Apprentissage discriminant de modèles neuronaux pour la traduction automatique

Quoc-Khanh Do — Alexandre Allauzen — François Yvon

LIMSI, CNRS, Univ. Paris-Sud, Université Paris-Saclay,
Campus universitaire bât 508, Rue John von Neumann
F - 91405 Orsay cedex
{dokhanh,allauzen,yvon}@limsi.fr

RÉSUMÉ. *Les méthodes utilisées pour entraîner des réseaux de neurones en traitement des langues reposent, pour la plupart, sur l'optimisation de critères qui sont décorrélés de l'application finale. Nous proposons un nouveau cadre d'apprentissage discriminant pour l'estimation des modèles neuronaux en traduction automatique. Ce cadre s'appuie sur la définition d'un critère d'apprentissage qui prend en compte, d'une part, la métrique utilisée pour l'évaluation automatique de la traduction et, d'autre part, le processus d'intégration de ces modèles au sein des systèmes de traduction automatique. Cette méthode est comparée aux critères d'apprentissage usuels que sont le maximum de vraisemblance et l'estimation contrastive bruitée. Les expériences menées sur les tâches de traduction des séminaires TED Talks et de textes médicaux, depuis l'anglais vers le français, montrent la pertinence d'un cadre d'apprentissage discriminant et l'importance d'une initialisation judicieuse, en particulier dans une perspective d'adaptation au domaine.*

ABSTRACT. *This paper proposes a new discriminative framework to train continuous-space translation models based. This framework relies on the definition of a new objective function that allows us to introduce the evaluation metric in the learning process as well as to consider how the model interacts with the rest of the translation system. This approach is compared with state-of-the-art estimation methods, such as the maximum likelihood training and noise contrastive estimation. Experiments are carried out on two English to French translation tasks using the TED Talks conference series as as well as medical texts. Experimental results show the effectiveness of the proposed approach, specifically in an adaptation scenario. We show that with a tailored initialization scheme, significant improvements can be obtained.*

MOTS-CLÉS : *modèles neuronaux, traduction automatique statistique, apprentissage discriminant.*

KEYWORDS: *neural networks, statistical machine translation, discriminative training.*

TAL. Volume 57 – n° 1/2016, pages 111 à 135

1. Introduction

Les modèles neuronaux occupent aujourd'hui une place importante en traitement automatique des langues (TAL) car ils permettent de construire et de manipuler des représentations numériques *continues* des objets linguistiques, qui améliorent l'efficacité des modélisations statistiques et ont permis des avancées significatives pour de nombreux domaines applicatifs. Historiquement, les modèles de langue neuronaux ont été une des premières réalisations marquantes, avec des applications en reconnaissance automatique de la parole (RAP), depuis les travaux pionniers de Nakamura *et al.* (1990) jusqu'aux développements ultérieurs (Bengio *et al.*, 2003 ; Schwenk, 2007 ; Mnih et Hinton, 2007 ; Le *et al.*, 2011 ; Mikolov *et al.*, 2011). Les modèles neuronaux ont été également appliqués à d'autres tâches complexes de modélisation linguistique, comme par exemple l'analyse syntaxique (Socher *et al.*, 2013), l'estimation de similarité sémantique (Huang *et al.*, 2012), les modèles d'alignement de mots (Yang *et al.*, 2013) ou encore la traduction automatique statistique (TAS) (Le *et al.*, 2012 ; Kalchbrenner et Blunsom, 2013 ; Devlin *et al.*, 2014 ; Cho *et al.*, 2014).

Une des caractéristiques importantes des modèles neuronaux pour le TAL est leur caractère continu. En effet les modèles probabilistes usuels reposent sur une représentation discrète des unités linguistiques considérées (morphèmes, mots, syntagmes, etc.). Ainsi, pour un modèle de traduction à base de segments (Koehn, 2010), l'occurrence d'un segment est considérée comme la réalisation d'une variable aléatoire discrète, dont l'espace de réalisation est l'ensemble des segments observés dans les données d'apprentissage. Au sein de cet espace, il n'existe aucune relation entre les éléments qui permettrait de modéliser une notion de similarité, par exemple sémantique ou syntaxique. Le caractère très inégal des distributions d'occurrences dans les textes implique alors que les modèles résultants sont souvent estimés à partir de petits nombres d'occurrences, qu'ils possèdent une faible capacité de généralisation et que la modélisation du contexte est computationnellement très coûteuse et donc souvent à horizon limité.

Par opposition, les modèles neuronaux (Bengio *et al.*, 2003) se caractérisent par une méthode d'estimation alternative qui se fonde sur une représentation *continue* des unités modélisées et en particulier des mots[1]. Dans le cas, par exemple, d'un modèle de langue, chaque mot du vocabulaire est représenté comme un point dans un espace métrique. La probabilité n-gramme d'un mot est alors une fonction des représentations continues des mots qui composent son contexte. Ces représentations, ainsi que les paramètres de la fonction d'estimation, sont appris conjointement par un réseau de neurones multicouche, une stratégie d'estimation qui permet que les mots partageant des similarités distributionnelles aient des représentations proches. Ainsi, ce type de modèle introduit la notion de similarité entre mots et son exploitation permet une meilleure utilisation des données textuelles. L'intégration de ce type de modèle a également permis des améliorations systématiques et significatives des performances en RAP et en TAS (Schwenk, 2007 ; Le *et al.*, 2011 ; Le *et al.*, 2012). Les représentations

1. Les modèles neuronaux sont souvent qualifiés de modèles continus.

continues peuvent, de plus, servir à de nombreuses tâches, comme par exemple l'étiquetage en parties du discours et en rôles sémantiques (voir (Turian *et al.*, 2010 ; Collobert *et al.*, 2011) pour une vue d'ensemble).

De nombreux travaux récents s'intéressent à l'utilisation de modèles continus en traduction automatique et plusieurs propositions ont été formulées en ce sens. Une part importante est dédiée aux modèles n-grammes de traduction (Schwenk *et al.*, 2007 ; Le *et al.*, 2012 ; Devlin *et al.*, 2014). Néanmoins ces travaux ont en commun d'apprendre les modèles de manière à maximiser la vraisemblance mesurée sur les données d'apprentissage. Or ce critère est peu corrélé avec, d'une part, les métriques utilisées pour évaluer la traduction et, d'autre part, il ne tient pas compte de la manière dont les modèles neuronaux sont intégrés au sein des systèmes de TAS. De plus, utiliser cet estimateur oblige à manipuler des modèles probabilistes dûment normalisés, ce qui représente un coût computationnel prohibitif étant donné les espaces de réalisation considérés. Ces problèmes ont déjà été rencontrés dans un cadre de modélisation des langues (voir références *supra*) et des solutions ont été proposées, qui reposent soit sur le recours à des solutions qui permettent d'alléger ce coût, comme l'utilisation d'une couche de sortie structurée, soit d'optimiser des critères d'apprentissage alternatifs. Notons que les mêmes écueils sont rencontrés avec les approches les plus récentes qui proposent un système de traduction neuronal s'appuyant sur des modèles récurrents (Sutskever *et al.*, 2014 ; Bahdanau *et al.*, 2014). En particulier, la fonction optimisée par ces systèmes est également la vraisemblance des données d'apprentissage, ce qui entraîne qu'ils sont également confrontés au problème de la normalisation pour les très grands vocabulaires (Jean *et al.*, 2015).

La contribution principale de cet article est la proposition d'un cadre discriminant pour l'apprentissage des modèles continus de traduction, qui permet d'orienter l'optimisation du modèle vers les difficultés du système de TAS et donc d'apprendre à discriminer les hypothèses considérées en fonction de la métrique utilisée lors de l'évaluation. Cette approche est comparée à deux méthodes d'estimation compétitives et bien établies : le maximum de vraisemblance et l'estimation contrastive bruitée. Les résultats expérimentaux montrent des gains significatifs en termes de scores BLEU, et donc l'intérêt d'un tel cadre d'apprentissage pour la TAS.

Cet article est organisé de la manière suivante : la section 2 introduit les modèles continus de traduction qui seront utilisés dans ces travaux ; puis les différentes méthodes d'apprentissage étudiées sont décrites à la section 3, avec en particulier la méthode discriminante ; les résultats expérimentaux sont enfin présentés à la section 4. Cet article étend le travail décrit dans (Do *et al.*, 2015) de la manière suivante. Tout d'abord, le cadre d'évaluation est étendu avec l'inclusion d'un nouveau scénario, nommé *apprentissage*, où les données utilisées pour l'estimation du système de traduction servent également à estimer les modèles de traduction neuronaux. Ce nouveau scénario permet de mesurer l'impact du critère discriminant dans un contexte moins favorable. De plus, une étude comparative est menée sur les distributions de bruit utilisables pour l'estimation contrastive bruitée, opposant la solution habituellement préconisée (une distribution unigramme) à la distribution uniforme. Enfin pour les ex-

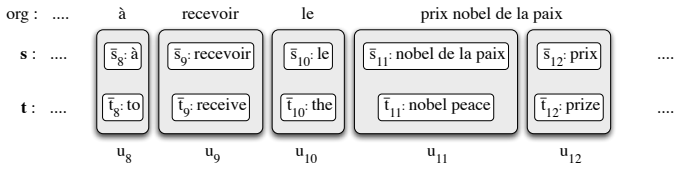

Figure 1. *Extrait d'une paire de phrases parallèles segmentées. La phrase source originale (org) est indiquée au-dessus de la phrase source réordonnée* $\mathbf{s}$ *et de la phrase cible* $\mathbf{t}$*. La paire de phrases* $(\mathbf{s}, \mathbf{t})$ *est décomposée en une séquence de* L *unités bilingues (n-uplets)* $u_1, ..., u_L$*. Chaque n-uplet* u_i *associe un segment source à un segment cible :* $\overline{s}_i$ *et* $\overline{t}_i$*.*

périences utilisant les données TED Talks, la configuration correspond désormais à celle utilisée lors de la campagne d'évaluation de 2014[2].

2. Modèles neuronaux pour la traduction automatique

Cette section propose une vue d'ensemble des modèles continus de traduction tels que nous allons les utiliser dans cet article. Si ce type de modèle s'intègre naturellement dans l'approche n-gramme en traduction automatique, il peut également être utilisé avec les approches usuelles à base de segments (Do *et al.*, 2014b). Pour plus de détails sur ces modèles et leur intégration dans les systèmes de TAS, le lecteur peut se reporter à (Le *et al.*, 2012 ; Schwenk, 2012).

2.1. *Approche n-gramme en traduction automatique*

L'approche n-gramme en traduction automatique est une variante de l'approche à base de segments (ou *phrase-based*) (Zens *et al.*, 2002 ; Koehn, 2010). Décrite dans (Casacuberta et Vidal, 2004) puis (Mariño *et al.*, 2006 ; Crego et Mariño, 2006), elle s'en distingue par une décomposition spécifique de la probabilité jointe d'une paire de phrases parallèles où l'on suppose que la phrase source a été réordonnée préalablement. Ainsi, notons $P(\mathbf{s}, \mathbf{t})$ cette probabilité jointe, où $\mathbf{s}$ est une phrase source de I mots $(s_1, ..., s_I)$ réordonnés, et $\mathbf{t}$ la phrase cible associée et composée de J mots cibles $(t_1, ..., t_J)$. Cette paire de phrases est décomposée en L unités bilingues appelées n-uplets, $(\mathbf{s}, \mathbf{t}) = (\mathrm{u}_1, ..., \mathrm{u}_L)$. Une illustration de cette décomposition est donnée figure 1.

2. `http://workshop2014.iwslt.org/`

Dans cette modélisation, les n-uplets sont les unités élémentaires de traduction[3], représentant une correspondance $\mathrm{u} = (\overline{s}, \overline{t})$ entre une séquence $\overline{s}$ de mots sources et une séquence de mots cibles $\overline{t}$. En utilisant l'hypothèse markovienne, la probabilité jointe peut être factorisée de la manière suivante :

$$P(\mathbf{s}, \mathbf{t}) = \prod_{i=1}^{L} P(\mathrm{u}_i | \mathrm{u}_{i-n+1}^{i-1}), \qquad [1]$$

où u_{i-n+1}^{i-1} représente la séquence de n-uplets $\mathrm{u}_{i-n+1}, ..., \mathrm{u}_{i-1}$. Le modèle complet d'une paire de phrases parallèles contient donc les variables latentes précisant d'une part le réordonnancement de la phrase source, ainsi que la segmentation en unités bilingues. Ces variables latentes définissent la dérivation de la phrase source qui engendre la phrase cible. Elles sont ignorées par la suite afin d'alléger les notations. Comme détaillé dans (Mariño *et al.*, 2006 ; Crego et Mariño, 2006), ces variables latentes sont inférées lors de la phase d'apprentissage à partir des données parallèles alignées automatiquement, et ce, en deux étapes : pour chaque paire de phrases parallèles, la phrase source est d'abord réordonnée de manière à suivre l'ordre des mots de la phrase cible, puis la segmentation en unités bilingues est effectuée.

Le modèle de traduction ainsi défini est un modèle de séquences utilisant l'hypothèse de n-gramme. La différence avec les modèles de langue monolingues est que les unités manipulées ne sont plus les mots mais les n-uplets. L'espace de réalisation considéré est alors bien plus grand qu'un inventaire monolingue de mots, alors que les données d'apprentissage disponibles se réduisent aux données parallèles. Ainsi, le caractère parcimonieux des données textuelles en général et des données parallèles en particulier rend difficile une estimation directe de ce type de modèle. Une solution est de décomposer les n-uplets en unités plus petites, comme, par exemple, en distinguant la partie source de la partie cible. L'équation [1] peut ainsi être décomposée de deux manières différentes :

$$\begin{aligned} & P(\mathrm{u}_i | \mathrm{u}_{i-n+1}^{i-1}) \\ & \quad = P(\overline{t}_i | \overline{s}_{i-n+1}^{i}, \overline{t}_{i-n+1}^{i-1}) P(\overline{s}_i | \overline{s}_{i-n+1}^{i-1}, \overline{t}_{i-n+1}^{i-1}) \\ & \quad = P(\overline{s}_i | \overline{t}_{i-n+1}^{i}, \overline{t}_{i-n+1}^{i-1}) P(\overline{t}_i | \overline{s}_{i-n+1}^{i-1}, \overline{t}_{i-n+1}^{i-1}), \end{aligned} \qquad [2]$$

où un n-uplet u_i se décompose en deux parties source et cible, respectivement $\overline{s}_i$ et $\overline{t}_i$. Considérons, par exemple, la première décomposition dont une illustration est donnée à la figure 2. Elle fait apparaître deux termes : le premier $P(\overline{t}_i | \overline{s}_{i-n+1}^{i}, \overline{t}_{i-n+1}^{i-1})$ peut s'apparenter à un modèle de traduction, alors que le second $P(\overline{s}_i | \overline{s}_{i-n+1}^{i-1}, \overline{t}_{i-n+1}^{i-1})$ s'intéresse au réordonnancement de la phrase source conditionnellement à l'historique de traduction $\overline{t}_{i-n+1}^{i-1}$.

3. Les n-uplets sont assimilables aux paires de segments ou bisegments (*phrase pairs*) utilisés dans l'approche plus classique à base de segments.

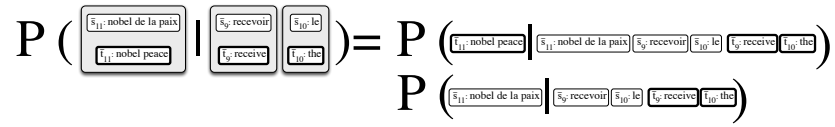

Figure 2. *Exemple de décomposition en segment source et cible d'une paire de phrases parallèles sous l'hypothèse* 3*-gramme. Reprenant l'exemple de la figure 1, il s'agit de prédire* u_{11} *connaissant* u_9 *et* u_{10}.

Désormais, deux espaces de réalisation sont impliqués, un par langue, qui recensent l'ensemble des segments. Il est encore possible de réduire ces espaces de réalisation en décomposant les segments en séquences de mots. Le modèle obtenu considère alors deux séquences de mots, l'une cible et l'autre source, synchronisées sur la segmentation en unités de traduction et dont la partie source a été réordonnée au préalable. L'hypothèse n-gramme porte désormais sur les mots et non sur les segments[4]. Cela correspond à un modèle n-gramme bilingue de mots tel qu'il est initialement décrit dans (Le *et al.*, 2012) et étendu dans (Devlin *et al.*, 2014).

2.2. *Architectures neuronales pour les modèles de traduction*

L'estimation des distributions n-grammes peut être réalisée par des réseaux de neurones multicouches, comme proposé dans (Bengio *et al.*, 2003 ; Schwenk, 2007) pour une application monolingue. Une architecture couramment utilisée est l'architecture *feed-forward*, illustrée sur la figure 3. Nous en résumons ici l'idée principale, les détails peuvent être trouvés dans (Le *et al.*, 2012) : les mots du contexte sont d'abord projetés dans un espace de représentation continue, chaque langue ayant sa matrice de projection (R_s et R_t respectivement pour les langues source et cible) ; leur concaténation permet d'obtenir une représentation continue du contexte bilingue ; puis une transformation non linéaire est appliquée afin de prédire le mot cible grâce à la couche de sortie.

Avec ce type d'architecture, la taille du vocabulaire de sortie est la principale limitation en termes de temps de calcul[5]. Des solutions ont été proposées concernant spécifiquement la couche de sortie afin de réduire le coût d'inférence et d'apprentissage tout en permettant l'usage d'un vocabulaire de taille réaliste. La première consiste à structurer la couche de sortie comme proposé par Mnih et Hinton (2008). Dans cet

4. Néanmoins, dans le cas des modèles neuronaux, la taille de l'historique n'implique pas, comme pour les modèles n-grammes discrets, une augmentation exponentielle du nombre de paramètres. Il est donc possible d'envisager des contextes plus larges. Dans cet article nous utiliserons des 10-grammes.

5. La majeure partie du coût computationnel se situe en effet au niveau de la couche de sortie, où il est nécessaire de normaliser la distribution en effectuant la somme sur l'ensemble du vocabulaire.

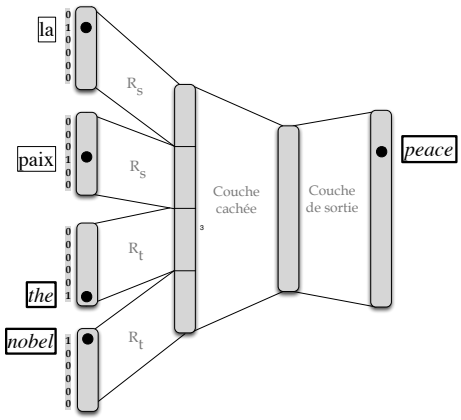

Figure 3. *Architecture neuronale pour l'estimation des distributions n-grammes bilingues (ici $n = 3$). Cette figure illustre l'estimation de la distribution $P(\overline{t}_i|\overline{s}^i_{i-n+1}, \overline{t}^{i-1}_{i-n+1})$ en poursuivant l'exemple de la figure 1.*

article nous utilisons la structure SOUL décrite dans (Le *et al.*, 2011 ; Le *et al.*, 2013). Le modèle n-gramme peut être alors considéré comme un modèle neuronal de classes hiérarchiques de mots. Le vocabulaire de sortie est représenté par un arbre, pour lequel chaque mot est une feuille. Dans ce cas, la simple couche de sortie représentée à la figure 3 est remplacée par une couche de sortie composée de plusieurs couches, une par nœud de l'arbre. Chacune de ces couches a une fonction d'activation de type *softmax*, permettant l'estimation de la distribution au sein de la classe qui lui est associée. La figure 4 représente la couche de sortie structurée d'un tel modèle.

Une autre solution propose un cadre d'apprentissage pour des modèles non normalisés, permettant de garder une couche de sortie de forme conventionnelle. Cette approche nommée « estimation contrastive bruitée » (ou NCE pour *noise contrastive estimation*) a été introduite par Gutmann et Hyvärinen (2010), puis appliquée aux modèles de langue par Mnih et Teh (2012), et enfin intégrée à un système de traduction dans (Vaswani *et al.*, 2013). Cette approche sera détaillée à la section 3.1.

Dans les deux cas, le modèle neuronal attribue un score positif à un mot w dans son contexte $\mathbf{c}$ noté $\mathbf{b}_{\boldsymbol{\theta}}(w, \mathbf{c})$, où $\boldsymbol{\theta}$ représente l'ensemble des paramètres du modèle à estimer. Concrètement, ce score positif est l'exponentielle de l'activation de la dernière couche linéaire du modèle, $\mathbf{b}_{\boldsymbol{\theta}}(w, \mathbf{c}) = \exp(\mathbf{a}_{\boldsymbol{\theta}}(w, \mathbf{c}))$. Les scores $\mathbf{b}_{\boldsymbol{\theta}}(.)$ peuvent ensuite être normalisés de manière efficace dans le cas du modèle SOUL, ou utilisés tels quels dans le cas d'un modèle NCE.

Des travaux récents (Sutskever *et al.*, 2014 ; Bahdanau *et al.*, 2014) ont introduit une architecture entièrement neuronale pour la TAS. Cette architecture utilise des réseaux récurrents pour, dans un premier temps, représenter la phrase source dans un espace continu, puis, dans un second temps pour engendrer la phrase cible. Une

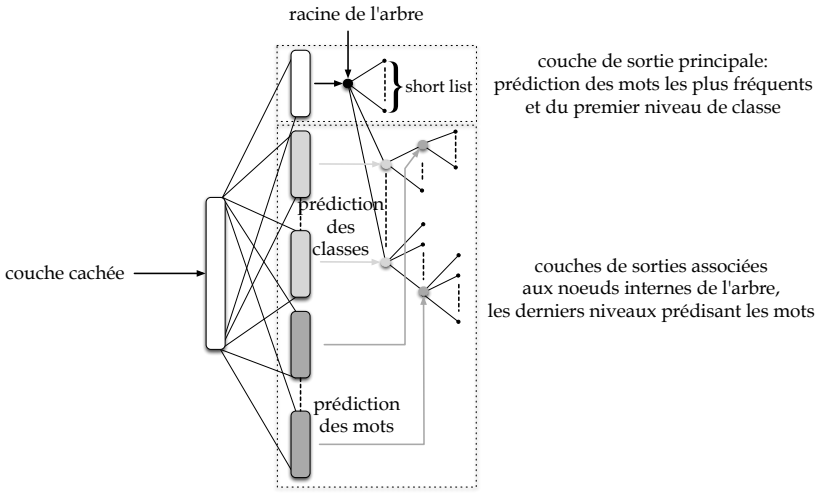

Figure 4. *La couche structurée en arbre du modèle SOUL permettant le calcul efficace de la probabilité d'un mot dans son contexte.*

des forces de ce type d'approches est de pouvoir se dispenser des alignements de mots, et de la définition préalable des unités de traduction (voir toutefois (Bahdanau *et al.*, 2014) qui réintroduit une forme d'alignement à travers *les modèles d'attention*). Néanmoins, ces modèles sont optimisés de manière à maximiser la vraisemblance mesurée sur les données d'apprentissage. Une extension intéressante serait de proposer un cadre discriminant d'apprentissage pour ce type d'approches afin de les rendre sensibles aux métriques d'évaluation utilisées en TAS.

3. Estimer des modèles de traduction

Les modèles de traduction neuronaux décrits dans la section 2 sont habituellement appris en maximisant la log-vraisemblance, ou plus récemment en utilisant l'estimation contrastive bruitée. Or ces critères d'apprentissage n'ont qu'un lien lointain avec, d'une part, leur utilisation usuelle en traduction automatique en interaction avec les autres modèles et, d'autre part, avec les métriques d'évaluation. En effet, à cause du coût computationnel qu'ils impliquent, les modèles neuronaux sont le plus souvent utilisés en post-traitement d'un système de traduction conventionnel. Leur rôle est alors d'aider le système à trier un ensemble d'hypothèses, les N-meilleures, en se fondant sur une mesure automatique de qualité de la traduction, la plupart du temps le score BLEU (Papineni *et al.*, 2002).

Dans cette section, nous commençons par décrire les deux critères d'apprentissage habituels des modèles de traduction neuronaux, puis nous formalisons (§ 3.2) un algorithme d'apprentissage discriminant visant à estimer directement les paramètres du

modèle de traduction, de manière à optimiser l'étape de réévaluation des N-meilleures hypothèses. Cette méthode s'appuie sur la définition d'une fonction objectif que nous présentons dans un troisième temps (§ 3.3).

3.1. *Maximum de vraisemblance et estimation contrastive bruitée*

Traditionnellement, les modèles de traduction neuronaux sont entraînés de manière à maximiser la vraisemblance. En pratique, les données d'apprentissage sont présentées comme un ensemble de n-grammes $\mathcal{S}_n$, et la fonction objectif à minimiser[6] est la suivante :

$$\mathcal{L}_{cll}(\boldsymbol{\theta}, \mathcal{S}_n) = \sum_{(w,\mathbf{c})\in\mathcal{S}_n} -\log \mathbf{p}_{\boldsymbol{\theta}}(w|\mathbf{c}) + \mathcal{R}(\boldsymbol{\theta}), \qquad [3]$$

où $\mathcal{R}(\boldsymbol{\theta})$ est le terme de régularisation L_2 défini par $\mathcal{R}(\boldsymbol{\theta}) = \gamma \times \frac{||\boldsymbol{\theta}||^2}{2}$ et γ est l'hyperparamètre associé. Ce critère $\mathcal{L}_{cll}(\boldsymbol{\theta}, \mathcal{S}_n)$ correspond en fait à la somme négative des log-probabilités conditionnelles des n-grammes contenus dans les données d'apprentissage. Pour calculer cette fonction objectif, il est nécessaire de normaliser la sortie du réseau de neurones sur tous les mots du vocabulaire $\mathcal{V}$, selon :

$$\mathbf{p}_{\boldsymbol{\theta}}(w|\mathbf{c}) = \frac{\mathbf{b}_{\boldsymbol{\theta}}(w, \mathbf{c})}{\sum_{w'\in\mathcal{V}} \mathbf{b}_{\boldsymbol{\theta}}(w', \mathbf{c})},$$

où $\mathbf{b}_{\boldsymbol{\theta}}(w, \mathbf{c}) = \exp(\mathbf{a}_{\boldsymbol{\theta}}(w, \mathbf{c}))$ et $\mathbf{a}_{\boldsymbol{\theta}}(w, \mathbf{c})$ désigne l'activité du neurone de sortie associé au mot w. La minimisation de $\mathcal{L}_{cll}(\boldsymbol{\theta}, \mathcal{S}_n)$ se fait par descente de gradient stochastique. Néanmoins, le coût de la normalisation peut être prohibitif pour les tailles de vocabulaires typiquement utilisées en traduction automatique, qui contiennent des dizaines, voire des centaines de milliers d'entrées. Dans cet article, nous utilisons le modèle SOUL proposé par Le *et al.* (2011) qui, grâce à une couche de sortie structurée en arbre, permet de ramener le temps de calcul à des niveaux raisonnables.

Une approche différente permet d'éviter le calcul induit par la normalisation : l'estimation contrastive bruitée ou *Noise Contrastive Estimation* (Gutmann et Hyvärinen, 2010). L'idée principale est de reformuler le problème comme une tâche de classification binaire entre, d'une part, les exemples positifs rencontrés dans les données d'apprentissage et, d'autre part, les exemples négatifs engendrés artificiellement selon une distribution de bruit $\mathbf{p}_N(.)$. Notons $\mathcal{X}^w$ la variable aléatoire binaire indiquant si le mot w est un exemple positif ou négatif. Nous faisons de plus l'hypothèse (justifiée ci-dessous) que les échantillons négatifs sont *a priori* M fois plus fréquents que les positifs, alors les probabilités *a priori* des deux événements sont données par :

$$\mathbf{p}(\mathcal{X}^{w'} = 1) = \frac{1}{M+1}; \mathbf{p}(\mathcal{X}^{w'} = 0) = \frac{M}{M+1}.$$

6. Dans la suite de cet article nous adoptons la convention habituelle en apprentissage automatique qui consiste à formuler l'apprentissage comme la minimisation d'une fonction objectif.

En supposant de plus que $\mathbf{p}_{\boldsymbol{\theta}}$ est une bonne approximation de la distribution empirique des exemples positifs, il est possible d'écrire :

$$\mathbf{p}(w'|\mathbf{c}, \mathcal{X}^{w'} = 1) = \mathbf{p}_{\boldsymbol{\theta}}(w'|\mathbf{c})$$
$$\mathbf{p}(w'|\mathbf{c}, \mathcal{X}^{w'} = 0) = \mathbf{p}_N(w'),$$

puis de déduire, en appliquant le théorème de Bayes, les probabilités *a posteriori* suivantes :

$$\begin{aligned}\mathbf{p}(\mathcal{X}^{w'} = 1|w', \mathbf{c}) &= \frac{\mathbf{p}_{\boldsymbol{\theta}}(w'|\mathbf{c})}{\mathbf{p}_{\boldsymbol{\theta}}(w'|\mathbf{c}) + M\mathbf{p}_N(w')} \\ \mathbf{p}(\mathcal{X}^{w'} = 0|w', \mathbf{c}) &= \frac{M\mathbf{p}_N(w')}{\mathbf{p}_{\boldsymbol{\theta}}(w'|\mathbf{c}) + M\mathbf{p}_N(w')}.\end{aligned} \quad [4]$$

La fonction à minimiser devient alors l'espérance de $-\log(\mathbf{p}(\mathcal{X}^{w'}|w', \mathbf{c}))$ sur l'ensemble d'exemples constitué d'un unique exemple positif (w), auxquels sont associés M exemples négatifs $\{w_1^*, \ldots, w_M^*\}$. Elle s'écrit alors de la manière suivante :

$$\begin{aligned}\mathcal{L}_{nce}(\boldsymbol{\theta}, \mathcal{S}_n) = \sum_{(w,\mathbf{c})\in\mathcal{S}_n} \Big[&- \log \frac{\mathbf{p}_{\boldsymbol{\theta}}(w|\mathbf{c})}{\mathbf{p}_{\boldsymbol{\theta}}(w|\mathbf{c}) + M\mathbf{p}_N(w)} \\ &- \sum_{i=1}^{M} \log \frac{M\mathbf{p}_N(w_i^*)}{\mathbf{p}_{\boldsymbol{\theta}}(w_i^*|\mathbf{c}) + M\mathbf{p}_N(w_i^*)} \Big] + \mathcal{R}(\boldsymbol{\theta}),\end{aligned} \quad [5]$$

où $(w, \mathbf{c})$ est un n-gramme issu des données d'apprentissage et $(w_i^*)_{i=1}^M$ l'ensemble des M exemples négatifs qui lui sont associés. Ces exemples négatifs sont tirés aléatoirement de la distribution de bruit. Dans (Gutmann et Hyvärinen, 2010 ; Mnih et Teh, 2012), les auteurs insistent sur l'importance du choix de cette distribution de bruit. Il semble en effet nécessaire qu'elle soit proche de la distribution empirique, tout en permettant un échantillonnage efficace. Ainsi, le choix le plus répandu est d'utiliser la distribution unigramme sur les données d'apprentissage.

Notons que, dans l'équation [5], le terme $\mathbf{p}_{\boldsymbol{\theta}}(w|\mathbf{c})$ apparaît au numérateur et au dénominateur (ou dénominateur seulement). Il est possible de se débarrasser du terme de normalisation, et donc de remplacer (avantageusement) $\mathbf{p}_{\boldsymbol{\theta}}(w|\mathbf{c})$ par $\mathbf{b}_{\boldsymbol{\theta}}(w, \mathbf{c})$, permettant d'éviter de calculer une somme coûteuse. De plus, il est possible de montrer que lorsque M tend vers l'infini, cette fonction objectif tend vers $\mathcal{L}_{cll}$. Ainsi l'estimation contrastive bruitée est une méthode d'optimisation formalisant le calcul approché de la constante de normalisation de la manière suivante : pour chaque n-gramme observé sur les données d'apprentissage, M mots (exemples négatifs) sont échantillonnés à partir de la distribution de bruit ; ainsi, au lieu d'effectuer la somme sur l'ensemble du vocabulaire, seuls M mots sont considérés.

3.2. *Méthode discriminante d'apprentissage pour la traduction automatique*

Malgré les progrès récents, l'inférence avec un réseau de neurones reste trop coûteuse pour que ce type de modèle puisse être intégré au décodage aussi facilement que les modèles de langue discrets utilisés dans les systèmes de traduction automatique[7]. L'usage est donc d'utiliser ces modèles lors d'une seconde étape de réévaluation des N-meilleures hypothèses.

Afin de définir ce cadre, supposons que pour chaque phrase source $\mathbf{s}$ à traduire, le décodeur génère une liste des N meilleures hypothèses $\{\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_N\}$. Chaque hypothèse $\mathbf{h}_i$ est constituée d'une phrase cible $\mathbf{t}_i$ et de la dérivation $\mathbf{a}_i$ permettant sa construction[8]. Elle est évaluée par le système de traduction grâce à la fonction suivante :

$$F_{\boldsymbol{\lambda}}(\mathbf{s}, \mathbf{h}) = \sum_{k=1}^{K} \lambda_k f_k(\mathbf{s}, \mathbf{h}), \qquad [6]$$

où K fonctions caractéristiques (f_k) sont pondérées par un jeu de poids $\boldsymbol{\lambda}$. Les fonctions caractéristiques utilisées dans cet article sont les mêmes que celles trouvées dans les systèmes usuels à base de segments (Crego *et al.*, 2011).

L'introduction d'un modèle continu lors de l'étape de réévaluation des hypothèses se traduit par l'ajout à $F_{\boldsymbol{\lambda}}(.)$ d'une fonction caractéristique supplémentaire $f_{\boldsymbol{\theta}}(\mathbf{s}, \mathbf{h})$, qui varie selon le modèle utilisé :

$$f_{\boldsymbol{\theta}}(\mathbf{s}, \mathbf{h}) = \begin{cases} \log \mathbf{p}_{\boldsymbol{\theta}}(\mathbf{s}, \mathbf{h}) = \sum\limits_{(w,\mathbf{c}) \in (\mathbf{s},\mathbf{h})} \log \mathbf{p}_{\boldsymbol{\theta}}(w|\mathbf{c}) & \text{(modèle SOUL),} \\ \sum\limits_{(w,\mathbf{c}) \in (\mathbf{s},\mathbf{h})} \log \mathbf{b}_{\boldsymbol{\theta}}(w, \mathbf{c}) = \sum\limits_{(w,\mathbf{c}) \in (\mathbf{s},\mathbf{h})} \mathbf{a}_{\boldsymbol{\theta}}(w, \mathbf{c}) & \text{(modèle NCE).} \end{cases} \qquad [7]$$

Dans les deux cas, il est nécessaire de prendre la somme sur tous les n-grammes extraits de la dérivation considérée. Comme précédemment, $\boldsymbol{\theta}$ désigne le vecteur de paramètres définissant le modèle continu de traduction. Ainsi, la fonction d'évaluation des hypothèses devient :

$$G_{\boldsymbol{\lambda},\boldsymbol{\theta}}(\mathbf{s}, \mathbf{h}) = F_{\boldsymbol{\lambda}}(\mathbf{s}, \mathbf{h}) + \lambda_{K+1} f_{\boldsymbol{\theta}}(\mathbf{s}, \mathbf{h}). \qquad [8]$$

Cette fonction dépend à la fois des paramètres $\boldsymbol{\theta}$ du modèle continu de traduction et des paramètres $\boldsymbol{\lambda}$ de la fonction d'évaluation. Dans l'approche que nous proposons,

7. Notons néanmoins les tentatives récentes d'intégration des modèles de traduction neuronaux dans le décodeur (Niehues et Waibel, 2012 ; Vaswani *et al.*, 2013 ; Devlin *et al.*, 2014 ; Alkhouli *et al.*, 2015).

8. $\mathbf{a}_i$ regroupe l'ensemble des variables latentes du processus de traduction. Dans le cas d'un système de traduction n-gramme, il s'agit du réordonnancement de la phrase source et du choix de la segmentation (cf. § 2.1).

Algorithme 1 Procédure d'optimisation jointe de $\boldsymbol{\theta}$ et $\boldsymbol{\lambda}$

```
Init. de θ et λ
Pour chaque itération faire
    Pour P paquets faire                                   ▷ λ fixé
        Calcul du sous-gradient de L(θ) pour chaque phrase s du paquet
        Mise à jour de θ
    Fin Pour
    Mise à jour de λ en utilisant le dév.                  ▷ θ fixé
Fin Pour
```

l'optimisation nécessite *d'alterner l'estimation des poids de mélange $\boldsymbol{\lambda}$ et l'apprentissage des paramètres $\boldsymbol{\theta}$ du modèle continu* : la première étape utilise classiquement les données de développement alors que la deuxième utilise les données parallèles d'apprentissage.

Cette procédure d'optimisation est décrite par l'algorithme 1. Les données d'apprentissage sont découpées en paquets de 128 phrases successives. Chacun de ces paquets sert à la mise à jour de $\boldsymbol{\theta}$ à $\boldsymbol{\lambda}$ constant et ces derniers sont réestimés tous les P paquets. Notons que cet algorithme nécessite la définition d'une fonction objectif $\mathcal{L}(\boldsymbol{\theta})$ pour le modèle continu de traduction, qui sera décrite à la section 3.3. Dans cet article, l'optimisation de $\boldsymbol{\lambda}$ utilise les outils standard, en l'occurrence l'algorithme *K-Best Mira* décrit dans (Cherry et Foster, 2012) et tel qu'il est implémenté dans la boîte à outils MOSES[9].

3.3. *Une fonction objectif discriminante*

Le critère d'apprentissage discriminant proposé dans cet article s'inspire à la fois des méthodes à vaste marge et des approches de *ranking*. Comme expliqué précédemment, chaque hypothèse de traduction $\mathbf{h}_i$ engendrée par le système de traduction est évaluée selon l'équation [8]. Mais sa qualité peut également être évaluée selon un critère de qualité de traduction, ici le score BLEU, ou plus précisément selon une approximation du score BLEU au niveau de la phrase, que l'on note $sBLEU(\mathbf{h}_i)$[10]. Si $\mathbf{h}^*$ désigne l'hypothèse ayant le meilleur score, il est possible de définir un critère visant à maximiser la marge (Freund et Schapire, 1999 ; McDonald *et al.*, 2005 ; Watanabe *et al.*, 2007) de la manière suivante :

$$\mathcal{L}_{mm}(\boldsymbol{\theta}, \mathbf{s}) = -G_{\boldsymbol{\lambda},\boldsymbol{\theta}}(\mathbf{s}, \mathbf{h}^*) + \max_{1 \leq j \leq N} \left(G_{\boldsymbol{\lambda},\boldsymbol{\theta}}(\mathbf{s}, \mathbf{h}_j) + \text{cost}_\alpha(\mathbf{h}_j) \right), \qquad [9]$$

9. `http://www.statmt.org/moses/`

10. Dans cet article, tous les résultats expérimentaux utilisent le score BLEU comme mesure d'évaluation, ce qui justifie l'utilisation du $sBLEU$ dans la fonction objectif discriminante. Il serait néanmoins possible d'utiliser une autre métrique comme le *Translation Edit Rate* à la fois dans le cadre discriminant et comme mesure d'évaluation finale.

où $\text{cost}_\alpha(\mathbf{h}_j) = \alpha\big(sBLEU(\mathbf{h}^*) - sBLEU(\mathbf{h}_j)\big)$ représente la fonction de coût et le paramètre α pondère sa contribution. Lorsque $\alpha = 0$, nous retrouvons la fonction objectif du perceptron structuré (Collins, 2002). Ce critère introduit une marge entre $\mathbf{h}^*$ et les autres hypothèses. Néanmoins, parmi les autres hypothèses, des traductions pourraient être acceptables et pourraient être considérées autrement que mauvaises. Ainsi, une alternative est de s'inspirer du classement par couple (ou *pairwise ranking*) comme le propose le système PRO de Hopkins et May (2011). Supposons que r_i désigne le rang de l'hypothèse $\mathbf{h}_i$ lorsque la liste des hypothèses est triée avec comme critère $sBLEU$, le système PRO définit la fonction objectif suivante :

$$\mathcal{L}_{pro}(\boldsymbol{\theta}, \mathbf{s}) = \sum_{1 \leq i,k \leq N} \mathbb{I}_{\{r_i + \delta \leq r_k, G_{\boldsymbol{\lambda},\boldsymbol{\theta}}(\mathbf{s},\mathbf{h}_i) < G_{\boldsymbol{\lambda},\boldsymbol{\theta}}(\mathbf{s},\mathbf{h}_k)\}} \left(-G_{\boldsymbol{\lambda},\boldsymbol{\theta}}(\mathbf{s}, \mathbf{h}_i) + G_{\boldsymbol{\lambda},\boldsymbol{\theta}}(\mathbf{s}, \mathbf{h}_k)\right). \quad [10]$$

Notons que cette fonction objectif ne considère qu'un sous-ensemble de $N(N-1)/2$ couples d'hypothèses. En effet, un couple d'hypothèses n'est pris en compte que si la différence absolue des rangs excède un seuil prédéfini δ.

Le critère que nous proposons pour optimiser les modèles de traduction neuronaux est une combinaison des deux critères précédents. Ce choix s'appuie sur les résultats expérimentaux de Do *et al.* (2014a), qui a introduit ces critères dans le cadre de l'adaptation de modèles. Pour un couple d'hypothèses $(\mathbf{h}_i, \mathbf{h}_k)$ tel que $r_i + \delta < r_k$, l'objectif est que la différence de score $G_{\boldsymbol{\lambda},\boldsymbol{\theta}}(\mathbf{s}, \mathbf{h}_i) - G_{\boldsymbol{\lambda},\boldsymbol{\theta}}(\mathbf{s}, \mathbf{h}_k)$ soit au-delà d'une certaine marge. Comme précédemment, la marge s'exprime grâce à l'approximation du score BLEU au niveau de la phrase et donc *via* la fonction de coût cost_α. Nous pouvons alors définir le sous-ensemble des hypothèses *critiques* comme :

$$\mathcal{C}_\delta^\alpha = \{(i,k) : 1 \leq i,k \leq N, r_i + \delta \leq r_k, \quad [11]$$
$$G_{\boldsymbol{\lambda},\boldsymbol{\theta}}(\mathbf{s}, \mathbf{h}_i) - G_{\boldsymbol{\lambda},\boldsymbol{\theta}}(\mathbf{s}, \mathbf{h}_k) < \text{cost}_\alpha(\mathbf{h}_k) - \text{cost}_\alpha(\mathbf{h}_i)\}.$$

La fonction objectif que nous allons utiliser se définit de la manière suivante :

$$\mathcal{L}_{pro-mm}(\boldsymbol{\theta}, \mathbf{s}) = \sum_{(i,k) \in \mathcal{C}_\delta^\alpha} \text{cost}_\alpha(\mathbf{h}_k) - \text{cost}_\alpha(\mathbf{h}_i) - G_{\boldsymbol{\lambda},\boldsymbol{\theta}}(\mathbf{s}, \mathbf{h}_i) + G_{\boldsymbol{\lambda},\boldsymbol{\theta}}(\mathbf{s}, \mathbf{h}_k). \quad [12]$$

Cette fonction objectif ne requiert pas, tout comme le NCE, que le score du modèle neuronal $f_{\boldsymbol{\theta}}(\mathbf{s}, \mathbf{h})$ inclus dans $G_{\boldsymbol{\lambda},\boldsymbol{\theta}}(\mathbf{s}, \mathbf{h}_i)$ soit normalisé. Il est donc possible d'apprendre un modèle de type SOUL selon ce critère mais également un modèle non normalisé de type NCE. Remarquons enfin que si $\alpha = 0$, cette fonction se ramène à celle de PRO (équation [10]).

4. Expériences

L'objectif principal de cette partie expérimentale est d'évaluer l'impact des méthodes d'apprentissage présentées à la section 3 sur les performances d'un système

Domaine	Scénario	Données d'apprentissage		Relation entre (1) & (2)
		Système de TAS (1)	Modèle neuronal (2)	
TED Talks	Apprent.	TED Talks / 180K	TED Talks / 180K	(2) = (1)
	Adapt.	WMT'2014 / 12M	TED Talks / 180K	$(2) \cap (1) = \emptyset$ $\mid (2) \mid \ll \mid (1) \mid$
Médical	Apprent. partiel	Textes médicaux P-A inclus/ $4,4$M	P-A / 200K	$(2) \subset (1)$ $\mid (2) \mid \ll \mid (1) \mid$
	« Adapt. »	Textes médicaux P-A exclus/ $4,2$M	P-A / 200K	$(2) \cap (1) = \emptyset$ $\mid (2) \mid \ll \mid (1) \mid$

Tableau 1. *Tableau récapitulatif des expériences. Deux domaines sont abordés, la traduction de l'anglais vers le français des TED Talks d'une part, et de textes médicaux, d'autre part. Pour chaque domaine, deux scénarios sont présentés : l'apprentissage (Apprent.) et l'adaptation (Adapt.). Ils correspondent à un usage distinct des données d'apprentissage nécessaires au système de TAS et au modèle de traduction neuronal.*

de traduction automatique qui intègre un modèle de traduction neuronal. Plus précisément, nous comparons le critère conventionnel d'apprentissage (le maximum de vraisemblance) avec, d'une part, l'estimation contrastive bruitée (NCE) et, d'autre part, avec notre critère d'apprentissage discriminant. Pour ce faire, une série d'expériences de traduction automatique depuis l'anglais vers le français est menée selon deux scénarios d'utilisation ainsi que deux domaines : la traduction des séminaires TED Talks et de textes médicaux.

4.1. *Données et tâches*

Le premier scénario d'utilisation, dénommé *apprentissage*, consiste à utiliser les mêmes données parallèles pour apprendre les paramètres du système de TAS et le modèle de traduction neuronal. Le second relève de l'*adaptation* au domaine. Il correspond au cas où un vaste corpus hors domaine est utilisé pour estimer les paramètres du système de TAS, alors que l'on dispose également d'un corpus parallèle du domaine mais nettement plus petit. Dans ce scénario, le corpus du domaine est uniquement utilisé pour apprendre le modèle de traduction neuronal, devenant ainsi le support de l'adaptation au domaine. L'intérêt de scénario est de permettre une adaptation au domaine sans avoir à réentraîner le système de TAS complet pour prendre en compte les nouvelles données du domaine. Le but recherché est donc d'obtenir une adaptation du système à un coût computationnel réduit. Enfin, une situation intermédiaire est également envisagée, *l'apprentissage partiel*, où seule une sous-partie des données d'apprentissage du système de TAS sert à estimer le modèle de traduction neuronal.

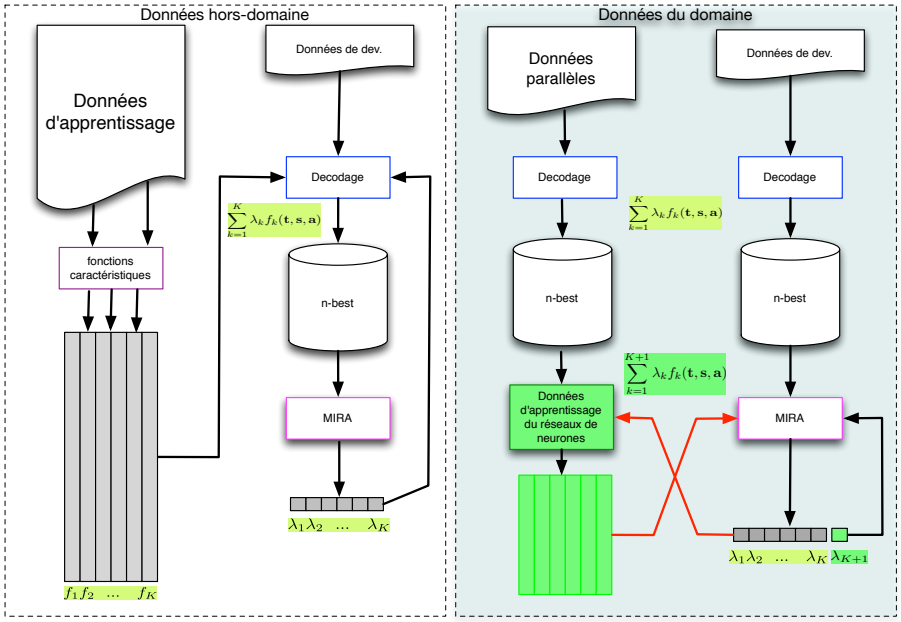

Figure 5. *Représentation du protocole expérimental*

La figure 5 représente ainsi le protocole expérimental. La partie de gauche correspond au processus habituel de construction d'un système de TAS : les fonctions caractéristiques standard (f_k) d'un système de TAS sont estimées à partir des données d'apprentissage, puis les coefficients d'interpolation $\boldsymbol{\lambda}$ sont optimisés en utilisant les données de développement, définissant ainsi la fonction $F_{\boldsymbol{\lambda}}(\mathbf{s}, \mathbf{h})$ (équation 6). Dans nos expériences, ce système est appris à partir de données partiellement ou totalement hors domaine. Puis, un modèle de traduction neuronal est intégré à ce système et la partie de droite représente l'approche discriminante d'optimisation proposée à la section 3.2. Les paramètres de ce modèle sont estimés à partir de la liste des N meilleures hypothèses engendrée par le système de TAS. L'algorithme 1 est utilisé pour conjointement estimer ces paramètres, ainsi que réévaluer les paramètres $\boldsymbol{\lambda}$ du système de TAS. Pour le système n-code sans modèle neuronal, les paramètres ont réestimé dans le cadre du scénario d'adaptation sur les données de développement du domaine.

Les expériences utilisent deux tâches de traduction automatique de l'anglais vers le français qui correspondent à deux domaines différents : la traduction de séminaires TED Talks et de textes médicaux. Concernant la traduction des séminaires TED Talks (Federico *et al.*, 2012), nous avons repris la tâche définie par la campagne d'évaluation internationale sur la traduction de la parole organisée dans le cadre des

ateliers IWSLT[11]. La tâche considérée est la traduction des séminaires dans leur version transcrite manuellement. Les données du domaine contiennent 180 k phrases parallèles ; des données hors domaine sont également disponibles en très grande quantité puisqu'elles correspondent aux données parallèles utilisées par la tâche de traduction de textes d'actualité liée à l'évaluation WMT[12], soit un peu plus de 12 M de paires de phrases.

Concernant le second domaine, la traduction de textes médicaux, le cadre utilisé est celui proposé lors de l'évaluation WMT'14[13] : les données hors domaine sont les mêmes que pour la tâche TED Talks, et que les données dédiées sont de nature très hétérogène et disponibles en grande quantité ($4,4$ M). De ces textes médicaux, nous avons retenu le corpus *Patent-Abstract* (P-A) qui contient 200 k phrases parallèles pour estimer le modèle de traduction neuronal. Le tableau 1 résume l'ensemble des informations concernant les données utilisées, selon le scénario et le domaine visés.

4.2. *Cadre expérimental*

Le système de traduction utilise n-code, une implémentation libre de l'approche n-gramme[14] et ses modèles ont été appris à partir d'une vaste quantité de données bilingues et monolingues dans le cadre de la campagne d'évaluation WMT. Le système est décrit plus précisément dans (Allauzen *et al.*, 2013).

Les modèles de traduction neuronaux sont appris uniquement sur les données du domaine. Chaque modèle, SOUL et NCE, est initialisé à partir de modèles neuronaux monolingues, estimés respectivement sur les parties source et cible du corpus bilingue. Tous les modèles n-grammes continus sont des 10-grammes et sont composés d'un espace de projection pour les mots (de dimension 500) ainsi que de deux couches cachées (respectivement de tailles $1\ 000$ et 500). Pour l'apprentissage discriminant, le système de traduction est d'abord utilisé pour engendrer une liste des 300 meilleures hypothèses pour chaque phrase source. Le seuil δ (cf. l'équation [11]) a été empiriquement fixé à 250 en fonction des scores BLEU sur les données de développement. Ces scores servent aussi pour choisir la meilleure itération qui correspond au modèle qui est ensuite évalué sur les données de test. Le critère d'évaluation de la traduction est le score BLEU (Papineni *et al.*, 2002).

Comme décrit à la section 2.1, il existe deux manières de décomposer la probabilité jointe d'une paire de phrases (voir l'équation [2]), et il est donc possible de définir quatre modèles de traduction neuronaux. Par souci de clarté, seul le modèle $P(\overline{t}_i|\overline{s}^i_{i-n+1}, \overline{t}^{i-1}_{i-n+1})$ est utilisé par la suite. Néanmoins, des tendances similaires ont été observées avec les autres modèles.

11. International Workshop on Spoken Language Translation : `http://workshop2014.iwslt.org/`
12. Workshop on Machine Translation : `http://statmt.org`.
13. `http://www.statmt.org/wmt14/medical-task/`
14. `https://ncode.limsi.fr`

	dév.	test	apprentissage
Apprentissage (TED 2014)			
MOSES (TED Talks)	27,3	31,7	
n-code (TED Talks)	28,1	32,3	**65,6**
+ MTC SOUL	29,1	33,3	
+ MTC NCE (uniforme)	28,7	32,8	
+ MTC NCE (unigramme)	28,9	33,1	64,1
+ MTC discriminant	29,0	**33,5**	64,9
Oracle	39,1	47,4	78,2
Adaptation (TED 2014)			
n-code (WMT)	28,5	32,0	33,3
+ MTC SOUL	29,4	32,8	
+ MTC NCE (uniforme)	28,9	32,2	
+ MTC NCE (unigramme)	29,2	33,0	34,9
+ MTC discriminant	**29,8**	**34,1**	**35,9**
Oracle	37,9	46,1	47,6

Tableau 2. *Résultats (scores* BLEU*) pour la tâche de traduction des séminaires TED Talks. Pour chaque scénario, le corpus du domaine est utilisé pour estimer le modèle neuronal de traduction (MTC) selon les approches présentées à la section 3 : le modèle utilisant une couche de sortie structurée de type SOUL, le modèle utilisant l'estimation contrastive bruitée (NCE) et enfin le modèle appris de manière discriminante. Le score oracle est obtenu en sélectionnant, parmi les listes de n-meilleures hypothèses, celles qui ont le meilleur score* $sBLEU$.

4.3. *Résultats expérimentaux*

Les tableaux 2 et 3 rassemblent les résultats expérimentaux pour respectivement les tâches de traduction de séminaires TED Talks et de textes médicaux. À des fins de comparaison, les résultats obtenus avec un système MOSES[15] sont également mentionnés. Ce dernier a été appris selon la configuration par défaut pour la tâche TED Talks, alors que pour les textes médicaux nous avons utilisé le système développé dans le cadre de l'évaluation WMT (Pécheux *et al.*, 2014).

4.3.1. *Comparaison des modèles SOUL et NCE*

En analysant les résultats du tableau 2, un premier constat est que l'ajout d'un modèle de traduction neuronal améliore les performances du système de traduction de base (n-code) dans tous les cas. Remarquons néanmoins que les modèles SOUL et NCE donnent lieu à des résultats comparables et permettent une amélioration du score BLEU entre 0,8 et 1 point selon les scénarios. Ainsi, ces deux manières d'estimer un modèle neuronal à partir de données parallèles permettent d'aboutir à des

15. `http://statmt.org/moses/`

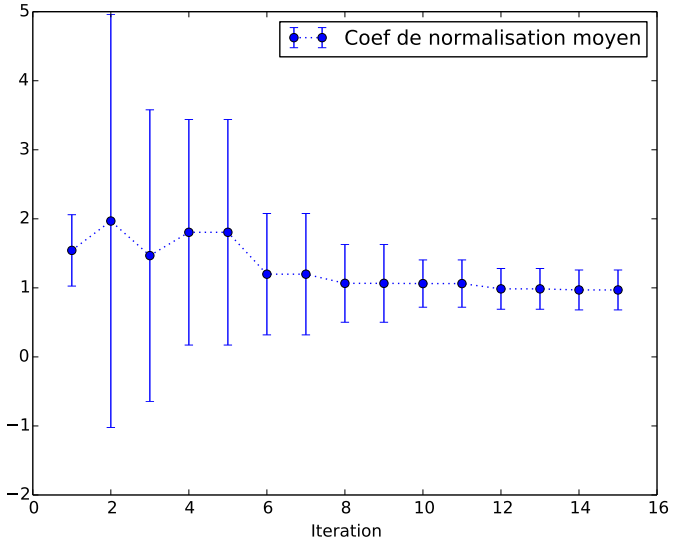

Figure 6. *Évolution du coefficient de normalisation (moyenne et écart-type) pour le modèle NCE. Ce coefficient est calculé pour chaque contexte* $\boldsymbol{c}$ *observé dans les données de développement selon :* $\sum_{w \in \mathcal{V}} \boldsymbol{b_\theta}(w, \boldsymbol{c})$.

performances similaires. La même tendance est observée dans le cas de la traduction de textes médicaux, comme on peut le voir dans le tableau 3.

Afin de mieux faire comprendre comment fonctionne l'apprentissage NCE, nous représentons sur la figure 6 l'évolution des coefficients de normalisation (moyenne et écart-type) du modèle au cours des itérations. Un tel coefficient est calculé pour chaque contexte présent dans les données de validation[16]. Nous observons que la valeur moyenne, ainsi que l'écart-type de ce coefficient, convergent rapidement. Néanmoins, l'écart-type reste élevé, montrant ainsi que le modèle NCE n'est pas exactement normalisé. La figure 7 représente l'évolution de la perplexité[17] mesurée sur les mêmes données de validation. On constate également une convergence rapide et un comportement similaire à celui d'un modèle estimé selon le maximum de vraisemblance. Remarquons néanmoins que les perplexités obtenues avec un modèle NCE sont en général plus élevées que celles obtenues avec un modèle SOUL. Une explication simple est que le modèle SOUL a pour objectif de minimiser la perplexité des données d'apprentissage alors que le critère NCE vise un autre objectif. De plus, nous

16. Le coefficient de normalisation se calcule pour un contexte $\mathbf{c}$ donné par $\sum_{w \in \mathcal{V}} \mathbf{b_\theta}(w, \mathbf{c})$.

17. Le modèle original n'étant pas normalisé, il est nécessaire d'effectuer cette normalisation pour le calcul de la perplexité.

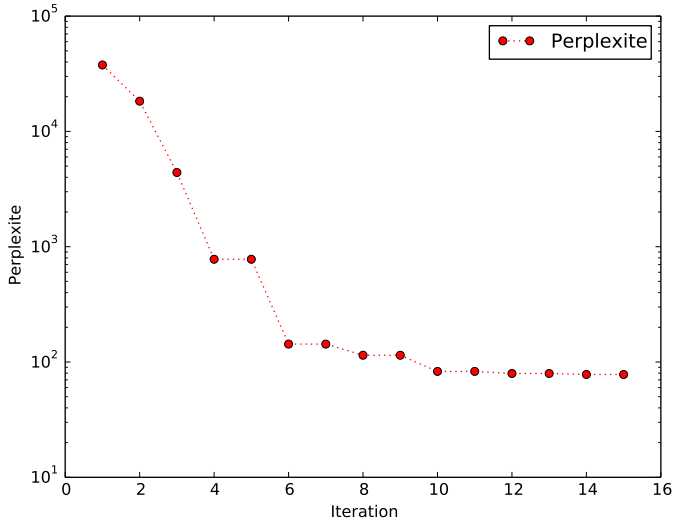

Figure 7. *Évolution de la perplexité mesurée sur les données de validation pour le modèle NCE*

avons constaté empiriquement que pour le modèle NCE, la perplexité n'était qu'un piètre prédicteur des performances en traduction que le modèle permet d'obtenir.

Un dernier point important est que la vitesse de convergence lors de l'apprentissage d'un modèle NCE dépend fortement de la distribution de bruit. Le choix de la distribution unigramme des mots estimée sur les données d'apprentissage semble être le meilleur. Une distribution de bruit uniforme, quant à elle, rend la convergence bien plus chaotique, et le modèle en résultant conduit à des gains en performances bien moindres, voire inexistants. Ainsi, pour la traduction des TED Talks, l'utilisation d'une distribution uniforme dégrade le score BLEU de 0,3 point dans le cas du scénario d'apprentissage et de 0,8 point dans le cas de l'adaptation[18]. Cette étude de l'impact de la distribution de bruit porte uniquement sur la tâche de traduction des TED Talks. Nous avons néanmoins observé le même phénomène pour la traduction de textes médicaux et sur d'autre tâches de traduction.

4.3.2. *Impact de l'apprentissage discriminant*

Dans le cadre de l'apprentissage discriminant, le modèle neuronal est intégré *via* son score non normalisé :

$$f_{\boldsymbol{\theta}}(\mathbf{s}, \mathbf{h}) = \sum_{(w,\mathbf{c})\in(\mathbf{s},\mathbf{h})} \mathbf{a}_{\boldsymbol{\theta}}(w, \mathbf{c}).$$

18. Notons que l'usage d'une distribution unigramme différente de celle observée sur les données d'apprentissage a également un effet négatif sur l'apprentissage et les performances.

Les résultats obtenus pour la tâche de traduction des TED Talks (tableau 2) montrent qu'il est possible d'obtenir des améliorations significatives en score BLEU grâce à l'utilisation du critère d'optimisation discriminant. Par rapport au système de TAS de base, le modèle discriminant permet une amélioration du score BLEU comprise entre 1,3 point (pour le scénario d'apprentissage) et 2,1 points (cas de l'adaptation). Cette différence de gain entre les deux scénarios est plus marquée si l'on compare le modèle discriminant avec les modèles de traduction neuronaux appris de manière conventionnelle (SOUL et NCE) : pour le scénario d'apprentissage, la méthode discriminante ne permet qu'une amélioration du score BLEU de 0,2, alors qu'elle est de 1,1 point pour le scénario d'adaptation.

Afin de mieux comprendre cette différence de performance selon le scénario, le tableau 2 intègre les scores BLEU mesurés sur les données qui servent à l'estimation des modèles neuronaux. On remarque ainsi que dans le cas du scénario d'apprentissage, ce score BLEU est très élevé, puisqu'il est de l'ordre du double de ce même score mesuré sur les données de test. Ceci suggère que le protocole d'apprentissage discriminant est d'une certaine manière sous-optimal, car les données d'entraînement ne reflètent pas la réalité à laquelle le système de traduction sera confronté lors du test : en particulier les hypothèses considérées lors de l'apprentissage sont nettement meilleures que celles traitées lors du test. Les scores BLEU oracles confirment ce constat avec une différence allant également du simple au double entre le score oracle mesuré sur les données d'apprentissage et sur l'ensemble d'évaluation. Dans le cas de l'adaptation, le constat est tout autre et l'on observe des différences bien moindres entre les scores BLEU (oracles ou pas) mesurés sur les données d'apprentissage et de test. Ainsi, les hypothèses vues par le modèle de traduction neuronal diffèrent peu en termes de qualité entre l'apprentissage et le test.

Comme l'ont montré des travaux antérieurs (Do *et al.*, 2014a ; Do *et al.*, 2015), l'initialisation des modèles neuronaux a un impact important sur les performances[19]. Si cela est vrai lorsque ces modèles sont appris, par exemple, selon le maximum de vraisemblance, pour l'apprentissage discriminant, l'impact de l'initialisation est encore plus important. En effet, si le modèle neuronal est au départ de piètre qualité, la phase d'estimation des poids des fonctions caractéristiques (voir par exemple l'équation [8]) aura tendance à réduire l'influence du modèle neuronal, laissant par la suite peu de marge de manœuvre à l'optimisation du critère discriminant. Dans ce travail, les modèles NCE servent de point d'initialisation pour optimiser la fonction objectif de l'apprentissage discriminant.

Les expériences de traduction des textes médicaux (tableau 3) montrent des tendances comparables. Notons néanmoins que l'amélioration du score BLEU pour le scénario d'apprentissage est nettement plus marquée que dans le cas des TED Talks puisqu'il est de 0,7 point lorsque l'on compare le modèle discriminant avec le modèle

19. Les modèles neuronaux, du fait des couches cachées, donnent lieu à des programmes d'optimisation non convexes. L'optimisation étant réalisée par descente de gradient stochastique, le point atteint à convergence dépend de l'initialisation.

	dév.	test	apprentissage
Apprentissage partiel			
MOSES (Patent-Abstract inclus)	$37,0$	$35,3$	
n-code (Patent-Abstract inclus)	$40,4$	$37,4$	$45,8$
+ MTC NCE	$40,8$	$38,1$	$45,2$
+ MTC discriminative	$41,8$	**38,8**	**46,0**
Oracle	$56,0$	$52,7$	$57,6$
Adaptation			
n-code (Patent-Abstract *exclus*)	$39,8$	$37,2$	$39,4$
+ MTC NCE	$41,2$	$38,2$	$40,4$
+ MTC discriminative	**41,8**	**38,9**	**41,5**
Oracle	$55,6$	$52,5$	$50,7$

Tableau 3. *Résultats (scores* BLEU*) pour la tâche de traduction des textes médicaux. Pour chaque scénario, le corpus du domaine, ici les données Patent-Abstract contenant* 200 *k phrases parallèles, est utilisé pour estimer le modèle de traduction neuronal (MTC). Le système de TAS est quant à lui appris en utilisant toutes les données parallèles autorisées par la tâche de WMT'14, le corpus Patent-Abstract étant inclus pour le scénario « Apprentissage partiel » et exclus dans le cas de l'adaptation.*

NCE de départ. Cette différence s'explique par le fait que les données utilisées pour estimer le modèle neuronal ne représentent qu'une partie des données d'apprentissage du système de TAS. Par conséquent, le score BLEU mesuré sur les données d'apprentissage et le score oracle restent comparables à ceux mesurés sur les données de test.

Enfin, le tableau 4 rassemble les vitesses et les temps de calcul liés à l'apprentissage et à l'inférence des différentes méthodes d'apprentissage décrites dans cet article. Ces mesures montrent que les modèles SOUL et NCE sont également équivalents en termes de complexité d'apprentissage et d'inférence.

5. Conclusions

Dans cet article nous avons proposé un cadre discriminant pour l'apprentissage et l'adaptation des modèles de traduction neuronaux. Ce cadre s'appuie sur la définition d'un critère d'optimisation qui permet, d'une part, d'introduire la mesure servant à évaluer la traduction, d'autre part, de prendre en compte l'état courant du système de base pendant l'entraînement, contrairement aux autres méthodes existantes comme l'estimation au maximum de vraisemblance et l'estimation contrastive bruitée. Ces trois critères sont décrits puis comparés expérimentalement dans le cadre d'une tâche de traduction automatique de l'anglais vers le français des séminaires TED Talks et de textes médicaux. Ces deux tâches nous permettent de considérer deux scénarios d'utilisation, l'apprentissage et l'adaptation.

	SOUL	NCE	Discriminant
Vitesse d'entraînement (mots/seconde)	1 000	1 000	
Nombre d'itérations	3	14	
Temps d'entraînement total, init. incl. (heures)	9	9	15
Vitesse d'inférence (mots/seconde)	20 000	25 000	idem NCE

Tableau 4. *Vitesse de traitement lors de l'apprentissage et de l'inférence, ainsi que le temps total d'apprentissage (comprenant la phase d'initialisation) des modèles décrits à la section 3. Si les vitesses d'entraînement des modèles SOUL et NCE sont équivalentes, l'inférence avec le modèle NCE est légèrement plus rapide. On note également que même si l'entraînement NCE demande plus d'itérations pour converger, son initialisation est bien plus simple que celle d'un modèle SOUL, qui nécessite d'organiser les mots du vocabulaire dans une structure d'arbre.*

Les résultats montrent d'une part qu'il est possible d'apprendre un modèle continu de traduction de manière discriminante. Les améliorations obtenues en score BLEU oscillent entre 0,7 et 1,1 point selon la tâche considérée. D'autre part, le cadre discriminant proposé semble particulièrement efficace dans une perspective d'adaptation. En effet, les gains obtenus se trouvent alors entre 1,7 et 2,1 points BLEU selon la tâche, sans qu'il soit nécessaire de reprendre à zéro l'entraînement du modèle neuronal.

En guise de futurs travaux, il semble intéressant d'explorer ce cadre d'apprentissage pour des paires de langues peu dotées en données parallèles. En effet ce cadre semble offrir une meilleure exploitation des données d'apprentissage.

6. Bibliographie

Alkhouli T., Rietig F., Ney H., « Investigations on Phrase-based Decoding with Recurrent Neural Network Language and Translation Models », *Proceedings of the Tenth Workshop on Statistical Machine Translation*, Lisbon, Portugal, p. 294-303, September, 2015.

Allauzen A., Pécheux N., Do Q. K., Dinarelli M., Lavergne T., Max A., Le H.-s., Yvon F., « LIMSI @ WMT13 », *Proceedings of the Workshkop on Statistical Machine Translation*, Sofia, Bulgaria, p. 62-69, 2013.

Bahdanau D., Cho K., Bengio Y., « Neural Machine Translation by Jointly Learning to Align and Translate », *CoRR*, 2014.

Bengio Y., Ducharme R., Vincent P., Jauvin C., « A neural probabilistic language model », *Journal of Machine Learning Research*, vol. 3, p. 1137-1155, 2003.

Casacuberta F., Vidal E., « Machine Translation with Inferred Stochastic Finite-State transducers », *Computational Linguistics*, vol. 30, n° 3, p. 205-225, 2004.

Cherry C., Foster G., « Batch Tuning Strategies for Statistical Machine Translation », *Proceedings of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies (NAACL-HLT)*, p. 427-436, June, 2012.

Cho K., van Merrienboer B., Gulcehre C., Bahdanau D., Bougares F., Schwenk H., Bengio Y., « Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation », *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Association for Computational Linguistics, Doha, Qatar, p. 1724-1734, October, 2014.

Collins M., « Discriminative training methods for hidden Markov models : theory and experiments with perceptron algorithms », *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, p. 1-8, 2002.

Collobert R., Weston J., Bottou L., Karlen M., Kavukcuoglu K., Kuksa P., « Natural Language Processing (Almost) from Scratch », *Journal of Machine Learning Research*, vol. 12, p. 2493-2537, 2011.

Crego J. M., Mariño J. B., « Improving statistical MT by coupling reordering and decoding », *Machine Translation*, vol. 20, n° 3, p. 199-215, 2006.

Crego J. M., Yvon F., Mariño J. B., « N-code : an open-source Bilingual N-gram SMT Toolkit », *Prague Bulletin of Mathematical Linguistics*, vol. 96, p. 49-58, 2011.

Devlin J., Zbib R., Huang Z., Lamar T., Schwartz R., Makhoul J., « Fast and Robust Neural Network Joint Models for Statistical Machine Translation », *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, Baltimore, Maryland, p. 1370-1380, 2014.

Do Q. K., Allauzen A., Yvon F., « Discriminative Adaptation of Continuous Space Translation Models », *International Workshop on Spoken Language Translation (IWSLT 2014)*, Lake Tahoe, USA, 2014a.

Do Q. K., Allauzen A., Yvon F., « Apprentissage discriminant des modèles continus de traduction », *Conférence sur le Traitement Automatique des Langues Naturelles (TALN 2015)*, Caen, France, p. 267-278, 22/06 au 25/06, 2015.

Do Q. K., Herrmann T., Niehues J., Allauzen A., Yvon F., Waibel A., « The KIT-LIMSI Translation System for WMT 2014 », *Proceedings of the Ninth Workshop on Statistical Machine Translation*, Association for Computational Linguistics, Baltimore, Maryland, USA, p. 84-89, June, 2014b.

Federico M., Stüker S., Bentivogli L., Paul M., Cettolo M., Herrmann T., Niehues J., Moretti G., « The IWSLT 2011 Evaluation Campaign on Automatic Talk Translation », *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, European Language Resources Association (ELRA), 2012.

Freund Y., Schapire R. E., « Large margin classification using the perceptron algorithm », *Machine learning*, vol. 37, n° 3, p. 277-296, 1999.

Gutmann M., Hyvärinen A., « Noise-contrastive estimation : A new estimation principle for unnormalized statistical models », *in* Y. Teh, M. Titterington (eds), *Proceedings of th International Conference on Artificial Intelligence and Statistics (AISTATS)*, vol. 9, p. 297-304, 2010.

Hopkins M., May J., « Tuning as Ranking », *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, Edinburgh, Scotland, UK., p. 1352-1362, July, 2011.

Huang E., Socher R., Manning C., Ng A., « Improving Word Representations via Global Context and Multiple Word Prototypes », *Proceedings of the 50th Annual Meeting of the*

Association for Computational Linguistics (Volume 1 : Long Papers), Association for Computational Linguistics, Jeju Island, Korea, p. 873-882, July, 2012.

Jean S., Cho K., Memisevic R., Bengio Y., « On Using Very Large Target Vocabulary for Neural Machine Translation », *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, Beijing, China, p. 1-10, July, 2015.

Kalchbrenner N., Blunsom P., « Recurrent Continuous Translation Models », *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Seattle, Washington, USA, p. 1700-1709, 2013.

Koehn P., *Statistical Machine Translation*, 1st edn, Cambridge University Press, New York, NY, USA, 2010.

Le H.-S., Allauzen A., Yvon F., « Continuous Space Translation Models with Neural Networks », *Proceedings of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies (NAACL-HLT)*, Montréal, Canada, p. 39-48, 2012.

Le H.-S., Oparin I., Allauzen A., Gauvain J.-L., Yvon F., « Structured Output Layer Neural Network Language Model », *Proceedings of ICASSP*, p. 5524-5527, 2011.

Le H.-S., Oparin I., Allauzen A., Gauvain J.-L., Yvon F., « Structured Output Layer Neural Network Language Models for Speech Recognition », *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 21, n° 1, p. 197-206, 2013.

Mariño J. B., Banchs R. E., Crego J. M., de Gispert A., Lambert P., Fonollosa J. A., Costa-Jussà M. R., « N-gram-based Machine Translation », *Computational Linguistics*, vol. 32, n° 4, p. 527-549, 2006.

McDonald R., Crammer K., Pereira F., « Online large-margin training of dependency parsers », *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, p. 91-98, 2005.

Mikolov T., Kombrink S., Burget L., Cernocký J., Khudanpur S., « Extensions of recurrent neural network language model », *Proceedings of ICASSP*, p. 5528-5531, 2011.

Mnih A., Hinton G. E., « Three new graphical models for statistical language modelling », *ICML*, p. 641-648, 2007.

Mnih A., Hinton G. E., « A Scalable Hierarchical Distributed Language Model », *in* D. Koller, D. Schuurmans, Y. Bengio, L. Bottou (eds), *Advances in Neural Information Processing Systems 21*, vol. 21, p. 1081-1088, 2008.

Mnih A., Teh Y. W., « A Fast and Simple Algorithm for Training Neural Probabilistic Language Models », *ICML*, 2012.

Nakamura M., Maruyama K., Kawabata T., Kiyohiro S., « Neural network approach to word category prediction for english texts », *Proceedings of the 13th conference on Computational linguistics (COLING)*, vol. 3, p. 213-218, 1990.

Niehues J., Waibel A., « Continuous space language models using restricted Boltzmann machines. », *Proceedings of International Workshop on Spoken Language Translation (IWSLT)*, Hong-Kong, China, p. 164-170, 2012.

Papineni K., Roukos S., Ward T., Zhu W. J., « BLEU : a method for automatic evaluation of machine translation », *in Proceedings of the 40th Annual meeting of the Association for Computational Linguistics*, p. 311-318, 2002.

Pécheux N., Gong L., Do Q. K., Marie B., Ivanishcheva Y., Allauzen A., Lavergne T., Niehues J., Max A., Yvon F., « LIMSI @ WMT'14 Medical Translation Task », *Proceedings of the Ninth Workshop on Statistical Machine Translation*, p. 246-253, June, 2014.

Schwenk H., « Continuous space language models », *Computer Speech and Language*, vol. 21, n° 3, p. 492-518, July, 2007.

Schwenk H., « Continuous Space Translation Models for Phrase-Based Statistical Machine Translation », *Proceedings of COLING 2012 : Posters*, The COLING 2012 Organizing Committee, Mumbai, India, p. 1071-1080, December, 2012.

Schwenk H., R. Costa-jussa M., R. Fonollosa J. A., « Smooth Bilingual N-Gram Translation », *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Prague, Czech Republic, p. 430-438, 2007.

Socher R., Bauer J., Manning C. D., Andrew Y. N., « Parsing with Compositional Vector Grammars », *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, Sofia, Bulgaria, p. 455-465, 2013.

Sutskever I., Vinyals O., Le Q. V., « Sequence to Sequence Learning with Neural Networks », *Advances in Neural Information Processing Systems (NIPS) 27*, p. 3104-3112, 2014.

Turian J., Ratinov L.-A., Bengio Y., « Word Representations : A Simple and General Method for Semi-Supervised Learning », *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, Uppsala, Sweden, p. 384-394, July, 2010.

Vaswani A., Zhao Y., Fossum V., Chiang D., « Decoding with Large-Scale Neural Language Models Improves Translation », *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Seattle, Washington, USA, p. 1387-1392, October, 2013.

Watanabe T., Suzuki J., Tsukada H., Isozaki H., « Online large-margin training for statistical machine translation », *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2007.

Yang N., Liu S., Li M., Zhou M., Yu N., « Word Alignment Modeling with Context Dependent Deep Neural Network », *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, Sofia, Bulgaria, p. 166-175, August, 2013.

Zens R., Och F. J., Ney H., « Phrase-Based Statistical Machine Translation », *KI '02 : Proceedings of the 25th Annual German Conference on AI*, Springer-Verlag, London, UK, p. 18-32, 2002.