# Traduction automatique neuronale basée sur l'attention

**Jérémie Huppé 1854753 , Antoine Daigneault-Demers 1879075 , Fabrice Charbonneau 1798064**
École Polytechnique de Montréal

## Résumé

Ce projet porte sur la traduction automatique neuronale basée sur l'attention. En se basant sur les expériences de l'article de Luong et al.[1], le but de ce travail est d'apprendre sur les techniques qu'ils ont utilisées, répliquer l'architecture neuronale présentée dans ce dernier et de comparer les résultats obtenus en utilisant un autre jeu de données. Luong et al. ayant utilisé un jeu de données pour la traduction de l'anglais à l'allemand, et vice-versa, nous proposons d'entraîner un modèle récurrent avec de attention à celui de Luong et al., mais en utilisant des phrases en anglais et en français. Pour ce projet, nous avons testé trois différentes architectures, dont le seul élément variable était l'attention. Nos configurations se résument à sans attention, puis avec les attentions globale et *local-p* présentées par Luong et al.[1]. Nous avons obtenu des BLEU scores de 33.71 pour notre architecture avec l'attention globale, 31.36 pour l'attention locale en utilisant moins d'*epochs*, et 30.17 pour l'architecture sans attention. Nous montrons des résultats similaires à ceux de Luong et al. et confirmons que l'utilisation d'un module d'attention permet d'obtenir de meilleurs résultats pour la traduction automatique neuronale.

## 1 Introduction

La traduction automatique neuronale, et plus généralement le traitement automatique de la langue, ont évidemment connu d'énormes progrès récemment dus à l'avancement des techniques d'apprentissage profond. Notamment, le gain constant en performance de calculs des processeurs et des cartes graphiques ont permis de pouvoir appliquer des concepts qui restaient théoriques sans ces gains de performance, permettant d'entraîner des modèles sur des grands jeux de données et ainsi les rendre capables de traduire du texte. Ces concepts se rapportent plus précisément aux réseaux de neurones récurrents, dont les LSTMs introduits en 1997[2]. La qualité des traductions automatiques s'est également améliorée à l'aide de l'utilisation d'autres pratiques telles que le prolongement lexical (*embedding*), l'attention, l'augmentation de données, et des techniques d'optimisation dans les réseaux de neurones comme le *dropout*, et autres.

### 1.1 Travaux antérieurs

L'article de Luong et al.[1] sur lequel se base ce projet présente des résultats dominant la compétition lors de sa publication. Par exemple, avec les données du *Workshop on Machine Translation* 2014, leur modèle a obtenu un BLEU score de 23.0, alors que le meilleur modèle existant et le gagnant du WMT'2014 avaient respectivement un BLEU score de 21.6 et 20.7. Ce gain de performance est dû à la combinaison de techniques déjà reconnues et l'innovation au niveau de l'attention provenant des auteurs, plus précisément une attention globale et une attention locale (nommée *local-p*). Les principes se résument à considérer tous les autres mots dans la phrase à traduire afin de tenter de traduire un mot donné pour l'attention globale, alors que l'attention locale pondère la contribution des autres mots selon leur position relative à celle d'une position donné. Les auteurs indiquent que l'incorporation de l'attention permet d'améliorer la traduction de manière générale, et permet de mieux traduire des phrases de longue taille ou contenant des noms propres.

Luong et al. mentionnent d'ailleurs que leur concept d'attention locale dans du texte est en partie inspiré d'une attention sélective présentée par Gregor et al.[3], dans un contexte de génération d'images. Le principe est semblable dans le sens où Gregor et al. utilisaient une approche permettant au modèle de sélectionner une sous-image, agissant comme un zoom. Luong et al. comparent ce zoom dans une image à un zoom à l'intérieur de la phrase à traduire.

Les études antérieures à celle de Luong et al.[1] mentionnées par ceux-ci comme celle de Jean et al.[4] de l'université de Montréal en 2015 et Buck et al.[5] obtenaient des résultats de l'état de l'art en utilisant respectivement du décodage pour un vocabulaire très grand, et un n-gramme en incluant les comptes bas et en éliminant les duplications.

Enfin, il est important de noter que de nombreuses et importantes percées ont été faites dans le domaine, ultérieurement à celle de l'article de Luong et al. en question. Par exemple, Google effectue de la recherche afin d'améliorer leur plateforme de traduction en ligne, et a proposé en 2017 une solution permettant d'effectuer de la traduction *one shot* en utilisant de l'apprentissage par transfert et un seul réseau pour plusieurs paires de langues, en plus

d'obtenir de meilleurs BLEU scores avec les données WMT pour un seul modèle au moment de la publication[6].

## 2 Approche théoritique

Le sujet de notre projet se base sur les réseaux récurrents, particulièrement les réseaux LSTM, et les techniques d'attention globale et locale. Les prochaines sous-sections couvriront plus ou moins grossièrement la théorie de ces concepts, ainsi qu'un résumé de l'architecture permettant d'appliquer ces techniques à la traduction automatique neuronale.

### 2.1 Réseaux récurrents

En reprenant les explications provenant de *Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) Network*[7], nous pouvons débuter par énoncer le problème d'exprimer une fonction modélisant l'évolution d'un signal, un vecteur de dimension $d$, selon le temps $t$:

$$\frac{d\vec{s}(t)}{dt} = \vec{f}(t) + \vec{\phi}$$

où $\vec{s}(t)$ est la valeur d'un vecteur du signal au temps $t$, $\vec{f}(t)$ est une fonction décrivant le comportement générale de l'évolution du signal $\vec{s}(t)$ et $\vec{\phi}$ un vecteur constant.
Un cas spécial de la fonction $\vec{f}(t)$ est le modèle additif suivant:

$$\vec{f}(t) = \vec{a}(t) + \vec{b}(t) + \vec{c}(t)$$

qui se retrouverait dans la littérature des neurosciences afin de modèliser le comportement des neurones. En sautant plusieurs étapes comme résoudre l'équation et en attribuant des fonctions plus concrètes à $\vec{a}(t)$, $\vec{b}(t)$ et $\vec{c}(t)$, il est possible d'en arriver à définir les termes suivants:

$$\vec{s}[n] = W_s\vec{s}[n-1] + W_r\vec{r}[n-1] + W_x\vec{x}[n] + \vec{\theta}_s$$
$$\vec{r}[n] = G(\vec{s}[n])$$

avec $W_s$, $W_r$ et $W_x$ des matrices de poids, $\vec{\theta}_s$ un vecteur de biais et $G(z)$ la fonction d'activation correspondant à la tangente hyperbolique $tanh$. Cela peut ainsi être visualisé sous une cellule RNN tel que montré sur la figure 1, en laissant tomber le biais.

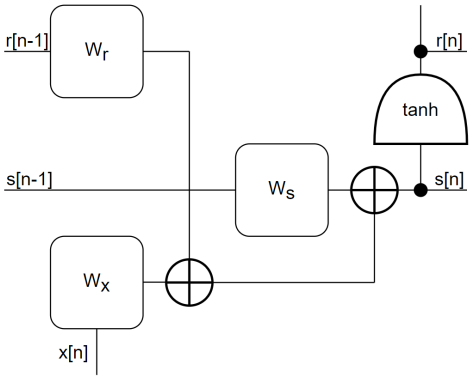


Figure 1: Cellule RNN de base

L'étape suivante est d'introduire le concept de *déroulement* à la figure 2, qui encore une fois découle d'un raisonnement mathématique que nous laisserons de côté. Cela permet d'appliquer le système plusieurs fois de suite sur les différentes valeurs de $\vec{x}$ et simule une mémoire en prenant en compte les sorties de la cellule précédente.

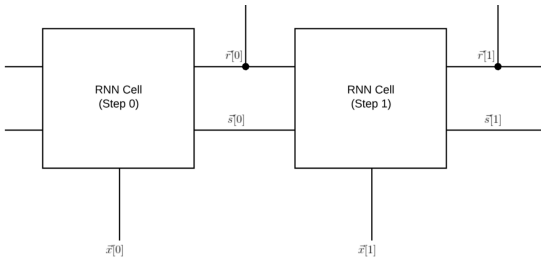


Figure 2: Déroulement de cellules RNN [7]

Les paramètres restent les mêmes pour chaque cellule dans le modèle déroulé. Tout comme un réseau de neurones classique, il est possible de mettre à jour les poids des matrices $W$ en calculant le gradient à partir de la sortie du réseau et des valeurs attendues, en effectuant une propagation arrière, nommée dans ce contexte BPTT (*Back Propagation Through Time*). Le calcul du gradient s'exprime de façon générale pour un RNN sous la forme suivante, utilisant la notation de la figure 3:

$$\frac{\partial E_2}{\partial U} = \frac{\partial E_2}{\partial h_2}\left(x_2^T + \frac{\partial h_2}{\partial h_1}\left(x_1^T + \frac{\partial h_1}{\partial h_0}x_0^T\right)\right)$$

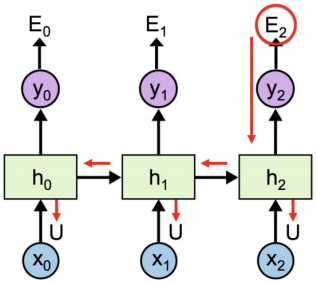


Figure 3: BPTT à l'intérieur d'un RNN (tiré du blog de Chris Olah et de la présentation de Nan Rosemary Ke - MILA)

En raison du fait que le nombre de cellules dans le modèle RNN déroulé peut être arbitrairement grand, les phénomènes d'explosion de disparition du gradient peuvent compliquer l'entraînement. Dans notre contexte de traduction, cela correspond à taille des phrases à traduire, plus celles-ci sont longues, plus le nombre de cellules RNN sera grand, augmentant ainsi les chances que ces problèmes surviennent.

### 2.2 LSTM (*Long Short-Term Memory*)

Les réseaux LSTM ont été inventés dans le but de régler le problème de disparition du gradient[7]. Ils complexifient les cellules RNN de base en ajoutant des *gates* afin de contrôler

les différents flux. La figure 4 présente l'architecture générale d'une cellule LSTM.

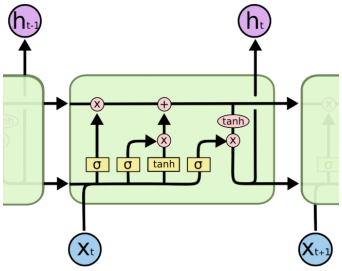


Figure 4: Cellule LSTM (tiré du blog de Chris Olah)

Les différentes *gates* présentes dans la cellule permettent de contrôler la mémoire avec les *forget gates*, le contenu et la quantité de l'information dans la cellule avec les *input gates*, ainsi que les sorties de la cellule. Encore une fois, nous évitons d'entrer trop en détails en ce qui concerne les raisonnements mathématiques justifiant ces choix. Par ailleurs, l'utilisation des LSTMs engendre un nombre de paramètres plus important.

## 2.3 Encodeur/décodeur

Une particularité importante de la traduction automatique est qu'il s'agit d'un problème *many to many*, c'est-à-dire où à la fois l'entrée et la sortie sont composées de plusieurs éléments (mots) dépendant du temps (ou, correspondant à une position dans un espace à une dimension, dans notre cas la position du mot dans la phrase). Pour pallier à ce problème particulier, il est possible d'utiliser un RNN et de diriger les sorties des cellules "décodeuses" vers la prochaine afin de la considérer comme l'entrée et ainsi former des phrases complètes. Il est à noter que lors de l'entraînement d'un tel réseau, il est commun d'utiliser la phrase "réponse" comme les entrées des cellules "décodeuses" afin d'assurer une bonne mise à jour des poids. Lors de la phase de test ou lors de l'utilisation pratique du modèle, le mot précédent prédit sera utilisé comme entrée pour la prochaine cellule de la couche du bas de la partie décodeur, mais lors de l'entraînement cela ne fait pas vraiment de sens, d'où utiliser plutôt la phrase référence (ou "réponse"). La figure 5 présente l'architecture générale d'un encodeur/décodeur.

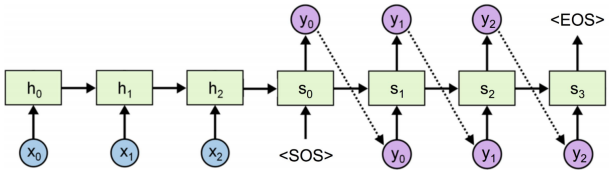


Figure 5: Encodeur/décodeur (Chris Olah)

Il est à noter que la figure 5 n'indique pas que les cellules $h$ de la partie encodeur possèdent des sorties, mais en réalité elles en ont (des *hidden states*) et elles seront pertinentes afin de pouvoir appliquer les mécanismes d'attention de la prochaine section. De plus, il est possible de superposer plusieurs couches (une couche correspondant à la *lignes* sur la figure 5) à l'aide de ces sorties.

## 2.4 Mécanismes d'attention

Les mécanismes d'attention sont basés sur le principe que les mots dans une phrase sont directement influencés par les autres mots de la phrase. Par exemple, l'attention globale suppose que si une phrase contient des mots en rapport avec un sujet particulier, les mots en lien avec ce sujet sont plus probables d'apparaître. L'attention locale pousse le principe plus loin en supposant que les mots qui sont physiquement plus proches d'un mot à prédire dans une phrase apportent une contribution plus grande que celles des mots qui sont plus loin. Luong et al. présentent ces mécanismes comme pouvant améliorer les performances de traduction automatique, en venant compléter un modèle *encoder/decoder*.

### Attention globale

L'attention globale se résume à pondérer les sorties des *hidden states* des cellules qui composent l'encodage pour calculer un vecteur de contexte utilisé dans le calcul de la sortie finale. La figure 6 présente le contenu de la couche d'attention, qui contient le vecteur de contexte ainsi que des poids d'alignement global.

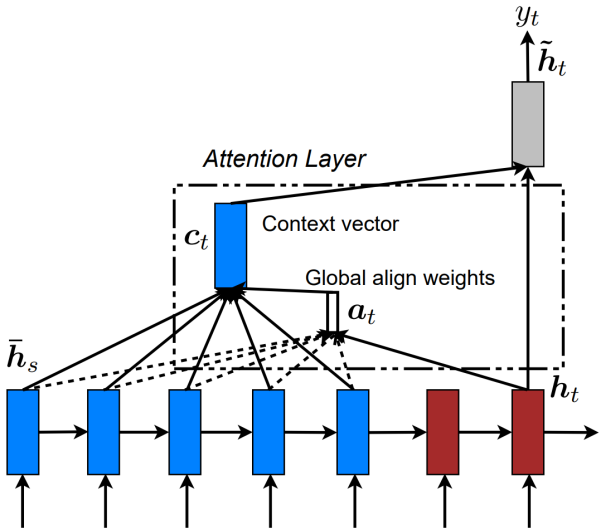


Figure 6: Modèle de l'attention globale (Luong et al.)

Luong et al. précisent la fonction $\tilde{h}_t = \tanh\left(W_c\left[c_t; h_t\right]\right)$ pour la sortie finale, et à chaque temps $t$, le modèle calcule le vecteur d'alignement $a_t$ avec

$$a_t(s) = \text{align}\left(h_t, \bar{h}_s\right) = \frac{\exp\left(\text{score}\left(h_t, \bar{h}_s\right)\right)}{\sum_{s'} \exp\left(\text{score}\left(h_t, \bar{h}_{s'}\right)\right)}$$

et

$$\text{score}\left(h_t, \bar{h}_s\right) = \begin{cases} h_t^\top \bar{h}_s & \text{dot} \\ h_t^\top W_a \bar{h}_s & \text{general} \\ v_a^\top \tanh\left(W_a\left[h_t; \bar{h}_s\right]\right) & \text{concat} \end{cases}$$

Finalement, le vecteur de contexte global $\boldsymbol{c}_t$ est calculé en tant que moyenne pondérée des $\overline{\boldsymbol{h}}_s$ selon $\boldsymbol{a}_t$. Le modèle peut ainsi apprendre des poids dans sa couche d'attention et profiter de la valeur ajoutée de ce mécanisme, en considérant tous les mots de la phrase d'entrée.

**Attention locale (*local-p*)**

Les auteurs indiquent que l'attention globale peut engendrer des problèmes en raison des poids associés à chacun des autres mots du texte à traduire. Cela peut s'avérer problématique au niveau des calculs additionnels, et est difficile à gérer pour de plus longues séquences, car cela implique plus de poids à apprendre. L'attention locale permet de limiter le mécanisme en ne prenant en compte qu'un sous-ensemble des mots de la phrase originale, ceux ayant une position près de celle du mot à prédire. De plus, cela facilite l'entraînement en simplifiant le modèle, diminuant le nombre des poids à apprendre. Afin d'atteindre un tel comportement, Luong et al. proposent d'ajouter au modèle précédent une fenêtre limitant les sorties $\overline{\boldsymbol{h}}_s$ pour le calcul de $\boldsymbol{a}_t$ et $\boldsymbol{c}_t$ en générant une valeur d'alignement positionnel $p_t$ pour chaque mot *target* à prédire, en fonction de sa position. Cela est montré à la figure 7.

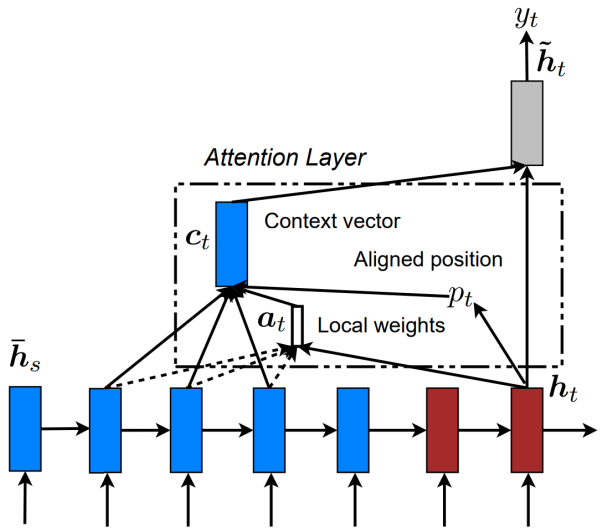


Figure 7: Modèle de l'attention locale (Luong et al.)

Le vecteur d'alignement positionnel est calculé de façon prédictive, contrastant avec l'alignement monotone dans une autre version de l'attention des auteurs nommée *local-m*:

$$p_t = S \cdot \text{sigmoid}\left(\boldsymbol{v}_p^\top \tanh\left(\boldsymbol{W_p}\boldsymbol{h}_t\right)\right)$$

Où $\boldsymbol{W_p}$ et $\boldsymbol{v}_p$ sont des paramètres apprenables par le modèle. En plus de la limitation apportée par la fenêtre, les auteurs appliquent une gaussienne tronquée centrée en $p_t$. Cela modifie ainsi les poids du vecteur d'alignement $\boldsymbol{a}_t(s)$:

$$\boldsymbol{a}_t(s) = \text{align}\left(\boldsymbol{h}_t, \overline{\boldsymbol{h}}_s\right) \exp\left(-\frac{(s-p_t)^2}{2\sigma^2}\right)$$

## 3 Méthodologie

Notre implémentation a été faite en Python avec les librairies Tensorflow et Keras. L'ensemble de données utilisé est l'ensemble French-English tiré du site manythings.org/anki. Seules les 60 000 premières entrées ont été utilisées comme exemples pour l'entraînement et la validation afin de limiter le temps d'entraînement du modèle. Les modèles ont été entraînés sur 20 *epochs*, ou jusqu'à ce que le *loss* ne change presque plus. Nous avons utilisé l'optimiseur Adam, un *batch size* de 128 et un *dropout* de taux 0,2.

Un prétraitement a été effectué sur les exemples avant de procéder à l'entraînement, inspiré du tutoriel de Tensorflow[8]. Ce dernier inclut le retrait de tout caractère non alphanumérique et qui ne fait pas partie des signes de ponctuations de base (. , ! et ?) ainsi que des accents, la séparation des contractions en deux mots distincts et l'ajout des tokens <*start*> et <*end*> aux bouts de chaque phrase. Les phrases étaient ensuite tokenisées, puis du padding a été ajouté afin que toutes les phrases aient la même taille. 80% des données utilisées ont servi pour l'entraînement et 20% pour l'ensemble de test.

Tous les modèles ont la même architecture de base. L'encodeur et le décodeur sont tout deux composés de deux couches LSTM à 1000 cellules. Les mots sont représentés par des *embeddings* de taille 1000. La différence au niveau de nos modèles se situe à l'utilisation de l'attention pour la prédiction. Notre premier modèle ne possède pas de tel mécanisme et la sortie $\boldsymbol{h}_t$ est utilisée directement pour faire les prédictions. Notre deuxième modèle utilise l'attention globale et notre dernier modèle utilise l'attention *local-p*, toutes deux présentées par Luong et al.[1]. Pour l'attention *local-p*, nous avons gardé la variable de la taille de la fenêtre à 10 même si les phrases que nous avons utilisées pour l'entraînement avaient une taille de 9 au maximum. Les effets de l'attention locale sont toutefois conservés puisqu'une attention particulière est mise autour de *pt* avec la gaussienne. Pour le score, nous avons utilisé la fonction *general* pour l'attention *local-p* et la fonction *dot* pour l'attention globale ce qui correspond aux meilleurs combinaisons *score - type d'attention* présenté dans l'article de Luong et al.[1].

Il y a plusieurs différences notables entre notre implémentation et celle de Luong et al. Premièrement, leur réseau possédait quatre couches LSTM. Ensuite, ils n'utilisaient pas l'optimiseur Adam, mais plutôt une descente du gradient stochastique de base avec un taux d'apprentissage de 1 en divisant ce dernier par 2 à chaque epoch passée la cinquième. De plus, leur implémentation utilisait la technique de renversement de la phrase source pour l'apprentissage.

Finalement, afin d'évaluer nos modèles, le BLEU score a été utilisé tout comme dans l'article de Luong.

## 4 Résultats

Notre implémentation et nos résultats sont disponibles sur notre répertoire GitHub en cliquant **ici**.

Afin de comparer nos différents modèles, nous présentons dans cette section l'évolution du *loss* sur notre ensemble de

validation, le BLEU score en fonction de la taille de la phrase traduite et le BLEU score moyen de chacun d'entre eux.

Au tableau 1, il est possible de voir le temps d'entraînement moyen par epoch pour chacun des modèles.

Tableau 1: Temps moyen d'exécution par epoch pour chaque configuration

| Configuration | Temps moyen par epoch |
|---|---|
| Sans Attention | ∼110 sec |
| Attention *Local-p* | ∼1750 sec |
| Attention Globale | ∼115 sec |

À la figure 8, il est possible de voir l'évolution de la fonction de perte de nos trois modèles.

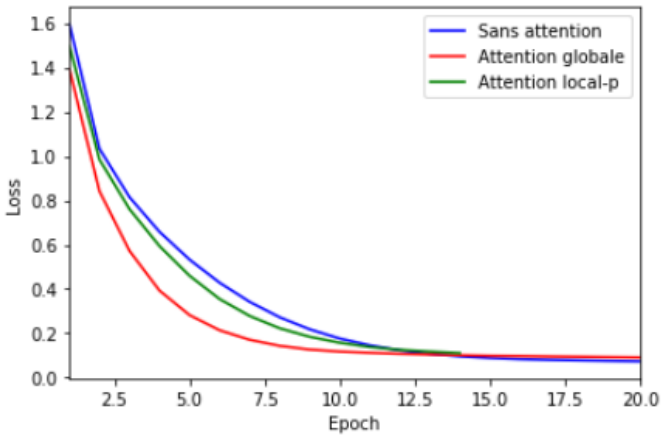


Figure 8: *Loss* selon les *epochs* écoulés pour chacune des architectures

À la figure 9, il est possible d'observer le BLEU score moyen en fonction de la longueur de la phrase traduite pour nos trois modèles.

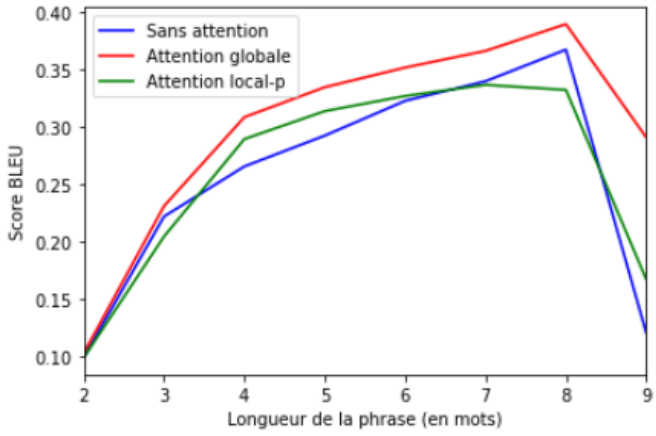


Figure 9: BLEU score selon la longueur des phrases pour chacune des architectures

Au tableau 2, il est possible de voir le BLEU score moyen global pour chacune de nos architecture.

Tableau 2: BLEU score moyen sur l'ensemble de test pour chaque configuration

| Configuration | BLEU |
|---|---|
| Sans Attention | 30.17% |
| Attention *Local-p* | 31.36% |
| Attention Globale | 33.71% |

# 5 Discussion

## 5.1 Analyse des résultats

Premièrement, il est possible de constater que notre implémentation de l'architecture avec la configuration *local-p* prend significativement plus de temps que les autres configurations: 1750 sec par epoch comparativement à 110 sec par epoch pour la configuration sans attention et 115 sec par epoch pour la configuration avec l'attention globale.

Puisque le temps nécessaire pour entraîner notre configuration avec l'attention *local-p* était très grand nous avons dû effectuer un entraînement de seulement 14 *epochs* comparativement à 20 *epochs* pour les deux autres configurations. Cela pourrait potentiellement justifier le fait que le résultat de BLEU score moyen pour la configuration avec l'attention *local-p* est plus bas que le BLEU score moyen de la configuration avec l'attention globale ce qui est contraire aux résultats obtenus par Luong et al.[1].

Le temps d'entraînement significativement plus grand pour notre implémentation de l'architecture avec la configuration *local-p* peut probablement être expliqué par le fait que nous avons utilisé des opérations coûteuses sur les tenseurs. Étant inexpérimentés avec l'utilisation de la librarie Tensorflow nous avons eu du mal à extraire la fenêtre de mots à différentes positions *pt* pour chaque exemple de la *mini-batch*. Plus précisément, nous avons premièrement effectué une suite d'opérations matricielles pour isoler les mots de la fenêtre avec un isolement du style *one-hot*: toutes les sorties $h_s$ de l'encodeur à considérer ont leur valeur tandis que les sorties à ne pas considérer ont leur valeur mise à 0. Ensuite, d'autre opérations matricelles sont effectuées pour évaluer le score. Finalement, une fonction softmax doit être appliquée sur le score trouvé. Or, le score que nous avons trouvé étant sous forme matricielle avec des zéros aux mots qui ne doivent pas être considérés une application banale de softmax prendrait en considération les zéros dans le calcul du softmax ce que nous voulons éviter. Nous avons donc implémenté un ensemble d'opérations afin que la fonction softmax s'applique uniquement aux éléments qui ne sont pas égaux à zéros. Nous soupçonnons que notre implémentation de cette fonction nommée *non_zero_softmax_on_matrix* est le goulot d'étranglement de notre implémentation de l'architecture avec la configuration *local-p*. Nous avons vérifié à plusieurs reprise que notre implémentation de l'attention *local-p* effectuait les bonnes opérations. Par contre, il est possible d'affirmer que ces opérations ne sont pas effectuées de la manière la plus optimale possible.

D'ailleurs, tel que mentionné plus tôt, la fenêtre utilisée pour notre implémentation de l'attention locale est fixée à 10, alors que les phrases ne dépassent pas une longueur de 9. Il serait intéressant de voir si diminuer cette valeur améliorerait les résultats.

Il est possible de constater dans le figure 8 que la *loss* diminue le plus rapidement pour le modèle avec l'attention globale, puis celui avec l'attention *local-p*, puis celui sans attention. Cependant, les trois modèles convergent approximativement vers la même valeur. Finalement, il est à noter que nous avons arrêté l'entraînement du modèle avec l'attention *local-p* après 14 *epochs* puisque le gradient s'était stabilisé et que l'entraînement prenait beaucoup plus de temps que pour les deux autres architectures.

Le figure 9 montre que le modèle avec l'attention globale donne les meilleurs résultats pour toutes les longueurs de phrases. Étonnamment, l'attention *local-p* n'offre pas les scores les plus élevés, contrairement aux résultats de Luong et al.[1]. Nous pensons que cela est dû au fait que nous avons arrêté l'entraînement de ce modèle plus tôt. Il aurait aussi été intéressant d'utiliser des phrases plus longues afin de voir à quel point l'attention aide dans cette situation et si l'attention *local-p* aurait offert de meilleures performances que l'attention globale.

Il est important de noter que les données utilisées sont les 60000 premières de l'ensemble de données qui est classé en ordre croissant de nombre de mots. Cela expliquerait les mauvais BLEU scores pour les phrases de longueurs 9: nous aurions tout simplement coupé au milieu des phrases de taille 9 et nous avons donc moins d'exemples de cette taille.

Il est possible de constater dans le tableau 2 que la configuration avec l'attention globale a le meilleur BLEU score et que la configuration sans attention a le moins bon BLEU score. Des explications justifiant pourquoi l'attention *local-p* n'a pas un meilleur BLEU score que l'attention globale ont été présentées dans les paragraphes précédents. Finalement, les moyennes des BLEU scores obtenus sur l'ensemble de test pour chaque configuration sont plus grands que les BLEU scores obtenus dans l'article de Luong et al.[1] cela peut être expliqué par le fait que nous un avons effectué nos tests sur un ensemble d'entraînement contenant seulement des phrases relativement courtes (moins de 9 mots) et que dans l'article de Luong et al.[1] ces derniers ont utilisé un ensemble de données avec des phrases beaucoup plus longues. Étant plus difficile d'obtenir un bon BLEU score sur des phrases plus longues cela explique la différence avec les BLEU scores que nous avons obtenus et les BLEU score présentés dans l'article de Luong et al.[1].

Enfin, les tableaux 3 à 5 en annexe présentent les traductions réalisées par nos différents modèles. Ces résultats montrent qu'en général, les phrases traduites par nos modèles ont une qualité qui reflète assez bien leur BLUE score. Par exemple, le modèle avec l'attention globale réussit de bien meilleures traductions que le modèle sans attention. Il est aussi possible de remarquer que toutes les traductions réussies à 100% par le modèle sans attention sont réussies par les deux autres et que celles réussies par le modèle avec attention *local-p* le sont aussi par le modèle avec l'attention générale.

Une critique que nous avons envers le BLUE score est qu'il représente parfois très mal à quel point une traduction a bien été réussie. Par exemple, si un modèle prédit "Où vous êtes vous rendues?" et non "Ou es tu allé?" pour la phrase "Where did you go?", cela donne un score de 5% alors que la signification est la même. Au contraire, si un modèle prédit "Il est arrivé au Japon" alors que la référence est "Il est arrivé à minuit" pour la phrase "He arrived at midnight", cela donne un score de 22%, alors que la signification n'est pas du tout la même. Au final, ces écarts doivent environ se compenser pour donner une idée générale de la qualité des traductions, mais il reste que ce système de notation semble pouvoir être amélioré.

## 5.2 Critique sur l'approche utilisée pour apprendre le sujet

Nous avons initialement essayé d'implémenter le projet sans s'inspirer de code existant. N'ayant jamais implémenté d'architectures neuronales avec des librairies comme Pytorch ou TensorFlow nous avons initialement eu beaucoup de difficulté à faire fonctionner une architecture LSTM profonde. Nous avons dédié beaucoup de temps à essayer plusieurs manières de coder l'architecture et de l'entraîner sans réussir à faire fonctionner un système de traduction complet. Nous avons ensuite décidé que pour réussir à avoir une architecture fonctionnelle et produire des résultats à temps pour la remise nous allions avoir besoin d'une ressource supplémentaire. Notre choix s'est arrêté sur le tutoriel Neural machine translation with attention[8] de Tensorflow. Ce tutoriel présente une architecture neuronale pour la traduction de phrases en espagnol en phrases en anglais. L'architecture présentée dans ce tutoriel est l'architecture proposée par Bahdanau et al. Cette architecture est différente de l'architecture proposée par Luong et al.[1]. En effet, l'architecture de Bahdanau et al. est consitutée d'un Gated Recurrent Unit - Cho et al. 2014. tandis que celle proposée par Luong et al.[1] est une pile de LSTM. De plus, le calcul de l'attention est complètement différent. Étant limités dans le temps, nous avons décidé de réutiliser le prétraitement proposé dans le tutoriel de Tensorflow[8] afin de pouvoir se concentrer sur l'implémentation de l'architecture de Luong et al.[1].

En rétrospective nous considérons que nous avons beaucoup appris. Le fait d'avoir premièrement tenté d'implémenter le tout sans inspiration nous a forcé à lire en profondeur la documentation de Tensorflow et de s'habituer avec la terminologie utilisée. La mise en place d'un système neuronal complet pour la traduction de phrases s'est avérée plus compliquée que nous l'avions initialement anticipé. Nous jugeons donc que l'utilisation du tutoriel Neural machine translation with attention[8] de Tensorflow pour nous donner une base s'est avéré une bonne idée pour contribuer à notre apprentissage du sujet. En effet, ce tutoriel n'utilisant pas la même architecture que celle proposée par Luong et al.[1] cela nous a poussé à réécrire et comprendre en profondeur cette architecture.

## 6 Conclusion

En conclusion, ce projet de traduction automatique neuronale avec attention s'est basé sur les expériences de l'article de Luong et al.[1]. Nous avons en partie répliqué l'architecture présentée dans ce dernier avons pu comparer nos résultats aux leurs. Luong et al. ayant utilisé un jeu de données pour la traduction de l'anglais à l'allemand, et vice-versa, nous proposons d'entraîner un modèle récurrent avec de l'attention semblable à celui de l'article de 2015, mais en utilisant des phrases en anglais et en français. Nous nous sommes limités à la traduction de l'anglais vers le françias. Trois différentes architectures ont été testées. Nos configurations se résument à sans attention, puis avec les attentions globale et *local-p* présentées par Luong et al.[1]. Nous avons obtenu des BLEU scores de 33.71 pour notre architecture avec l'attention globale, 31.36 pour l'attention locale en utilisant moins d'*epochs*, et 30.17 pour l'architecture sans attention. Nous montrons des résultats similaire à ceux de Luong et al. et confirmons que l'utilisation d'un module d'attention permet d'obtenir de meilleurs résultats pour la traduction automatique neuronale. Nous n'avons pas trouvé que l'attention *local-p* donnait de meilleurs résultats que l'attention globale, mais cela est possiblement dû au fait que notre entraînement a été effectué sur un ensemble de données et un nombre d'*epochs* limités.

Il serait intéressant de refaire l'expérience, mais en entraînant l'architecture avec l'attention *local-p* sur un plus grand nombre d'*epochs* afin de pouvoir vérifier les résultats obtenus par Luong et al. De plus, avec plus de ressources et de temps, il serait intéressant d'entraîner les architectures sur une plus grande partie des données disponibles, et donc avec des phrases plus longues. Aussi, l'utilisation du même jeu de données qu'ont pris Luong et al. pourrait aider à confirmer les résultats obtenus. Une autre piste d'amélioration serait l'optimisation au niveau de l'attention locale afin de faire diminuer le temps des calculs pris pour l'entraînement de notre modèle. Enfin, il serait intéressant d'explorer les autres avenues proposées par la recherche plus récente en ce qui concerne la traduction automatique.

## Références


[1] Luong Minh-Thang, Pham Hieu, and D. Manning Christopher. Effective approaches to attention-based neural machine translation. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, EMNLP 2015:1412–1421, 2015.

[2] Hochreiter Sepp and Schmidhuber Jürgen. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[3] Gregor Karol, Danihelka Ivo, Graves Alex, and Wierstra Daan. DRAW: A recurrent neural network for image generation. *CoRR*, abs/1502.04623, 2015.

[4] Sébastien Jean, Kyunghyun Cho, Roland Memisevic, and Yoshua Bengio. On using very large target vocabulary for neural machine translation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1–10, Beijing, China, 2015. Association for Computational Linguistics.

[5] Buck Christian, Heafield Kenneth, and Ooyen Bas van. N-gram counts and language models from the common crawl. *LREC*, 2014.

[6] Melvin Johnson, Mike Schuster, Quoc V Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, et al. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351, 2017.

[7] Alex Sherstinsky. Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network. *arXiv preprint arXiv:1808.03314*, 2018.

[8] Tensorflow.org. Neural machine translation with attention. https://www.tensorflow.org/tutorials/text/nmt_with_attention.

[9] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

# A Tableaux des résultats de traductions

Tableau 3: Sans Attention

| # | source | traduction | référence | # mots | BLEU |
|---|---|---|---|---|---|
| 1 | tom is not here . | tom n est pas ici . <end> | tom n est pas ici . | 5 | 100% |
| 2 | it s an old picture . | c est un vieux tableau . <end> | c est une vieille image . | 6 | 10% |
| 3 | this is your fate . | c est votre destinee . <end> | c est votre sort . | 5 | 29% |
| 4 | let s have sushi . | avons y experience . <end> | prenons des sushi . | 5 | 8% |
| 5 | where did you go ? | ou vous etes vous rendu ? <end> | ou es tu alle ? | 5 | 5% |
| 6 | how s your morning been ? | comment s est passee votre matinee ? <end> | comment votre matinee s est elle passee ? | 6 | 10% |
| 7 | he arrived at midnight . | il est parti en amerique . <end> | il est arrive a minuit . | 5 | 10% |
| 8 | she kicked him . | elle lui a donne un coup de pied . <end> | elle lui donna un coup de pied . | 4 | 51% |
| 9 | i want to go to boston . | je veux aller a boston . <end> | je veux aller a boston . | 7 | 100% |
| 10 | i spilled the milk . | j ai crochete la porte . <end> | j ai renverse le lait . | 5 | 10% |
| 11 | i am looking at that . | je suis en train de faire ca . <end> | je regarde ca . | 6 | 7% |
| 12 | she suddenly kissed me . | elle m a soudainement embrasse . <end> | elle m a soudainement embrassee . | 5 | 54% |
| 13 | he kept his word . | il a tenu parole . <end> | il tint parole . | 5 | 13% |
| 14 | i told him what to do . | je lui ai dit quoi faire . <end> | je lui ai dit quoi faire . | 7 | 100% |
| 15 | come and get it . | venez la chercher . <end> | viens la chercher . | 5 | 40% |
| 16 | how absurd ! | c est affreux ! <end> | quelle absurdite ! | 3 | 8% |
| 17 | i want a closer look . | je veux qu on me rende confiance en vous . <end> | je veux regarder de plus pres . | 6 | 5% |
| 18 | watch out for that man . | cherche cet homme la . <end> | guette cet homme la . | 6 | 67% |
| 19 | everyone changes . | tout le monde ment . <end> | tout le monde change . | 3 | 29% |
| 20 | let s eat out tonight . | allons manger ce soir . <end> | ce soir , mangeons dehors . | 6 | 10% |
| 21 | keep driving . | continue a rouler ! <end> | roulez ! | 3 | 8% |
| 22 | i should go . | je devrais y aller . <end> | je devrais partir . | 4 | 13% |
| 23 | they re downstairs . | ils sont en dessous . <end> | ils sont en bas . | 4 | 29% |
| 24 | i d like you to drive . | j aimerais que tu conduises . <end> | je voudrais que ce soit toi qui conduises . | 7 | 6% |
| 25 | i ve done everything . | j ai tout fait . <end> | j ai tout fait . | 5 | 100% |
| 26 | they basked in the sun . | ils ont pris un bain de soleil . <end> | ils se sont dores au soleil . | 6 | 7% |
| 27 | you look nervous . | vous avez l air nerveuses . <end> | vous avez l air nerveuse . | 4 | 54% |
| 28 | it s instinct . | c est totalement . <end> | c est l instinct . | 4 | 15% |
| 29 | tom is mowing the lawn . | tom tond la pelouse . <end> | tom est en train de tondre la pelouse . | 6 | 13% |
| 30 | they all looked happy . | ils avaient l air tous contents . <end> | ils avaient l air tous satisfaits . | 5 | 64% |
| 31 | are you ready ? | etes vous prete ? <end> | es tu pret ? | 4 | 8% |
| 32 | i like the way you sing . | j aime ta maniere de parler . <end> | j aime ta maniere de chanter . | 7 | 64% |
| 33 | where should we go ? | ou devrions nous aller ? <end> | ou devrions nous nous rendre ? | 5 | 23% |
| 34 | tom is bald . | tom est obstine . <end> | tom est chauve . | 4 | 19% |
| 35 | are you happy right now ? | etes vous heureuse en ce moment ? <end> | etes vous heureuses en ce moment ? | 6 | 49% |
| 36 | no problem at all ! | pas de probleme ! <end> | aucun probleme ! | 5 | 17% |
| 37 | put the rifle down . | pose le fusil . <end> | posez le fusil . | 5 | 40% |
| 38 | what do you want now ? | que voulez vous maintenant ? <end> | que veux tu maintenant ? | 6 | 13% |
| 39 | are you ready to do that ? | etes vous prets a faire cela ? <end> | etes vous pret a faire cela ? | 7 | 49% |
| 40 | tom borrowed my car . | tom a rase sa vie . <end> | tom a emprunte ma voiture . | 5 | 10% |

Tableau 4: Attention Globale

| # | source | traduction | référence | # mots | BLEU |
|---|---|---|---|---|---|
| 1 | tom is not here . | tom n est pas ici . <end> | tom n est pas ici . | 5 | 100% |
| 2 | it s an old picture . | c est un vieux tableau . <end> | c est une vieille image . | 6 | 10% |
| 3 | this is your fate . | c est votre destinee . <end> | c est votre sort . | 5 | 29% |
| 4 | let s have sushi . | prenons des sushi . <end> | prenons des sushi . | 5 | 100% |
| 5 | where did you go ? | ou etes vous alles ? <end> | ou es tu alle ? | 5 | 6% |
| 6 | how s your morning been ? | comment ta matinee s est elle passee ? <end> | comment votre matinee s est elle passee ? | 6 | 71% |
| 7 | he arrived at midnight . | il est arrive a six heures . <end> | il est arrive a minuit . | 5 | 43% |
| 8 | she kicked him . | elle lui jeta un coup de pied . <end> | elle lui donna un coup de pied . | 4 | 59% |
| 9 | i want to go to boston . | je veux aller a boston . <end> | je veux aller a boston . | 7 | 100% |
| 10 | i spilled the milk . | j ai renverse le lait . <end> | j ai renverse le lait . | 5 | 100% |
| 11 | i am looking at that . | je suis au courant . <end> | je regarde ca . | 6 | 6% |
| 12 | she suddenly kissed me . | elle m a soudainement embrasse . <end> | elle m a soudainement embrassee . | 5 | 54% |
| 13 | he kept his word . | il a tenu parole . <end> | il tint parole . | 5 | 13% |
| 14 | i told him what to do . | je lui ai dit quoi faire . <end> | je lui ai dit quoi faire . | 7 | 100% |
| 15 | come and get it . | venez la chercher . <end> | viens la chercher . | 5 | 40% |
| 16 | how absurd ! | quelle absurdite ! <end> | quelle absurdite ! | 3 | 56% |
| 17 | i want a closer look . | je veux plus de singapour . <end> | je veux regarder de plus pres . | 6 | 9% |
| 18 | watch out for that man . | fais attention a cet homme la . <end> | guette cet homme la . | 6 | 41% |
| 19 | everyone changes . | tout le monde change . <end> | tout le monde change . | 3 | 100% |
| 20 | let s eat out tonight . | allons voir ce soir . <end> | ce soir , mangeons dehors . | 6 | 10% |
| 21 | keep driving . | avance ! <end> | roulez ! | 3 | 15% |
| 22 | i should go . | je devrais y aller . <end> | je devrais partir . | 4 | 13% |
| 23 | they re downstairs . | ils sont en dessous . <end> | ils sont en bas . | 4 | 29% |
| 24 | i d like you to drive . | j aimerais que tu conduises . <end> | je voudrais que ce soit toi qui conduises . | 7 | 6% |
| 25 | i ve done everything . | j ai tout fait . <end> | j ai tout fait . | 5 | 100% |
| 26 | they basked in the sun . | ils ont pris un bain de soleil . <end> | ils se sont dores au soleil . | 6 | 7% |
| 27 | you look nervous . | tu as l air nerveuse . <end> | vous avez l air nerveuse . | 4 | 51% |
| 28 | it s instinct . | c est clair . <end> | c est l instinct . | 4 | 15% |
| 29 | tom is mowing the lawn . | tom tond la pelouse . <end> | tom est en train de tondre la pelouse . | 6 | 13% |
| 30 | they all looked happy . | ils avaient l air tous contents . <end> | ils avaient l air tous satisfaits . | 5 | 64% |
| 31 | are you ready ? | etes vous prete ? <end> | es tu pret ? | 4 | 8% |
| 32 | i like the way you sing . | j aime votre maniere de vous . <end> | j aime ta maniere de chanter . | 7 | 10% |
| 33 | where should we go ? | ou devrions nous aller ? <end> | ou devrions nous nous rendre ? | 5 | 23% |
| 34 | tom is bald . | tom est chauve . <end> | tom est chauve . | 4 | 100% |
| 35 | are you happy right now ? | etes vous heureuse en ce moment ? <end> | etes vous heureuses en ce moment ? | 6 | 49% |
| 36 | no problem at all ! | pas de probleme ! <end> | aucun probleme ! | 5 | 17% |
| 37 | put the rifle down . | pose le fusil . <end> | posez le fusil . | 5 | 40% |
| 38 | what do you want now ? | que voulez vous maintenant ? <end> | que veux tu maintenant ? | 6 | 13% |
| 39 | are you ready to do that ? | etes vous prete a faire cela ? <end> | etes vous pret a faire cela ? | 7 | 49% |
| 40 | tom borrowed my car . | tom a emprunte ma voiture . <end> | tom a emprunte ma voiture . | 5 | 100% |

Tableau 5: Attention Locale-p

| # | source | traduction | référence | # mots | BLEU |
|---|---|---|---|---|---|
| 1 | tom is not here . | tom n est pas ici . <end> | tom n est pas ici . | 5 | 100% |
| 2 | it s an old picture . | c est une vieille passagere . <end> | c est une vieille image . | 6 | 54% |
| 3 | this is your fate . | c est votre destinee . <end> | c est votre sort . | 5 | 29% |
| 4 | let s have sushi . | prenons les amour ! <end> | prenons des sushi . | 5 | 8% |
| 5 | where did you go ? | ou vous etes vous rendue ? <end> | ou es tu alle ? | 5 | 5% |
| 6 | how s your morning been ? | comment ta matinee est elle propre ? <end> | comment votre matinee s est elle passee ? | 6 | 8% |
| 7 | he arrived at midnight . | il est arrive au japon . <end> | il est arrive a minuit . | 5 | 22% |
| 8 | she kicked him . | elle lui a donne un coup de pied . <end> | elle lui donna un coup de pied . | 4 | 51% |
| 9 | i want to go to boston . | je veux aller a boston . <end> | je veux aller a boston . | 7 | 100% |
| 10 | i spilled the milk . | j ai renverse le lait . <end> | j ai renverse le lait . | 5 | 100% |
| 11 | i am looking at that . | je touche a la main . <end> | je regarde ca . | 6 | 5% |
| 12 | she suddenly kissed me . | elle m a soudainement embrasse . <end> | elle m a soudainement embrassee . | 5 | 54% |
| 13 | he kept his word . | il a tenu parole . <end> | il tint parole . | 5 | 13% |
| 14 | i told him what to do . | je lui ai dit que ca peut faire . <end> | je lui ai dit quoi faire . | 7 | 35% |
| 15 | come and get it . | viens le chercher . <end> | viens la chercher . | 5 | 19% |
| 16 | how absurd ! | comme c est excitant ! <end> | quelle absurdite ! | 3 | 5% |
| 17 | i want a closer look . | je veux un morceau de plus en plus rapide . <end> | je veux regarder de plus pres . | 6 | 7% |
| 18 | watch out for that man . | occupe toi de cet homme la . <end> | guette cet homme la . | 6 | 41% |
| 19 | everyone changes . | tout le monde change . <end> | tout le monde change . | 3 | 100% |
| 20 | let s eat out tonight . | allons manger ceci . <end> | ce soir , mangeons dehors . | 6 | 5% |
| 21 | keep driving . | continue a rouler ! <end> | roulez ! | 3 | 8% |
| 22 | i should go . | je devrais y aller . <end> | je devrais partir . | 4 | 13% |
| 23 | they re downstairs . | ils sont en dessous . <end> | ils sont en bas . | 4 | 29% |
| 24 | i d like you to drive . | j aimerais que tu conduises . <end> | je voudrais que ce soit toi qui conduises . | 7 | 6% |
| 25 | i ve done everything . | j ai tout fait . <end> | j ai tout fait . | 5 | 100% |
| 26 | they basked in the sun . | ils ont pris un bain de soleil . <end> | ils se sont dores au soleil . | 6 | 7% |
| 27 | you look nervous . | tu as l air nerveuse . <end> | vous avez l air nerveuse . | 4 | 51% |
| 28 | it s instinct . | c est verrouille . <end> | c est l instinct . | 4 | 15% |
| 29 | tom is mowing the lawn . | tom tond la pelouse . <end> | tom est en train de tondre la pelouse . | 6 | 13% |
| 30 | they all looked happy . | ils avaient tous l air contentes . <end> | ils avaient l air tous satisfaits . | 5 | 11% |
| 31 | are you ready ? | etes vous prets ? <end> | es tu pret ? | 4 | 8% |
| 32 | i like the way you sing . | j aime votre style . <end> | j aime ta maniere de chanter . | 7 | 8% |
| 33 | where should we go ? | ou devrions nous aller ? <end> | ou devrions nous nous rendre ? | 5 | 23% |
| 34 | tom is bald . | tom est chauve . <end> | tom est chauve . | 4 | 100% |
| 35 | are you happy right now ? | es tu heureuse en ce moment ? <end> | etes vous heureuses en ce moment ? | 6 | 41% |
| 36 | no problem at all ! | pas de probleme ! <end> | aucun probleme ! | 5 | 17% |
| 37 | put the rifle down . | pose le fusil . <end> | posez le fusil . | 5 | 40% |
| 38 | what do you want now ? | que voulez vous maintenant ? <end> | que veux tu maintenant ? | 6 | 13% |
| 39 | are you ready to do that ? | es tu pret a faire ca ? <end> | etes vous pret a faire cela ? | 7 | 18% |
| 40 | tom borrowed my car . | tom a emprunte ma voiture . <end> | tom a emprunte ma voiture . | 5 | 100% |