouvons caractériser la difficulté de retrouver des exemples dans un domaine par la taille du domaine, puisqu'un grand corpus offre une plus grande opportunité de trouver des phrases similaires, mais aussi par la redondance lexicale et phraséologique. Dans la majorité des domaines, NFA obtient, grâce aux méthodes CLIR, des scores proches de l'oracle **fuzzy-gold**, démontrant l'efficacité de notre méthode.

Comparaison des fonctions de perte : La fonction de perte infoNCE, utilisée pour entraîner **LaBSE** et **dense**, a pour objectif d'identifier des segments parallèles au sein de collections multilingues. De tels systèmes modélisent les phrases dans un espace vectoriel avec une topologie rapprochant les phrases similaires. Cette similarité ne reflète pas uniquement des caractéristiques lexicales et n'a aucune raison *a priori* de former une topologie respectant une quelconque notion de distance/similarité consistante et explicable. L'objectif de **dense+bow** avec une fonction de perte sac-de-mots consiste à enrichir l'encodage avec des informations lexicales cross-lingues : il doit être possible de retrouver le vocabulaire d'une phrase dans la langue L2 à partir de l'encodage dans la langue L1. Une telle méthode permet effectivement une augmentation de la similarité lexicale (tableau 1) et des scores BLEU (tableau 1). Enfin, nous proposons trois fonctions de perte pour précisément transformer la topologie de la représentation des phrases afin de respecter une notion de similarité. Parmi nos trois stratégies d'ajustement, **MSE** et **MAE** (éq. 3) permettent d'obtenir les plus grands gains BLEU.

Effets de la contamination des données : Nous avons identifié deux sources de contamination de données. Premièrement, les données d'apprentissage de NFA contiennent une portion des données de test variable selon les domaines (voir annexe C). Intuitivement, une telle contamination devrait rendre le modèle moins sensible à la sélection d'exemples pertinents. Les résultats par domaine du tableau 3 montrent néanmoins une marge de progression entre les résultats sans exemples et l'oracle. L'écart BLEU étant large pour la plupart des domaines (à l'exception de News-Commentary, Europarl et Subtitles), nous pouvons voir distinctement l'avantage de l'utilisation de méthodes CLIR. Deuxièmement, les données d'entraînement de NLLB contiennent certains domaines, ce qui a pour potentiel effet de surévaluer **fuzzy-bt** dans les domaines contaminés. Dans la totalité des cas, **fuzzy-bt** reste en deçà des résultats des autres méthodes.

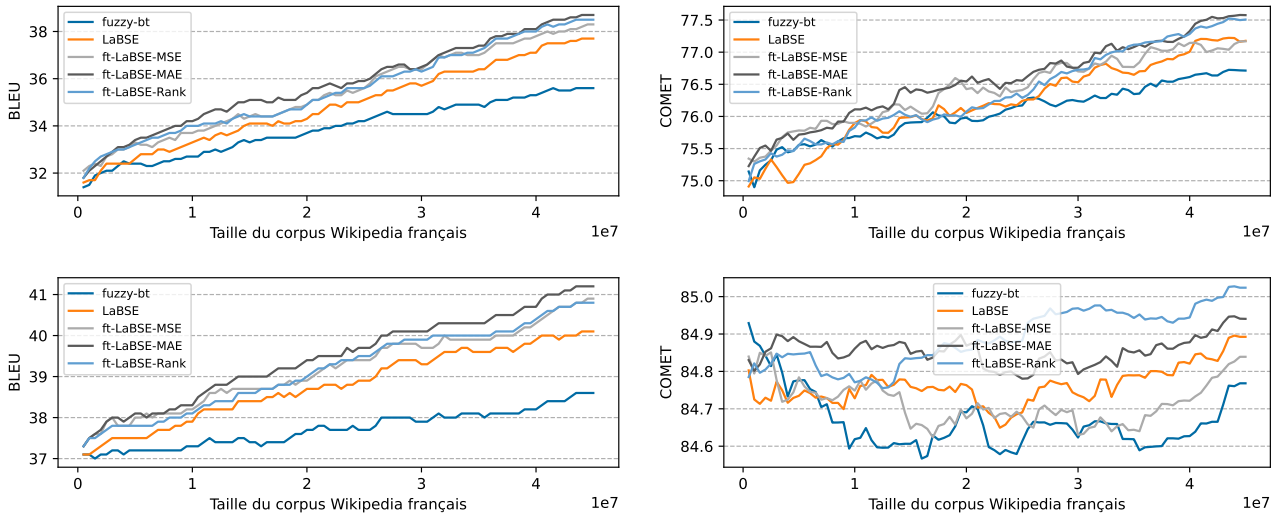

FIGURE 1 – Évolution des scores BLEU et COMET de TM^3-LevT (en haut) et NFA (en bas) en augmentant la taille du corpus de recherche.

6.1 Expérience à Large Échelle

Recherche dans un corpus monolingue : Nous utilisons les mêmes systèmes pour effectuer la recherche d'exemples dans le corpus Wikipedia, contenant 45 millions de segments. Les seuils de filtrage utilisés pour les modèles sont les mêmes, c'est-à-dire choisis pour obtenir un taux d'exemple récupérés dans la TM de 50% sur la partition de validation. Notons que nous excluons **LASER**, qui entraîne des coûts d'indexation très élevés, ainsi que nos modèles **dense** (ajustés ou non) qui obtenaient des résultats inférieurs dans les expériences contrôlées. Wikipedia est un domaine qui présente une grande variété de thèmes avec un haut niveau de formalité. Il est également exempt de toute contamination avec les modèles NFA et NLLB. Le changement du corpus depuis lequel les exemples sont recherchés (passant de 200k segments à 45M) permet une augmentation générale des scores BLEU et COMET (voir tableau 4). L'ajustement avec **MSE** ou **MAE** permet d'obtenir des scores BLEU bien supérieurs à ceux associés à la recherche dans la TM : +3,9 points BLEU pour **ft-LaBSE-MAE** avec TM^3-LevT ; +3,4 pour **ft-LaBSE-MAE** avec NFA.

Influence de la taille du corpus monolingue : Pour analyser l'effet de la taille du corpus monolingue au sein duquel les exemples sont récupérés, nous augmentons progressivement le nombre

	TM³-LevT		NFA	
	BLEU	COMET	BLEU	COMET
sans exemple	-	-	36,4	84,8
fuzzy-bt	35,6	76,7	38,6	84,8
LaBSE	37,7	77,2	40,1	**85,0**
ft-LaBSE-MSE	38,3	77,2	40,9	84,8
ft-LaBSE-MAE	**38,7**	**77,6**	**41,2**	84,9
ft-LaBSE-Rank	38,5	77,5	40,8	**85,0**

TABLE 4 – Scores BLEU et COMET avec les systèmes TM³-LevT et NFA dont les exemples sont recherchés sur le corpus Wikipedia.

	TM		mono	
	TE ↑	DL ↑	TE ↑	DL ↑
fuzzy-bt	39,9	24,0	60,2	31,0
LaBSE	30,9	25,6	59,6	35,2
ft-LaBSE-MSE	33,8	27,2	77,7	37,7
ft-LaBSE-MAE	31,3	27,5	59,0	38,1
ft-LaBSE-Rank	25,7	26,7	61,2	36,7

TABLE 5 – Comparaison du taux d'exemple (TE) et de la similarité de Levenshtein moyenne (DL) sur la partition de test Wikipedia, selon le système et le corpus de recherche (TM ou corpus monolingue).

de phrases constituant l'ensemble de recherche avec un pas de 500k phrases jusqu'à 45M. Nous calculons les scores de traduction pour différents modèles (**LaBSE** et variantes ainsi que **fuzzy-bt**) dans les graphiques de la figure 1. Les modèles bénéficient clairement de l'augmentation de la taille du corpus de recherche.

Nous observons également deux scores additionnels dans le tableau 5 : le taux d'exemple (pourcentage de phrases sources pour lesquelles au moins un exemple est récupéré) et la similarité de Levenshtein DL moyenne (entre le meilleur exemple non filtré et la référence). Nous observons une augmentation du taux d'exemple de 30% en moyenne en récupérant les exemples dans le corpus monolingue. Quant au score DL moyen, le meilleur exemple est supérieur, en moyenne, de 10 points.

7 Conclusion et Perspectives

Nous avons exploré diverses approches pour utiliser des exemples issus de corpus monolingues dans des systèmes de TA à base d'exemples. Notre conclusion principale, basée sur deux paires de langue et 16 domaines, est que la recherche cross-lingue peut être tout aussi efficace que les méthodes classiques de *fuzzy-matching* lexical grâce à notre fonction de perte lexicale. Ces techniques permettent d'obtenir une augmentation significative du score BLEU par rapport à notre point de référence sur le corpus Wikipédia (jusqu'à +3,8 points), montrant ainsi l'avantage de notre méthode à large échelle. Notre implémentation CLIR correspond à un coût calculatoire équivalent au FM.

Nous identifions plusieurs façon d'enrichir nos résultats : (a) en ajustant les modèles sur de plus grands corpus multilingues selon des configurations comparables à **LASER** ou **LaBSE** ; (b) utiliser d'autres encodeurs plus récents comme SONAR (Duquenne *et al.*, 2023) ; (c) utiliser des méthodes CLIR pour entraîner nos modèles plutôt que de se contenter de la recherche FM ; (d) revisiter d'autres aspects comme la relaxation du seuil pour la sélection des exemples. Une autre extension de nos travaux consisterait à adopter les méthodes CLIR dans l'apprentissage en contexte, qui, pour le moment, nécessitent le côté source des exemples (Moslem *et al.*, 2023). Nous pouvons alors ajuster de tels modèles pour n'avoir qu'à utiliser le côté cible.

8 Remerciements

Ce projet a été partiellement financé par l'ANR dans le cadre du projet TraLaLam (ANR-23-IAS1-0006). Il a également bénéficié des resources HPC/AI de GENCI-IDRIS (2022-AD011013583 et 2023-AD010614012). Nous remercions également Dakun Zhang pour son aide précieuse, et les relecteurs et relectrices pour leur retours.

Références

AGRAWAL S., ZHOU C., LEWIS M., ZETTLEMOYER L. & GHAZVININEJAD M. (2023). In-context examples selection for machine translation. In A. ROGERS, J. BOYD-GRABER & N. OKAZAKI, Éds., *Findings of the Association for Computational Linguistics : ACL 2023*, p. 8857–8873, Toronto, Canada : Association for Computational Linguistics. DOI : 10.18653/v1/2023.findings-acl.564.

AHARONI R. & GOLDBERG Y. (2020). Unsupervised domain clusters in pretrained language models. In D. JURAFSKY, J. CHAI, N. SCHLUTER & J. TETREAULT, Éds., *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, p. 7747–7763, Online : Association for Computational Linguistics. DOI : 10.18653/v1/2020.acl-main.692.

ARTETXE M. & SCHWENK H. (2019a). Margin-based parallel corpus mining with multilingual sentence embeddings. In A. KORHONEN, D. TRAUM & L. MÀRQUEZ, Éds., *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, p. 3197–3203, Florence, Italy : Association for Computational Linguistics. DOI : 10.18653/v1/P19-1309.

ARTETXE M. & SCHWENK H. (2019b). Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics*, **7**, 597–610. DOI : 10.1162/tacl_a_00288.

ARTETXE M. & SCHWENK H. (2019c). Massively Multilingual Sentence Embeddings for Zero-Shot Cross-Lingual Transfer and Beyond. *Transactions of the Association for Computational Linguistics*, **7**, 597–610. Place : Cambridge, MA Publisher : MIT Press, DOI : 10.1162/tacl_a_00288.

ARTHERN P. J. (1978). Machine translation and computerised terminology systems - a translator's viewpoint. In B. M. SNELL, Éd., *Translating and the Computer*, London, UK : Aslib Proceedings.

BAWDEN R. & YVON F. (2023). Investigating the translation performance of a large multilingual language model : the case of BLOOM. In M. NURMINEN, J. BRENNER, M. KOPONEN, S. LATOMAA, M. MIKHAILOV, F. SCHIERL, T. RANASINGHE, E. VANMASSENHOVE, S. A. VIDAL, N. ARANBERRI, M. NUNZIATINI, C. P. ESCARTÍN, M. FORCADA, M. POPOVIC, C. SCARTON & H. MONIZ, Éds., *Proceedings of the 24th Annual Conference of the European Association for Machine Translation*, p. 157–170, Tampere, Finland : European Association for Machine Translation.

BELLET A., HABRARD A. & SEBBAN M. (2015). *Metric learning*. Morgan & Claypool Publishers.

BOGOYCHEV N. & SENNRICH R. (2020). Domain, translationese and noise in synthetic data for neural machine translation. *CoRR*, **abs/1911.03362**.

BOUTHORS M., CREGO J. & YVON F. (2023). Towards example-based NMT with multi-Levenshtein transformers. In H. BOUAMOR, J. PINO & K. BALI, Éds., *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, p. 1830–1846, Singapore : Association for Computational Linguistics. DOI : 10.18653/v1/2023.emnlp-main.113.

BOUTHORS M., CREGO J. & YVON F. (2024). Retrieving examples from memory for retrieval augmented neural machine translation : A systematic comparison. In K. DUH, H. GO-

MEZ & S. BETHARD, Éds., *Findings of the Association for Computational Linguistics : NAACL 2024*, p. 3022–3039, Mexico City, Mexico : Association for Computational Linguistics. DOI : 10.18653/v1/2024.findings-naacl.190.

BOWKER L. (2002). *Computer-aided translation technology : A practical introduction*. University of Ottawa Press.

BROWN T., MANN B., RYDER N., SUBBIAH M., KAPLAN J. D., DHARIWAL P., NEELAKANTAN A., SHYAM P., SASTRY G., ASKELL A., AGARWAL S., HERBERT-VOSS A., KRUEGER G., HENIGHAN T., CHILD R., RAMESH A., ZIEGLER D., WU J., WINTER C., HESSE C., CHEN M., SIGLER E., LITWIN M., GRAY S., CHESS B., CLARK J., BERNER C., MCCANDLISH S., RADFORD A., SUTSKEVER I. & AMODEI D. (2020). Language models are few-shot learners. In H. LAROCHELLE, M. RANZATO, R. HADSELL, M. BALCAN & H. LIN, Éds., *Advances in Neural Information Processing Systems*, volume 33, p. 1877–1901 : Curran Associates, Inc.

BULTE B. & TEZCAN A. (2019). Neural fuzzy repair : Integrating fuzzy matches into neural machine translation. In A. KORHONEN, D. TRAUM & L. MÀRQUEZ, Éds., *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, p. 1800–1809, Florence, Italy : Association for Computational Linguistics. DOI : 10.18653/v1/P19-1175.

CAI D., WANG Y., LI H., LAM W. & LIU L. (2021). Neural machine translation with monolingual translation memory. In C. ZONG, F. XIA, W. LI & R. NAVIGLI, Éds., *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1 : Long Papers)*, p. 7307–7318, Online : Association for Computational Linguistics. DOI : 10.18653/v1/2021.acl-long.567.

CAKIR F., HE K., XIA X., KULIS B. & SCLAROFF S. (2019). Deep metric learning to rank. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

CHENG X., GAO S., LIU L., ZHAO D. & YAN R. (2022). Neural machine translation with contrastive translation memories. In Y. GOLDBERG, Z. KOZAREVA & Y. ZHANG, Éds., *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, p. 3591–3601, Abu Dhabi, United Arab Emirates : Association for Computational Linguistics. DOI : 10.18653/v1/2022.emnlp-main.235.

COSTA-JUSSÀ M. R., CROSS J., ÇELEBI O., ELBAYAD M., HEAFIELD K., HEFFERNAN K., KALBASSI E., LAM J., LICHT D., MAILLARD J., SUN A., WANG S., WENZEK G., YOUNGBLOOD A., AKULA B., BARRAULT L., GONZALEZ G. M., HANSANTI P., HOFFMAN J., JARRETT S., SADAGOPAN K. R., ROWE D., SPRUIT S., TRAN C., ANDREWS P., AYAN N. F., BHOSALE S., EDUNOV S., FAN A., GAO C., GOSWAMI V., GUZMÁN F., KOEHN P., MOURACHKO A., ROPERS C., SALEEM S., SCHWENK H., WANG J. & TEAM N. (2024). Scaling neural machine translation to 200 languages. *Nature*, **630**(8018), 841–846. ISBN : 1476-4687 tex.date-added : 2024-08-21 15 :32 :57 +0200 tex.date-modified : 2024-08-21 15 :32 :57 +0200, DOI : 10.1038/s41586-024-07335-x.

DEVLIN J., CHANG M.-W., LEE K. & TOUTANOVA K. (2019). BERT : Pre-training of deep bidirectional transformers for language understanding. In J. BURSTEIN, C. DORAN & T. SOLORIO, Éds., *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long and Short Papers)*, p. 4171–4186, Minneapolis, Minnesota : Association for Computational Linguistics. DOI : 10.18653/v1/N19-1423.

DORR B. J. (1994). Machine translation divergences : A formal description and proposed solution. *Computational Linguistics*, **20**(4), 597–633.

DOUZE M., GUZHVA A., DENG C., JOHNSON J., SZILVASY G., MAZARÉ P.-E., LOMELI M., HOSSEINI L. & JÉGOU H. (2024). The Faiss library. *CoRR*, **abs/2401.08281**.

DUQUENNE P.-A., SCHWENK H. & SAGOT B. (2023). Sonar : Sentence-level multimodal and language-agnostic representations. *CoRR*, **abs/2308.11466**.

FENG F., YANG Y., CER D., ARIVAZHAGAN N. & WANG W. (2022). Language-agnostic BERT sentence embedding. In S. MURESAN, P. NAKOV & A. VILLAVICENCIO, Éds., *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, p. 878–891, Dublin, Ireland : Association for Computational Linguistics. DOI : 10.18653/v1/2022.acl-long.62.

GILLICK D., PRESTA A. & TOMAR G. S. (2018). End-to-end retrieval in continuous space. *CoRR*, **abs/1811.08008**.

GU J., WANG C. & ZHAO J. (2019). Levenshtein transformer. In H. WALLACH, H. LAROCHELLE, A. BEYGELZIMER, F. D'ALCHÉ-BUC, E. FOX & R. GARNETT, Éds., *Advances in Neural Information Processing Systems*, volume 32 : Curran Associates, Inc.

GU J., WANG Y., CHO K. & LI V. O. (2018). Search Engine Guided Neural Machine Translation. *Proceedings of the AAAI Conference on Artificial Intelligence*, **32**(1). DOI : 10.1609/aaai.v32i1.12013.

HE Q., HUANG G., CUI Q., LI L. & LIU L. (2021). Fast and accurate neural machine translation with translation memory. In C. ZONG, F. XIA, W. LI & R. NAVIGLI, Éds., *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1 : Long Papers)*, p. 3170–3180, Online : Association for Computational Linguistics. DOI : 10.18653/v1/2021.acl-long.246.

HENDY A., ABDELREHIM M., SHARAF A., RAUNAK V., GABR M., MATSUSHITA H., KIM Y. J., AFIFY M. & AWADALLA H. H. (2023). How good are GPT models at machine translation ? a comprehensive evaluation. *CoRR*, **abs/2302.09210**. DOI : 10.48550/ARXIV.2302.09210.

KAY M. (1997). The proper place of men and machines in language translation. *Machine Translation*, **12**(1/2), 3–23.

KOEHN P. & SENELLART J. (2010). Convergence of translation memory and statistical machine translation. In V. ZHECHEV, Éd., *Proceedings of the Second Joint EM+/CNGL Workshop : Bringing MT to the User : Research on Integrating MT in the Translation Industry*, p. 21–32, Denver, Colorado, USA : Association for Machine Translation in the Americas.

KULIS B. (2013). Metric learning : A survey. *Foundations and Trends® in Machine Learning*, **5**(4), 287–364. DOI : 10.1561/2200000019.

LEVENSHTEIN V. I. (1965). Binary codes capable of correcting deletions, insertions, and reversals. *Soviet physics. Doklady*, **10**, 707–710.

LI H., SU Y., CAI D., WANG Y. & LIU L. (2022). A survey on retrieval-augmented text generation. *CoRR*, **abs/2202.01110**.

MOSLEM Y., HAQUE R., KELLEHER J. D. & WAY A. (2023). Adaptive machine translation with large language models. In M. NURMINEN, J. BRENNER, M. KOPONEN, S. LATOMAA, M. MIKHAILOV, F. SCHIERL, T. RANASINGHE, E. VANMASSENHOVE, S. A. VIDAL, N. ARANBERRI, M. NUNZIATINI, C. P. ESCARTÍN, M. FORCADA, M. POPOVIC, C. SCARTON & H. MONIZ, Éds., *Proceedings of the 24th Annual Conference of the European Association for Machine Translation*, p. 227–237, Tampere, Finland : European Association for Machine Translation.

NIWA A., TAKASE S. & OKAZAKI N. (2022). Nearest neighbor non-autoregressive text generation. *CoRR*, **abs/2208.12496**. DOI : 10.48550/ARXIV.2208.12496.

PAPINENI K., ROUKOS S., WARD T. & ZHU W.-J. (2002). Bleu : a method for automatic evaluation of machine translation. In P. ISABELLE, E. CHARNIAK & D. LIN, Éds., *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, p. 311–318, Philadelphia, Pennsylvania, USA : Association for Computational Linguistics. DOI : 10.3115/1073083.1073135.

PHAM M. Q., XU J., CREGO J., YVON F. & SENELLART J. (2020). Priming neural machine translation. In L. BARRAULT, O. BOJAR, F. BOUGARES, R. CHATTERJEE, M. R. COSTA-JUSSÀ, C. FEDERMANN, M. FISHEL, A. FRASER, Y. GRAHAM, P. GUZMAN, B. HADDOW, M. HUCK, A. J. YEPES, P. KOEHN, A. MARTINS, M. MORISHITA, C. MONZ, M. NAGATA, T. NAKAZAWA & M. NEGRI, Éds., *Proceedings of the Fifth Conference on Machine Translation*, p. 516–527, Online : Association for Computational Linguistics.

POST M. (2018). A call for clarity in reporting BLEU scores. In O. BOJAR, R. CHATTERJEE, C. FEDERMANN, M. FISHEL, Y. GRAHAM, B. HADDOW, M. HUCK, A. J. YEPES, P. KOEHN, C. MONZ, M. NEGRI, A. NÉVÉOL, M. NEVES, M. POST, L. SPECIA, M. TURCHI & K. VERSPOOR, Éds., *Proceedings of the Third Conference on Machine Translation : Research Papers*, p. 186–191, Brussels, Belgium : Association for Computational Linguistics. DOI : 10.18653/v1/W18-6319.

RADFORD A., WU J., CHILD R., LUAN D., AMODEI D., SUTSKEVER I. *et al.* (2019). Language models are unsupervised multitask learners. *OpenAI blog*, **1**(8), 9.

REHEMAN A., ZHOU T., LUO Y., YANG D., XIAO T. & ZHU J. (2023). Prompting neural machine translation with translation memories. *Proceedings of the AAAI Conference on Artificial Intelligence*, **37**(11), 13519–13527. DOI : 10.1609/aaai.v37i11.26585.

REI R., C. DE SOUZA J. G., ALVES D., ZERVA C., FARINHA A. C., GLUSHKOVA T., LAVIE A., COHEUR L. & MARTINS A. F. T. (2022a). COMET-22 : Unbabel-IST 2022 submission for the metrics shared task. In P. KOEHN, L. BARRAULT, O. BOJAR, F. BOUGARES, R. CHATTERJEE, M. R. COSTA-JUSSÀ, C. FEDERMANN, M. FISHEL, A. FRASER, M. FREITAG, Y. GRAHAM, R. GRUNDKIEWICZ, P. GUZMAN, B. HADDOW, M. HUCK, A. JIMENO YEPES, T. KOCMI, A. MARTINS, M. MORISHITA, C. MONZ, M. NAGATA, T. NAKAZAWA, M. NEGRI, A. NÉVÉOL, M. NEVES, M. POPEL, M. TURCHI & M. ZAMPIERI, Éds., *Proceedings of the Seventh Conference on Machine Translation (WMT)*, p. 578–585, Abu Dhabi, United Arab Emirates (Hybrid) : Association for Computational Linguistics.

REI R., TREVISO M., GUERREIRO N. M., ZERVA C., FARINHA A. C., MAROTI C., C. DE SOUZA J. G., GLUSHKOVA T., ALVES D., COHEUR L., LAVIE A. & MARTINS A. F. T. (2022b). CometKiwi : IST-unbabel 2022 submission for the quality estimation shared task. In P. KOEHN, L. BARRAULT, O. BOJAR, F. BOUGARES, R. CHATTERJEE, M. R. COSTA-JUSSÀ, C. FEDERMANN, M. FISHEL, A. FRASER, M. FREITAG, Y. GRAHAM, R. GRUNDKIEWICZ, P. GUZMAN, B. HADDOW, M. HUCK, A. JIMENO YEPES, T. KOCMI, A. MARTINS, M. MORISHITA, C. MONZ, M. NAGATA, T. NAKAZAWA, M. NEGRI, A. NÉVÉOL, M. NEVES, M. POPEL, M. TURCHI & M. ZAMPIERI, Éds., *Proceedings of the Seventh Conference on Machine Translation (WMT)*, p. 634–645, Abu Dhabi, United Arab Emirates (Hybrid) : Association for Computational Linguistics.

ROBERTSON S. & WALKER S. (1994). Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *Some Simple Effective Approximations to the 2-Poisson Model for Probabilistic Weighted Retrieval*, p. 232–241. DOI : 10.1007/978-1-4471-2099-5_24.

RUDIN C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, **1**(5), 206–215. DOI : 10.1038/s42256-019-0048-x.

SCHULTZ M. & JOACHIMS T. (2003). Learning a distance metric from relative comparisons. In S. THRUN, L. SAUL & B. SCHÖLKOPF, Éds., *Advances in Neural Information Processing Systems*,

volume 16 : MIT Press.

SOHN K. (2016). Improved deep metric learning with multi-class n-pair loss objective. In D. LEE, M. SUGIYAMA, U. LUXBURG, I. GUYON & R. GARNETT, Éds., *Advances in Neural Information Processing Systems*, volume 29 : Curran Associates, Inc.

SUÁREZ J. L., GARCÍA S. & HERRERA F. (2021). A tutorial on distance metric learning : Mathematical foundations, algorithms, experimental analysis, prospects and challenges. *Neurocomputing*, **425**, 300–322. DOI : https ://doi.org/10.1016/j.neucom.2020.08.017.

TIEDEMANN J. (2012). Parallel data, tools and interfaces in OPUS. In N. CALZOLARI, K. CHOUKRI, T. DECLERCK, M. U. DOĞAN, B. MAEGAARD, J. MARIANI, A. MORENO, J. ODIJK & S. PIPERIDIS, Éds., *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, p. 2214–2218, Istanbul, Turkey : European Language Resources Association (ELRA).

VASWANI A., SHAZEER N., PARMAR N., USZKOREIT J., JONES L., GOMEZ A. N., KAISER L. & POLOSUKHIN I. (2017). Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, p. 6000–6010, Red Hook, NY, USA : Curran Associates Inc.

VILAR D., FREITAG M., CHERRY C., LUO J., RATNAKAR V. & FOSTER G. (2023). Prompting PaLM for translation : Assessing strategies and performance. In A. ROGERS, J. BOYD-GRABER & N. OKAZAKI, Éds., *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, p. 15406–15427, Toronto, Canada : Association for Computational Linguistics. DOI : 10.18653/v1/2023.acl-long.859.

WANG Y., WANG L., LI Y., HE D. & LIU T. (2013). A theoretical analysis of NDCG type ranking measures. In S. SHALEV-SHWARTZ & I. STEINWART, Éds., *Proceedings of the 26th Annual Conference on Learning Theory*, volume 30 de *Proceedings of Machine Learning Research*, p. 25–54, Princeton, NJ, USA : PMLR.

XIA M., HUANG G., LIU L. & SHI S. (2019). Graph based translation memory for neural machine translation. *Proceedings of the AAAI Conference on Artificial Intelligence*, **33**(01), 7297–7304. DOI : 10.1609/aaai.v33i01.33017297.

XU J., CREGO J. & SENELLART J. (2020). Boosting neural machine translation with similar translations. In D. JURAFSKY, J. CHAI, N. SCHLUTER & J. TETREAULT, Éds., *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, p. 1580–1590, Online : Association for Computational Linguistics. DOI : 10.18653/v1/2020.acl-main.144.

XU J., CREGO J. & YVON F. (2023). Integrating translation memories into non-autoregressive machine translation. In A. VLACHOS & I. AUGENSTEIN, Éds., *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, p. 1326–1338, Dubrovnik, Croatia : Association for Computational Linguistics. DOI : 10.18653/v1/2023.eacl-main.96.

ZEBAZE A., SAGOT B. & BAWDEN R. (2024). In-context example selection via similarity search improves low-resource machine translation. *CoRR*, **abs/2408.00397**.

ZHANG B., HADDOW B. & BIRCH A. (2023). Prompting large language model for machine translation : A case study. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23 : JMLR.org.

ZHENG K., WANG L., WANG Z., CHEN B., ZHANG M. & TU Z. (2023). Towards a unified training for Levenshtein transformer. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 1–5. DOI : 10.1109/ICASSP49357.2023.10094646.

A Données Parallèles

Les données en-fr sont une version nettoyée via un filtre basé sur COMETKiwi (Rei *et al.*, 2022b), contruite à partir des données de (Xu *et al.*, 2020). Chaque domaine est associé à une partition de test de 1000 phrases. Le corpus de-en est repris directement de (Aharoni & Goldberg, 2020), également utilisé dans de nombreux travaux (Aharoni & Goldberg, 2020; Cai *et al.*, 2021; Agrawal *et al.*, 2023; Bouthors *et al.*, 2024). Les fichiers de test contiennent 2000 phrases. Le tableau 6 indique les statistiques des domaines considérés.

	Anglais-Français											Allemand-Anglais				
	ECB	EME	Epp	GNO	JRC	KDE	News	PHP	TED	Ubu	Wiki	KDE	Kor	JRC	EME	Sub
size	149k	272k	1,9M	44k	492k	126k	133k	7k	148k	6k	597k	223k	18k	467k	248k	500k
%	3.8	7.0	49.0	1.1	12.7	3.3	3.4	0.2	3.8	0.2	15.4	15,3	1,2	32,1	17,0	34,3

TABLE 6 – Taille de chaque domaine en nombre de phrases et sa proportion.

B Ajustement des Modèles Encodeurs

Dans nos configurations, l'ajustement (finetuning) met à jour l'ensemble des paramètres. Le taux d'apprentissage global est de 1e-4, sauf a et b (éq. 4) qui ont un taux d'appretissage de 1e-2. Chaque source $\mathbf{x}$ est présentée avec trois $\tilde{\mathbf{y}}$ (éq. 3) déterminés par les trois meilleures correpondances obtenues via **`dense+bow`**. Le score de validation est NDCG (Wang *et al.*, 2013).

La recherche d'exemples similaires est utilise la librairie FAISS [6] (Douze *et al.*, 2024).

C Architectures RANMT

Les deux architectures s'appuient sur l'architecture Transformer (Vaswani *et al.*, 2017). Nous entraînons une instance de `TM`^{3}`-LevT` pour chaque paire de langues en utilisant les données d'entraînement disponibles. Quant au modèle `NFA`, il est entraîné sur une plus grande quantité de données. Le modèle `NFA` avec la direction de traduction anglais vers français est entraîné sur 8 millions de phrases extraites d'une grande variété de corpus publics. La version avec la direction de traduction anglais vers allemand a quant à elle recours à $11,5$ millions de phrases. Notons qu'il y a un cas de contamination des données de test que nous reportons de manière transparente dans les tableaux de résultats. Certains corpus ont été échantillonnés avec un taux variable correspondant à la probabilité que le modèle `NFA` ait été exposé à une paire de phrase du test lors de son entraînement. Nous nommons cette probabilité *taux de contamination*. Enfin, nous avons recours au modèle de traduction multilingue `NLLB` [7] (Costa-jussà *et al.*, 2024) en tant que *baseline*. Il est par ailleurs utilisé pour rétrotraduire les phrases cibles dans la configuration **`fuzzy-bt`** (voir paragraphe suivant). Nous notons pour ce système un cas de contamination des données ECB, EMEA, JRC-Acquis et OpenSubtitles, indiqué de manière transparente dans les tableaux de résultats.

6. https://faiss.ai/.
7. Implémentation HuggingFace *NLLB-200-distilled-1.3B*.

D Coût Computationnel de la Recherche

Le coût computationnel peut être réparti en deux catégories : un coût fixe (qui ne dépend pas du nombre de phrases à traduire) et un coût variable (proportionnel au nombre de phrases à traduire). La recherche d'exemples pour l'apprentissage correspond à la première catégorie. L'encodage de l'ensemble des phrases de la langue cible par un encodeur (pour les méthodes CLIR) ou l'indexation de la TM (pour les méthodes FM) appartiennent également à la catégorie des coûts fixes. Quant aux coûts variables, la recherche FM requiert la tokénisation de la phrase source puis la recherche (avec ou sans filtre BM25) des phrases les plus similaires selon DL ; les systèmes CLIR doivent encoder la phrase source, puis effectuer une recherche kNN dans l'espace des phrases monolingues de la langue cible. Dans nos expériences, nous avons retiré le filtre pour permettre d'obtenir les plus hauts scores `BLEU` possibles (**`fuzzy-src`**, **`fuzzy-bt`** et **`fuzzy-gold`**), ce qui multiplie la latence de recherche d'exemples par ~ 100. Cependant, lorsqu'un filtre BM25 est appliqué pour présélectionner $L = 100$ phrases, nous pouvons faire le constat suivant : les méthodes FM et CLIR permettent toutes les deux de récupérer l'exemple le plus proche en environ $\sim$ 1ms. Nous avons obtenu ce résultat sur le corpus ECB (150k phrases) avec des conditions optimisées :

— les méthodes de recherche FM sont effectuées sur un CPU à 8 cœurs, parallélisant la recherche d'exemples sur 8 phrases sources ;

— concernant les systèmes CLIR, les phrases sources sont encodées dans un lot de 50 phrases et la recherche kNN est parallélisée grâce à l'implémentation de FAISS. Nous avons utilisé un GPU V100-32Go lors de nos expériences, puisqu'un CPU est ~ 100 fois plus lent.

Notons que les systèmes CLIR dans leur configuration optimale nécessitent de conserver en mémoire GPU le modèle encodeur et les vecteurs d'encodage du corpus monolingue de la langue cible, impliquant une limite pratique (lorsque le corpus monolingue est gigantesque par exemple).