Apport de l'information temporelle des contextes pour la représentation vectorielle continue des mots

Killian Janod[2] *, Mohamed Morchid*[1] *, Richard Dufour*[1] *, Georges Linares*[1]
[1]LIA - University of Avignon (France)
[2]ORKIS - Aix en Provence (France)
[1]firstname.lastname@univ-avignon.fr , [2]killian.janod@orkis.com

Résumé. Les représentations vectorielles continues des mots sont en plein essor et ont déjà été appliquées avec succès à de nombreuses tâches en traitement automatique de la langue (TAL). Dans cet article, nous proposons d'intégrer l'information temporelle issue du contexte des mots au sein des architectures fondées sur les sacs-de-mots continus (*continuous bag-of-words* ou *CBOW*) ou sur les Skip-Grams. Ces approches sont manipulées au travers d'un réseau de neurones, l'architecture CBOW cherchant alors à prédire un mot sachant son contexte, alors que l'architecture Skip-Gram prédit un contexte sachant un mot. Cependant, ces modèles, au travers du réseau de neurones, s'appuient sur des représentations en sac-de-mots et ne tiennent pas compte, explicitement, de l'ordre des mots. En conséquence, chaque mot a potentiellement la même influence dans le réseau de neurones. Nous proposons alors une méthode originale qui intègre l'information temporelle des contextes des mots en utilisant leur position relative. Cette méthode s'inspire des modèles contextuels continus. L'information temporelle est traitée comme coefficient de pondération, en entrée du réseau de neurones par le CBOW et dans la couche de sortie par le Skip-Gram. Les premières expériences ont été réalisées en utilisant un corpus de test mesurant la qualité de la relation sémantique-syntactique des mots. Les résultats préliminaires obtenus montrent l'apport du contexte des mots, avec des gains de 7 et 7,7 points respectivement avec l'architecture Skip-Gram et l'architecture CBOW.

Abstract.

Contribution of temporal context information to a continuous vector representation of words

Word embedding representations are gaining a lot of attention from researchers and have been successfully applied to various Natural Language Processing (NLP) tasks. In this paper, we propose to integrate temporal context information of words into the continuous bag-of-words (CBOW) and Skip-gram architectures for computing word-vector representations. Those architectures are shallow neural-networks. The CBOW architecture predicts a word given its context while the Skip-gram architecture predicts a context given a word. However, in those neural-networks, context windows are represented as bag-of-words. According to this representation, every word in the context is treated equally : the word order is not taken into account explicitly. As a result, each word will have the same influence on the network. We then propose an original method that integrates temporal information of word contexts using their relative position. This method is inspired from Continuous Context Models. The temporal information is treated as weights, in input by the CBOW and in the output layer by the Skip-Gram. The quality of the obtained models has been measured using a Semantic-Syntactic Word Relationship test set. Results showed that the incorporation of temporal information allows a substantial quality gain of 5 and 0.9 points respectively in comparison to the classical use of the CBOW and Skip-gram architectures.

Mots-clés : Réseau de neurones, Représentation vectorielle continue, Information contextuelle, Word2vec , Modèle de langue.

Keywords: Neural network, Continuous vectorial representation, Contextual information, Word2vec, language model.

1 Introduction

Les modèles sémantiques représentant des langages projettent leurs termes dans un espace dans lequel les relations sémantiques entre ces termes peuvent être observées ou mesurées. La technique récente des Word2vec (Mikolov *et al.*, 2013a) construit un réseau de neurones permettant de projeter les termes (contenus dans une fenêtre sémantique définie) d'une langue étudiée dans un espace de représentation vectorielle. Cette projection permet aux mots de sens similaires d'être localisés dans une région de l'espace sémantique proche, comme par exemple les termes "Paris" et "Londres", qui, selon le corpus étudié, peuvent partager l'idée de "Capitale".

Une telle représentation considère le terme dans un environnement contenant un nombre restreint de mots. De plus, ce voisinage réduit ne permet pas de coder d'éventuelles relations entre les termes. En effet, ce groupe de termes est considéré indépendamment de leurs positions ou en "sac-de-mots". Malgré les bons résultats observés lors de l'utilisation des méthodes issues des Word2vec lors de la tâche de représentation sémantique, cette représentation vectorielle ne tient pas compte de la disposition des mots dans le contexte. Cette information est pourtant cruciale lorsque l'on définit le "sens" d'un mot en fonction de son contexte. En effet, plus les termes du contexte sont éloignés du terme central à définir, moins leur relation doit avoir d'importance.

Dans ce papier, nous proposons de pallier cette faiblesse en pondérant les termes contenus dans la fenêtre en fonction de leur distance vis-à-vis du terme central à définir. Cette nouvelle représentation sera évaluée lors de tâches similaires à celles définies dans (Mikolov *et al.*, 2013a), pour les deux architectures introduites dans (Mikolov *et al.*, 2013a) : ***CBOW*** et ***Skip-Gram***. Nous montrons ainsi que la position des termes dans un contexte est une information essentielle, permettant de mieux définir le sens d'un terme dans ce contexte.

Le reste de ce papier est organisé comme suit. La section 2 présente un état-de-l'art des différentes représentations de termes. L'approche proposée, fondée sur la technique des Word2vec, est ensuite détaillée dans la section 3. La section 4 présente les expériences ainsi que les résultats observés avant de conclure dans la section 5.

2 Travaux antérieurs

Les "sacs-de-mots" (Salton, 1989) sont aujourd'hui communément utilisés dans de nombreuses tâches du traitement automatique du langage (TAL). Cette représentation a pour particularité de traiter tous les mots de façon identique, ainsi la séquentialité des mots et leurs relations sont ignorées. D'autres approches ont alors été proposées pour réintroduire la séquentialité des mots, comme par exemple l'approche n-grammes. Cette approche vise à prendre en compte, pour un mot donné, les n mots contenus dans son contexte passé. D'autres stratégies ont ensuite émergé pour capturer la proximité sémantique des mots. La plupart d'entre-elles s'appuient sur l'Hypothèse de Distribution (Sahlgren, 2008) qui implique que des mots représentés dans un même contexte ont un sens proche. Certaines de ces méthodes ont débouché sur des représentations vectorielles continues des mots. Ainsi, les méthodes revues par (Baroni & Lenci, 2010) proposent des modèles où les mots sont représentés par leurs relations à l'ensemble des contextes du corpus d'apprentissage. Cette représentation génère cependant souvent des vecteurs de grande dimension et creux. D'autres stratégies, telles que l'allocation latente de Dirichlet (LDA) (Blei *et al.*, 2003), consistent à découvrir les thèmes latents d'un corpus de texte puis à projeter les mots dans ces espaces thématiques. Cette représentation vectorielle associe pour chaque mot sa distribution dans l'ensemble des thèmes. Ces méthodes ont souvent besoin d'être employées avec une décomposition en valeur singulière, pour réduire leur dimensionnalité, éviter les matrices creuses, et conserver un maximum de la significativité.

Récemment, de nouvelles méthodes de représentation de mots fondées sur les réseaux de neurones se sont développées. Chaque mot est alors représenté par un vecteur plein, de taille modéré, qui correspond à une projection du mot dans un espace où les distances modélisent les relations inter-mots. Ces méthodes sont principalement issues des modèles de langue neuronaux (Bengio *et al.*, 2003; Collobert *et al.*, 2011) et sont déjà utilisées dans plusieurs tâches du TAL (Do *et al.*, 2014; Vaswani *et al.*, 2013). L'approche Word2vec (Mikolov *et al.*, 2013a), fondée elle-aussi sur les réseaux de neurones, suscite un intérêt grandissant dans le domaine du TAL. Une méthode similaire, appelée Glove (Pennington *et al.*, 2014), consiste à factoriser une matrice de co-occurence des mots appris sur une grande quantité de textes. La limite de ces approches Word2vec et Glove est que le la position des mots dans une séquence n'est pas prise en compte : l'ensemble de la séquence de mots est considérée comme un "sac-de-mots", l'ordre des mots étant ignoré.

Nous proposons dans cet article de pallier cette faiblesse en introduisant l'information temporelle des mots (*i.e.* un poids selon leur position) dans la représentation vectorielle continue des mots Word2vec qui l'ignore pour l'instant.

3 Approche proposée

3.1 Méthode d'origine : les sacs-de-mots continus et Skip-Gram de Word2vec

Word2vec est une méthode fondée sur des réseaux de neurones artificiels et définie dans (Mikolov *et al.*, 2013a). Cette méthode propose deux architectures de réseaux de neurones : l'architecture en "sac-de-mots" continus (CBOW) et l'architecture Skip-gram. Ces deux architectures se présentent sous la forme de réseaux de neurones artificiels simples. Ils sont constitués de trois couches : une couche d'entrée, une couche cachée et une couche de sortie. La couche d'entrée contient soit un "sac-de-mots" (CBOW), soit un mot seul (Skip-gram). La couche cachée correspond à la projection des mots d'entrée dans la matrice des poids. Cette matrice est partagée par tous les mots (matrice globale). Enfin, la couche de sortie est composée de neurones "softmax". Pour des raisons de complexité algorithmique due à la couche "softmax", les auteurs dans (Mikolov *et al.*, 2013b) ont introduit deux alternatives appelées "échantillons négatifs" et "softmax hiérarchique". Le couplage de ces fonctions avec la simplicité de ces réseaux leur permettent d'être entraînés sur de très grandes quantités de textes, et ainsi d'obtenir des modélisations de meilleure qualité que les modèles plus complexes à base de récurrence ou de convolution par exemple (Mikolov *et al.*, 2013b, 2010). Un exemple de ces deux architectures est présenté dans la figure 1.

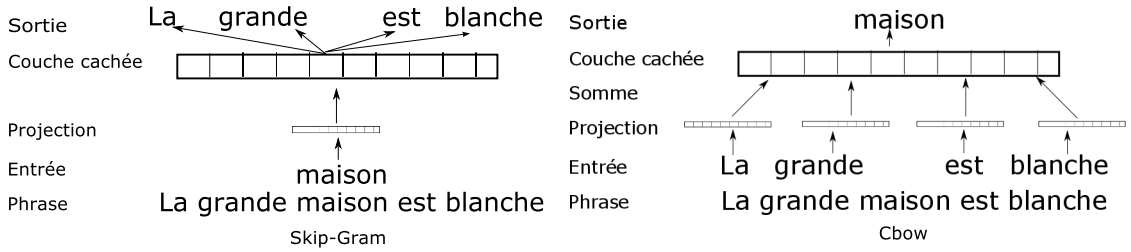

FIGURE 1 – Exemples des architectures CBOW et Skip-gram de Word2vec.

Ces modèles sont capables de capturer des régularités sémantiques et syntaxiques (Mikolov *et al.*, 2013c). En effet, la distance qui sépare la projection de deux mots peut représenter une relation complexe telle que la notion de "singulier-pluriel" ou "masculin-féminin" (Mikolov *et al.*, 2013a) comme le montre la figure 2.

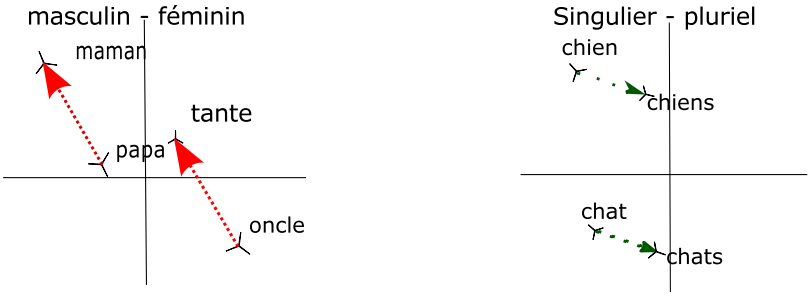

FIGURE 2 – Exemples de relations de mots dans l'espace Word2vec.

3.1.1 Approche par sac-de-mots continus (CBOW)

L'architecture CBOW est un réseau de neurones devant prédire un mot à partir de son contexte. La couche d'entrée représente la présence ou l'absence des mots dans le contexte de manière binaire (*i.e.* 1 pour la présence, 0 pour l'absence). Chaque mot dans le contexte est projeté dans la matrice des poids du modèle. La somme (ou moyenne) de ces représentations passe ensuite par la couche de sortie. Enfin, le modèle compare sa sortie avec le mot seul et corrige sa représentation par rétro-propagation du gradient. Ce modèle cherche à maximiser l'équation :

$$\frac{1}{T}\sum_{t=1}^{T}\log p(m_t|m_{t-\frac{c}{2}}...m_{t+\frac{c}{2}}) \quad (1)$$

où T correspond à l'ensemble des mots dans le corpus et c correspond à la taille de la fenêtre du contexte de chaque mot. Cette architecture présente plusieurs avantages : en effet, en plus d'être efficiente d'un point-de-vue algorithmique (Mikolov *et al.*, 2013a), elle permet à la fois une meilleure modélisation des mots fréquents et une meilleure capture des relations syntaxiques.

3.1.2 Approche par Skip-Gram

L'architecture Skip-Gram tente de prédire, pour un mot donné, le contexte dont il est issu. La couche d'entrée de ce réseau est alors un vecteur ne contenant qu'un seul mot. Le mot est projeté dans la couche cachée puis dans la couche de sortie. Le contexte est ensuite réduit de façon aléatoire à chaque itération. Le vecteur de sortie est ensuite comparé à chacun des mots du contexte réduit et le réseau se corrige par rétro-propagation du gradient. De cette manière, la représentation du mot d'entrée va se rapprocher de chacun des mots présents dans le contexte.

Le réseau de neurones Skip-gram essaie de maximiser une variation de l'équation 1 comme suit :

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{j=t-c,j\neq t}^{t+c}\log p(m_j|m_t) : \tag{2}$$

Comparativement au CBOW, cette architecture permet une meilleure modélisation des mots peu fréquents et permet de mieux capturer les relations sémantiques (Mikolov *et al.*, 2013a).

3.2 Évolution proposée : Distance inter-contexte

L'architecture CBOW traite les mots du contexte en "sacs-de-mots", négligeant l'information chronologique d'apparition des mots et donc de distance dans le contexte. L'architecture Skip-Gram utilise implicitement cette information en réduisant de façon aléatoire la taille du contexte à chaque itération. Par conséquent, la taille du contexte devient un paramètre d'autant plus important puisqu'une fenêtre trop grande dévalorisera le contexte proche des mots et une fenêtre trop courte ne pourra capturer des relations éloignées. Pour répondre à cette problématique, nous avons ajouté une information temporelle de distance inter-contexte (Bigot *et al.*, 2013a,b), le contexte étant centré sur un mot à prédire. Chacun des éléments du contexte se voit attribuer une pondération selon la distance qui le sépare du mot central comme suit :

$$\frac{\alpha}{b+\beta log(d)} \tag{3}$$

où d correspond à la distance en nombre de mots entre le mot au centre du contexte et l'élément du contexte à pondérer. α, b, β et d sont utilisés comme coefficients pour faire varier l'importance de l'information temporelle. Ainsi les contextes proches se voient renforcés et les contextes éloignés auront un impact s'ils sont suffisamment fréquents.

4 Expériences

Nous proposons de comparer les performances des modèles classiques (*i.e.* sans tenir compte de l'information temporelle) et des modèles intégrant la position des mots du contexte dans le réseau de neurones (*i.e.* avec information temporelle) pour les architectures CBOW et Skip-gram de la méthode Word2vec. Les modèles sont entraînés à partir d'une très grande quantité de données (partie 4.1) puis évalués au travers de différentes configurations sur un corpus de test permettant de mesurer la qualité de la relation sémantique-syntaxique des mots (voir partie 4.2).

4.1 Protocole expérimental

Quatre corpus ont été utilisés pour l'apprentissage des modèles (Mikolov *et al.*, 2013a) :

— Le corpus One Billion Word Language Modeling Benchmark de 30 914 405 documents (700 260 470 mots).
— Le premier million de caractères de Wikipedia anglais de 124 303 documents (124 301 845 mots).

— Le corpus GigaWord Anglais de 1994 à 2011 de 190 344 429 documents (3 771 326 692 mots).
— Le Brown Corpus de 57 341 documents (1 019 149 mots).

Pour comparer les performances de notre approche, nous avons choisi de faire varier les paramètres d'apprentissages du réseau de neurones, à savoir la taille du contexte utilisé (5 mots, 10 mots, et le document entier), la taille de la couche cachée du réseau de neurones (120 et 300 neurones), et la fonction de distance utilisée pour l'information temporelle ($\frac{1+log(2)}{1+log(d)}$ et $\frac{log(10)}{5*log(d)}$ appelées respectivement *distance 1* et *distance 2*). Pour chaque condition d'apprentissage, deux modèles sont appris : le modèle classique et le modèle que nous proposons intégrant l'information temporelle du contexte des mots.

Chaque modèle est ensuite évalué avec la tâche de recherche de mots analogues définie dans (Mikolov *et al.*, 2013a). Cette tâche vérifie, pour des ensembles de couples de mots partageant une même relation, que le modèle a bien appris la relation en question. Par exemple, la relation "Capital de" est représentée par la distance entre la paire de mots "Paris" et "France", et par la distance entre la paire de mots "Rome" et "Italie". Pour vérifier qu'une relation similaire lie les deux mots de chaque paire, la question suivante est projetée dans l'espace du modèle par : $Paris - France + Italie = Rome$, qui se traduit par la proximité du vecteur $Rome$ avec le vecteur formé par $Paris - France + Italie$. La tâche est constituée de 19 000 paires de couples comme celle-ci modélisant plusieurs relations différentes comme capitale, monnaie, genre, adjectif opposé, singulier-pluriel...Une fois toutes les relations évaluées, un score est attribué au modèle correspondant au pourcentage de relations correctement modélisées.

4.2 Résultats

Globalement, les tableaux 1, 2 et 3 montrent que les modèles intégrant l'information temporelle sont toujours meilleurs que les modèles de base. Le tableau 2 démontre que plus le contexte utilisé est grand, plus la fonction de distance apporte de l'information jusqu'à l'utilisation des documents entiers comme contexte. Dans notre corpus, moins de 1 % des documents ont plus de 100 mots. Nous utilisons donc un contexte de taille 100 pour prendre en compte le document entier comme contexte. Nos meilleurs gains sont ainsi obtenus avec le document entier (7 points sur le CBOW et 7,7 points sur le Skip-Gram).

Le tableau 2 nous indique que la fonction de distance 2 est plus favorable au modèle CBOW, avec un gain de 4,3 %, mais moins favorable pour le Skip-Gram, avec un gain de 2,1 %. Dans ces mêmes conditions, avec la distance 1, le gain ne dépasse pas 0,9 % pour le CBOW, alors que le gain pour le Skip-Gram atteint 5 %. Enfin dans le tableau 3, nous observons que les modèles possédant une petite couche cachée, ont un gain plus faible que ceux ayant une couche cachée de plus grande taille.

	Skip-gram			CBOW		
Nb de neurones	300					
Taille du contexte	10	15	100	10	15	100
Sans Distance	50,0	50,9	43,7	39	38,9	36,9
Avec Distance 1	55,0	53,7	51,4	39,9	39,6	43,9

TABLE 1 – Performance des modèles selon la taille du contexte des mots (en %).

	Skip-gram	CBOW
Nb de neurones	300	
Taille du contexte	10	
Sans Distance	50,0	39,0
Avec Distance 1	55,0	39,9
Avec Distance 2	52,1	43,3

TABLE 2 – Comparaison des performances selon les distances (en %).

En observant manuellement les exemples de mots des modèles (voir tableau 4), nous remarquons que les modèles avec distance ont tendance à regrouper entre eux des voisins similaires, au contraire des modèles classiques qui ne semblent pas les regrouper. Par exemple, pour "Holidays", l'intégration de la distance permet de regrouper "holiday", "vacation" et "festivities", qui rappellent des mots autour des vacances, "thanksgiving", "easter" et "christmas" à des fêtes religieuses particulières. Ces *catégories* sont moins marquées en l'absence de l'information de distance.

	Skip-gram		CBOW	
Taille du contexte	10			
Taille couche cachée	300	120	300	120
Sans Distance	50,0	43,9	39,0	29,0
Avec Distance 1	55,0	45,1	39,9	30,3

TABLE 3 – Performance sans et avec information temporelle dans un petit espace de projection (10 mots).

Holidays	
Avec Distance	Sans Distance
Holiday	vacations
vacation	thanksgiving
festivities	vacation
thanksgiving	christmas
easter	celebration
christmas	easter

Meat	
Avec Distance	Sans Distance
chicken	pork
beef	not-pasterised
pork	mutton
milk	eggs
eggs	cattle
seafood	chicken

Motherboard	
Avec Distance	Sans Distance
cpu	cpu
cpus	chipset
microprocessor	geforce4
chips	microprocessor
agp	pentium-m
peripherals	cpus

TABLE 4 – Exemples de similarités obtenues pour des mots particuliers sans et avec information temporelle (distance).

5 Conclusion

Ce papier propose un méthode renforçant l'information temporelle des contextes des mots dans les représentations vectorielles. En effet, ces approches, telles que l'approche Word2vec, considèrent l'ensemble des mots d'une séquence indépendamment de leurs positions dans celle-ci. Cette approche en "sac-de-mots" ne permet donc pas de conserver la structure temporelle de la séquence, chaque mot ayant la même importance dans le réseau de neurones. Nous avons alors proposé de pondérer, dans ce réseau de neurones, les termes de la séquence considérée en fonction de leur distance vis-à-vis du terme central à définir. Les expériences préliminaires, menées sur un corpus de test mesurant la qualité de la relation sémantique-syntactique des mots, montrent l'apport du contexte des mots, avec des gains de 7 et 7,7 points respectivement avec l'architecture Skip-Gram et l'architecture CBOW. Nous prévoyons, en perspective, d'étendre cette étude en évaluant le nombre de mots du contexte de manière plus précise ainsi que d'évaluer l'impact d'autres méthodes de poids pour l'intégration de l'information temporelle.

Références

BARONI M. & LENCI A. (2010). Distributional memory : A general framework for corpus-based semantics. *Computational Linguistics*, **36**(4), 673–721.

BENGIO Y., DUCHARME R. & VINCENT P. (2003). A neural probabilistic language model. *Journal of Machine Learning Research*, **3**, 1137–1155.

BIGOT B., LINARÈS G., FREDOUILLE C., DUFOUR R. & LIA C. (2013a). Combining Acoustic Name Spotting and Continuous Context Models to improve Spoken Person Name Recognition in Speech. *Interspeech*, p. 2539–2543.

BIGOT B., SENAY G., LINARES G., FREDOUILLE C. & DUFOUR R. (2013b). Person name recognition in asr outputs using continuous context models. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, p. 8470–8474 : IEEE.

BLEI D. M., NG A. Y. & JORDAN M. I. (2003). Latent dirichlet allocation. *the Journal of machine Learning research*, **3**, 993–1022.

COLLOBERT R., WESTON J., BOTTOU L., KARLEN M., KAVUKCUOGLU K. & KUKSA P. (2011). Natural Language Processing (almost) from Scratch. *The Journal of Machine Learning Research*, **12**, 2493–2537.

DO Q.-K., ALLAUZEN A. & YVON F. (2014). Modèles de langue neuronaux : une comparaison de plusieurs stratégies d'apprentissage. In *TALN 2014*.

MIKOLOV T., CORRADO G., CHEN K. & DEAN J. (2013a). Efficient Estimation of Word Representations in Vector Space. *Proceedings of the International Conference on Learning Representations (ICLR 2013)*, p. 1–12.

MIKOLOV T., KARAFIÁT M., BURGET L., CERNOCKỲ J. & KHUDANPUR S. (2010). Recurrent neural network based language model. In *INTERSPEECH 2010, 11th Annual Conference of the International Speech Communication Association, Makuhari, Chiba, Japan, September 26-30, 2010*, p. 1045–1048.

MIKOLOV T., SUTSKEVER I., CHEN K., CORRADO G. S. & DEAN J. (2013b). Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, p. 3111–3119.

MIKOLOV T., YIH W.-T. & ZWEIG G. (2013c). Linguistic regularities in continuous space word representations. In *HLT-NAACL*, p. 746–751.

PENNINGTON J., SOCHER R. & MANNING C. D. (2014). Glove : Global vectors for word representation. *Proceedings of the Empiricial Methods in Natural Language Processing (EMNLP 2014)*, **12**.

SAHLGREN M. (2008). The distributional hypothesis. *Italian Journal of Linguistics*, **20**(1), 33–54.

SALTON G. (1989). Automatic text processing : the transformation. *Analysis and Retrieval of Information by Computer*.

VASWANI A., ZHAO Y., FOSSUM V. & CHIANG D. (2013). Decoding with large-scale neural language models improves translation. In *EMNLP*, p. 1387–1392 : Citeseer.