TALEP @ DEFT'15 : Le plus cooool des systèmes d'analyse de sentiment

Mickael Rouvier Benoit Favre Balamurali Andiyakkal Rajendran
Aix-Marseille University, CNRS, LIF UMR 7279, 13000, Marseille, France
prenom.nom@lif.univ-mrs.fr

Résumé. Nous présentons dans cet article les systèmes développés par l'équipe TALEP au LIF pour la campagne d'évaluation DEFT'15. La campagne comporte deux tâches : classification des tweets selon leur polarité et classification fine des tweets. Plusieurs systèmes basés sur des modèles probabilistes ont été développés pour chacune des tâches. Puis un système de fusion a été développé combinant les scores des précédents systèmes. La bonne robustess des sytèmes individuels et le système de fusion entre le corpus d'apprentissage et de test nous a permis d'obtenir de bons résultats, bien que très contrastés selon les tâches.

Abstract.

TALEP @ DEFT'15 : The coooolest sentiment analysis systems

This paper describes the systems developed by TALEP team LIF for the DEFT'15 evaluation campaign. This campaign includes two different tasks : valence classification of tweets and fine-grained classification of the tweets. Several systems, all based on probabilistic models, were developed. A final fusion step was developed combining the scores of previous steps. The good robustness of the individual systems and the fusion system between the training and testing corpora allowed us to obtain good results, although well contrasted over the various task.

Mots-clés : analyse de sentiment, réseaux de neurones profonds, word embeddings.

Keywords: sentiment analysis, deep neural network, word embeddings.

1 Introduction

Cette onzième édition du Défi Fouille de Texte (DEFT) était consacrée à l'analyse des sentiments des tweets en français. L'équipe Traitement Automatique du Langage Ecrit et Parlé (TALEP) du Laboratoire d'Informatique Fondamentale (LIF) a participé à cette édition. C'est la première participation dans DEFT de l'équipe TALEP du LIF. L'objectif de notre participation s'inscrit dans le cadre du projet européen SENSEI [1] fondé sur l'étude des conversations humaines ; nous sommes amenés à analyser les sentiments, opinions, émotions des corpus, tels que des transcriptions de conversations téléphoniques ou des commentaires web.

Cette année trois tâches ont été proposées. La première tâche (T1) consiste à classifier les tweets en fonction de l'expression qu'ils expriment (positive, négative ou neutre). La seconde permet de classifier les tweets selon une polarité fine, cette tâche est divisée en deux sous-tâches : la première (T2.1) identifie la classe générique exprimée dans le tweet (opinion, information, sentiment ou émotion) et la seconde (T2.2) identifie le tweet en fonction de l'une des 18 classes proposées [2]. La dernière et troisième tâche (T3) consiste à détecter : la source (l'empan du texte qui désigne explicitement la personne qui exprime l'opinion/sentiment/émotion), la cible (l'empan du texte qui désigne explicitement l'objet de l'opinion/sentiment/émotion) et l'expression du tweet (l'empan de texte dont la valeur sémantique correspond à l'une des 18 classes). L'équipe TALEP a participé seulement aux deux premières tâches (T1, T2.1 et T2.2).

Nous décrivons dans cet article les techniques et les méthodes automatiques utilisées pour ce défi. La section 2 présente le corpus ainsi que les métriques utilisées lors du défi DEFT'2015. Nous présentons ensuite l'architecture du système dans la section 3 et les étapes de pré-traitement de ce système dans la section 4. Dans la section 5, nous présentons les systèmes dont les résultats apparaissent dans la section 6.

1. http://www.sensei-conversation.eu
2. déplaisir, dérangement, mépris, surprise négative, peur, colère, ennui, tristesse, plaisir, apaisement, amour, surprise positive, satisfaction, insatisfaction, accord, valorisation, désaccord, dévalorisation

2 Description de la tâche

2.1 Corpus

Les organisateurs ont mis à la disposition des participants un corpus d'apprentissage composé de 7929 tweets et un corpus de test composé de 3379 tweets. Ce corpus n'a pas été entièrement annoté sur toutes les tâches. Le Tableau 1 donne le nombre de tweets annotés sur le corpus d'apprentissage et de test pour les tâches 1, 2.1 et 2.2 :

	T1	T2.1	T2.2
Apprentissage	7929	6754	3183
Test	3379	3379	1361

TABLE 1 – Nombre de tweets annotés sur le corpus d'apprentissage et de test pour les tâches 1, 2.1 et 2.2.

Afin de tester nos méthodes, de régler leurs paramètres et de palier au phénomène de sur-apprentissage, nous avons décidé de scinder l'ensemble d'apprentissage en 3 sous-ensembles approximativement de la même taille. La procédure d'apprentissage a été la suivante : 2 des 3 sous-ensembles sont concaténés pour produire un corpus d'entraînement et le troisième est utilisé pour le test. La procédure est effectuée trois fois afin que chacun des sous-ensembles du corpus d'apprentissage soit utilisé une fois pour le test. Les ensembles ainsi concaténés seront appelés dorénavant ensembles de développement et le restant ensemble de validation.

2.2 Métriques

Les différents systèmes sont évalués en terme de document correctement classifié (*Accuracy*) et de macro-précision. A noter que dans le cadre du défi DEFT'15, la métrique officielle est la macro-précision. L'*accuracy* et la macro-précision sont calculés comme suit :

$$Accuracy = \frac{\sum_i \text{Nb de documents correctement attribués à la classe i}}{\text{Nb de documents}} \quad (1)$$

$$Macro\_Precision = \frac{\sum_i N_i \cdot P_i}{\sum_i N_i} \quad (2)$$

où N_i est le nombre de documents appartenant à la classe i et P_i est la précision de la classe i.

3 Architecture du système

Le système proposé repose sur une architecture à 2 niveaux (Figure 1). Le premier niveau consiste à obtenir différents points de vue d'un tweet en faisant tourner différents systèmes d'analyse de sentiment. Nous proposons d'utiliser 5 systèmes qui sont décrits plus en détail dans les prochains paragraphes :

— **SVM** : ce système ré-implémente l'approche état-de-l'art de (Mohammad *et al.*, 2013) qui consiste à utiliser un classifieur de type SVM et des uni-grammes, un lexique d'émotion et des paramètres morphologiques.
— **DNN** : ce système est très proche du précédent (SVM) et consiste à utiliser un classifieur de type réseau de neurones profonds, des bi-grammes et un lexique d'émotion.
— **CNN** : ce système ré-implémente l'approche de (Collobert *et al.*, 2011; Kim, 2014) qui consiste à utiliser un classifieur de type réseau de neurones convolutionnels (CNN) et des *Word embeddings*.
— **Doc2Vec** : ce système ré-implémente l'approche de (Le & Mikolov, 2014), qui consiste à utiliser un classifieur de type SVM et comme paramètre des *Doc2Vec*.
— **SuperVector** : ce système est une nouvelle approche qui consiste à utiliser un classifieur de type réseau de neurones profonds sur les statistiques de premier ordre obtenu à partir d'un modèle de mises-von fisher et des *Word embeddings*.

Le second niveau permet de combiner les systèmes du niveau-1. Ainsi, les scores donnés par les différents systèmes du niveau-1 sont groupés dans un vecteur. Un classifieur de type SVM est utilisé sur ce vecteur pour détecter la classe du tweet.

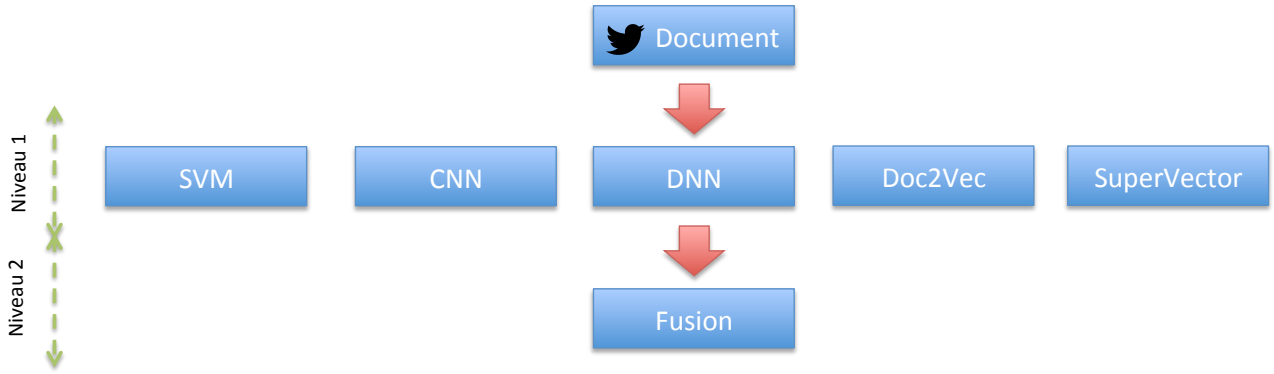

FIGURE 1 – Architecture générale du système TALEP @ DEFT'15

4 Pré-traitements

4.1 Corpus embeddings

La tâche de classification d'analyse de sentiment nécessite de disposer de larges bases de données annotées, afin de capturer l'ensemble des variabilités. De récentes approches ont montré l'intérêt d'intégrer des informations sémantiques dans les systèmes d'analyse de sentiment. Ces approches ont notamment permis de réduire la taille des corpus d'apprentissages. Une approche intéressante est la sémantique distributionnelle qui est une théorie empirique de la sémantique fondée sur l'hypothèse que les mots similaires apparaissent dans les mêmes contextes.

Les *Word embeddings* est une approche de la sémantique distributionelle qui permet de représenter des mots sous la forme de vecteurs de nombres réels. Ces *embeddings* présentent d'intéressantes propriétés de regroupement, puisqu'elle permet de regrouper ensemble les mots qui sont sémantiquement et syntaxiquement proche (Mikolov *et al.*, 2013b). Par exemple, les mots "café" et "thé" vont être très proches dans cet espace. Le but est d'utiliser ces traits dans les classifieurs.

Pour apprendre les *Word embeddings*, nous avons créé un corpus non-annoté de tweets de sentiment en français. Ces tweets ont été récupérés sur la plateforme Twitter [3] en effectuant des recherches avec des mots-clefs porteurs d'émotion, sentiment ou d'opinion. Ces mots-clefs peuvent être des termes (comme par exemple : vexé, annihilé, ridicule...), des hashtags (#good, #like, #mauvais...) ou des smileys (;), :-), :-D). Ce corpus est composé d'environ 16 millions de tweets en français, téléchargés entre le 27 février et le 14 avril 2015. Le corpus collecté ainsi que tous les mots-clefs servant à collecter celui-ci sont disponibles ici : `https://github.com/mrouvier/tweet_corpus_fr`.

Dans nos expériences, nous utilisons le toolkit Word2Vec (Mikolov *et al.*, 2013a) pour extraire les *Word embeddings*. Cette approche consiste à entraîner un réseau de neurones linéaires, où la matrice des poids de la couche linéaire peut ainsi être interprétée comme une projection linéaire permettant de passer de l'espace des mots à une représentation vectorielle. Nous utilisons l'approche Continuous Bag of Words (CBOW) qui consiste à entraîner le réseau à prédire un mot à partir de son contexte.

4.2 Pré-traitements

Une étape de pré-traitement est appliquée aux tweets :

— Encodage des caractères : tous les tweets sont encodées au format UTF-8

3. `http://www.twitter.com/`

— Encodage des balises HTML : certains caractères ont des significations spéciales en HTML, et doivent être remplacés par des entités HTML (comme par exemple : &lt ;, &gt ;...)
— Minuscule : tous les caractères sont convertis en minuscule
— Rallongement : le rallongement des caractères qui consiste à répéter plusieurs fois un caractère dans un mot. C'est une méthode souvent utilisée sur le web pour insister sur un fait. Ce rallongement est souvent corrélé à un sentiment. Si un caractère est répété plus de trois fois, il sera réduit à trois caractères.
— Tokenization : la tokenization réalise le découpage d'une phrase en unités pré-lexicales. Cette tokenization est basée sur le toolkit macaon (maca_tokenize) (Nasr *et al.*, 2010). Il repose sur une grammaire régulière qui définit un ensemble de types d'atomes. Un analyseur lexical détecte les séquences de caractères (en fonction de la grammaire) et leur associe un type. Nous avons rajouté les atomes pour la détection des smileys, hashtags et noms d'utilisateurs (atome spécifique aux tweets).
— Ponctuation : nous supprimons ici tous les caractères de ponctuation.
— Stemming : le stemming consiste à supprimer le suffixe et le préfixe des mots, laissant ainsi son radical. L'algorithme utilisé est le *Porter stemming algorithm*.

L'ensemble des outils est disponible ici : `https://github.com/mrouvier/tweet_tokenizer_fr`.

5 Systèmes

Nous allons dans cette section présenter les cinq systèmes du niveau-1 puis le système de fusion du niveau-2.

5.1 SVM

Notre premier système appelé *SVM* consiste à ré-implémenter l'approche état-de-l'art pour l'analyse de sentiments sur les tweets de (Mohammad *et al.*, 2013). Ce système utilise comme classifieur un *Support Vector Machine* (SVM) et comme paramètre : un sac-de-mots (uni-gramme), un lexique d'émotion ainsi que des paramètres morphologiques.

Concernant le lexique d'émotion, de nombreux travaux ont montré l'importance d'utiliser des dictionnaires de polarités (Hatzivassiloglou & McKeown, 1997; Taboada *et al.*, 2011; Kanayama & Nasukawa, 2006; Wilson *et al.*, 2005). Ces lexiques d'émotion permettent d'augmenter l'espace des traits ou bien d'affiner la sélection des traits pertinents.

Nous proposons de créer un lexique d'émotion en français en traduisant de manière automatique les lexiques d'émotions disponibles en anglais (Bing Liu's Opinion Lexicon, MPQA Subjectivity Lexicon, SentiWordNet et Harvard General Inquirer). L'approche classique consiste à utiliser un système de traduction automatique. Malheureusement ces systèmes montrent leur limite lorsque l'on veut traduire un corpus de spécialité (comme ici les tweets).

Nous proposons d'utiliser l'approche *Bilingual word embeddings* (Zou *et al.*, 2013). Cette approche consiste à estimer une matrice de projection (mapping) d'un jeu d'embedding à un autre, tout en préservant (à des degrés différents) les structures syntaxique et sémantique. Plus concrètement, nous estimons un jeu de *Word embeddings* en anglais et en français. A l'aide d'un dictionnaire, nous apprenons une matrice de projection qui consiste à réduire la distance entre l'ensemble des paires du dictionnaire, d'un jeu d'embeddings, d'une langue à une autre. Ainsi, la traduction d'un mot en une autre langue se fait à l'aide des embeddings et de cette matrice de projection.

Cette approche a permis, par exemple, de traduire correctement l'expression "looool" par "mdrrr" ; ce qui n'aurait pas été le cas à l'aide d'un système de traduction classique.

Nous utilisons les paramètres morphologiques suivants :

— *All-caps* : Le nombre de mots en majuscule.
— *Emoticons* : Est-ce que le dernier token du tweet est un émoticon ?
— *Elongated units* : Le nombre de mots dont les caractères se répètent plus de deux fois (par exemple : looooool)
— *Punctuation* : Le nombre de séquences contigües de points, points d'exclamation et points d'interrogation

Dans nos expériences, les *Word embeddings* sont appris sur des corpus de tweets. Le dictionnaire utilisé est celui obtenu après alignement des bibtex du corpus Europarl. La taille des *Words embeddings* est de 300 et nous utilisons le classifieur libLINEAR [4].

4. http ://www.csie.ntu.edu.tw/ cjlin/liblinear/

5.2 DNN

Le deuxième système appelé *DNN* utilise comme classifieur un réseau de neurones profonds (Deep Neural Network - DNN) et deux paramètres : un sac-de-mots (bi-gramme) et un lexique de polarité (présenté dans la section précédente). Le DNN est composé de deux couches cachées, contenant chacune 2048 neurones. Les fonctions d'activation utilisées sont des *tanh* et nous utilisons comme DNN le toolkit Kaldi (Povey *et al.*, 2011).

5.3 CNN

Le troisième système appelé *CNN* consiste à ré-implémenter l'approche proposée dans (Collobert *et al.*, 2011; Kim, 2014). Cette approche est basée sur l'utilisation d'un réseau de neurones convolutionnels (Convolutionnal Neural Network - CNN), composé de trois couches cachées : la première consiste à extraire un vecteur pour chacun des mots ; la seconde est une couche convolutionnelle (qui partage les poids entre tous les mots) ; et la dernière est une couche de max-pooling.

Dans nos expériences, nous initialisons la première couche cachée du CNN avec la matrice des *Word embeddings* obtenue sur le corpus embeddings. La taille des *Word embeddings* est de 300, la taille du vecteur convolutionnel est de 400 et nous utilisons un dropout à 0.4.

Tout le matériel nécessaire (code source et données) pour reproduire les résultats sont accessibles ici : `https://github.com/mrouvier/tweet_cnn_fr`

5.4 Doc2vec

Ce quatrième système appelé *Doc2Vec* consiste à extraire un vecteur continu d'une phrase. Cette approche consiste à ré-implémenter l'approche proposée dans (Le & Mikolov, 2014). C'est une approche non-supervisée similaire à l'approche *Word2Vec*. L'avantage de cette approche est que le vecteur est extrait sur des phrases de tailles variables. Dans nos expériences, la taille du vecteur du système est de dimension 600 et le classifieur utilisé est un SVM.

5.5 Supervector

Ce cinquième et dernier système appelé *SuperVector* est basé sur l'idée des Speakers Embeddings proposé dans (). L'idée est de structurer l'espace des *Word embeddings* avec un modèle mises-von fisher, puis d'extraire les statistiques de premier ordre de ce modèle. Le SuperVector obtenu est utilisé comme paramètre d'entrée d'un DNN qui contient deux couches cachées de 2048 neurones chacune. La fonction d'activation utilisée est une *tanh*. Le toolkit utilise pour les DNN est celui de Kaldi.

5.6 Fusion

Dans l'optique d'améliorer les résultats, nous proposons de fusionner des systèmes, ce qui permet d'augmenter facilement la robustesse des règles de classification en multipliant les points de vue sur le même phénomène. Cette approche a été utilisée régulièrement dans les différents défis DEFT (Oger *et al.*, 2010; Torres-Moreno *et al.*, 2007, 2009; Grouin, 2014) et a permis d'améliorer les gains de classification. Nous proposons de réaliser la fusion au niveau des scores fournis par chaque système en ajoutant chacun d'eux dans un vecteur utilisé avec un classifieur de type SVM. L'idée est d'apprendre au SVM les différentes régularités existantes entre les systèmes.

6 Résultats

Nous reportons dans cette section les résultats pour les tâches 1, 2.1 et 2.2.

6.1 Tâche 1

Le Tableau 2 reporte les résultats obtenus par le système de fusion et les systèmes du niveau-1 (les systèmes *SVM*, *CNN*, *Doc2Vec*, *DNN* et *SuperVector*) pour la tâche 1. Le meilleur système du niveau-1 est le système *CNN* qui permet d'obtenir une macro-précision de 71.74%. Le système de fusion permet d'obtenir un gain de 1.86 points.

Système	Accuracy	Macro-precision
Fusion	**0.7257**	**0.7360**
SVM	0.6893	0.6882
CNN	0.7100	0.7174
Doc2Vec	0.6055	0.5970
DNN	0.6632	0.6600
SuperVector	0.5857	0.574

TABLE 2 – Résultat en terme d'*accuracy* et de macro-précision pour le système de fusion et les systèmes de niveau 1 sur la tâche 1.

6.2 Tâche 2.1

Le Tableau 3 reporte les résultats du système de fusion et les systèmes de niveau-1 pour la tâche 2.1. On constate que le meilleur système du niveau-1 est le système *CNN*. Il permet d'obtenir une macro-précision de 57.26%. Malheureusement dans cette tâche, le système de fusion n'a pas permis d'améliorer la macro-précision et obtient une macro-précision de 55.82%.

Système	Accuracy	Macro-precision
Fusion	**0.6188**	0.5582
SVM	0.5963	0.5355
CNN	0.6011	**0.5624**
Doc2Vec	0.5407	0.4350
DNN	0.5750	0.5000
SuperVector	0.5439	0.4702

TABLE 3 – Résultat en termes d'*accuracy* et de macro-précision pour le système de fusion et les systèmes de niveau 1 sur la tâche 2.1.

6.3 Tâche 2.2

Le Tableau 4 reporte les résultats du système de fusion et des systèmes de niveau 1 pour la tâche 2.2. Le meilleur système de niveau 1 est le système *SVM* qui permet d'obtenir 31.9% de macro-précision. Le système de fusion a permis d'améliorer les résultats et d'obtenir 32.69% de macro-précision (soit un gain de 0.79 point).

Système	Accuracy	Macro-precision
Fusion	**0.612**	**0.3269**
SVM	0.5981	0.319
CNN	0.6113	0.3065
Doc2Vec	0.5672	0.2805
DNN	0.5893	0.3062
SuperVector	0.5511	0.2922

TABLE 4 – Résultat en termes d'*accuracy* et de macro-précision pour le système de fusion et les systèmes de niveau 1 sur la tâche 2.2.

7 Conclusion et perspectives

La classification de tweets est une tâche qui peut être très difficile en fonction du type de tweets. Comme cela avait été constaté lors des défis précédents, "La lecture directe ne suffit pas toujours pour se forger un avis et privilégier une classification par rapport à une autre" (Torres-Moreno *et al.*, 2007). Nous avons utilisé des approches de représentation numérique et probabiliste afin de rester aussi indépendant que possible des sujets traités. Concernant les systèmes de base, le CNN obtient de bonnes performances sur l'ensemble des trois tâches, et la fusion des systèmes ont permis d'améliorer les résultats.

Remerciements

Ces travaux de recherche ont été financés en partie par l'Union Européenne à travers le projet SENSEI [5] (FP7/2007-2013 - nº 610916 – SENSEI).

Références

COLLOBERT R., WESTON J., BOTTOU L., KARLEN M., KAVUKCUOGLU K. & KUKSA P. (2011). Natural language processing (almost) from scratch.

GROUIN C. (2014). Les 10 ans du défi fouille de texte deft.

HATZIVASSILOGLOU V. & MCKEOWN K. R. (1997). Predicting the semantic orientation of adjectives. In *Proceedings of the Computational linguistics*.

KANAYAMA H. & NASUKAWA T. (2006). Fully automatic lexicon expansion for domain-oriented sentiment analysis. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

KIM Y. (2014). Convolutional neural networks for sentence classification.

LE Q. V. & MIKOLOV T. (2014). Distributed representations of sentences and documents.

MIKOLOV T., CHEN K., CORRADO G. & DEAN J. (2013a). Efficient estimation of word representations in vector space.

MIKOLOV T., YIH W.-T. & ZWEIG G. (2013b). Linguistic regularities in continuous space word representations. In *Proceedings of the conference on Human Language Technology (HLT)*.

MOHAMMAD S. M., KIRITCHENKO S. & ZHU X. (2013). Nrc-canada : Building the state-of-the-art in sentiment analysis of tweets. In *Proceedings of the Second Joint Conference on Lexical and Computational Semantics (SEMSTAR)*.

NASR A., BÉCHET F. & REY J.-F. (2010). Macaon : Une chaîne linguistique pour le traitement de graphes de mots. *Actes de Traitement Automatique du Langage Naturel (TALN)*.

OGER S., ROUVIER M., CAMELIN N., KESSLER R., LEFÈVRE F. & TORRES-MORENO J.-M. (2010). Système du lia pour la campagne deft'10 : datation et localisation d'articles de presse francophones. *Actes du sixième Défi Fouille de Textes (DEFT)*.

POVEY D., GHOSHAL A., BOULIANNE G., BURGET L., GLEMBEK O., GOEL N., HANNEMANN M., MOTLÍČEK P., QIAN Y., SCHWARZ P. *et al.* (2011). The kaldi speech recognition toolkit.

TABOADA M., BROOKE J., TOFILOSKI M., VOLL K. & STEDE M. (2011). Lexicon-based methods for sentiment analysis. *Proceedings of the Computational linguistics*.

TORRES-MORENO J. M., EL-BÈZE M., BÉCHET F. & CAMELIN N. (2007). Comment faire pour que l'opinion forgée à la sortie des urnes soit la bonne ? application au défi deft 2007. *Actes du troisième Défi Fouille de Textes (DEFT)*.

TORRES-MORENO J.-M., EL-BÈZE M., BÉCHET F. & CAMELIN N. (2009). Fusion probabiliste appliquée à la détection et classification d'opinions. *Actes du cinquième Défi Fouille de Textes (DEFT)*.

WILSON T., WIEBE J. & HOFFMANN P. (2005). Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the conference on Human Language Technology (HLT)*.

ZOU W. Y., SOCHER R., CER D. M. & MANNING C. D. (2013). Bilingual word embeddings for phrase-based machine translation. In *Proceedings of the Empirical Methods in Natural Language Processing (EMNLP)*, p. 1393–1398.

5. http://www.sensei-conversation.eu