24th European Conference on Artificial Intelligence - ECAI 2020
Santiago de Compostela, Spain

Learning Joint Job Embeddings using a Job-Oriented Asymmetrical Pairing System

Timothé Bernard[1] and **Tanguy Moreau**[1] and **Clément Viricel**[1] and **Paul Mougel**[1] and **Christophe Gravier** [2] and **Frédérique Laforest**[3]

Abstract. A main objective of Human Resources (HR) professionals is to suggest relevant job opportunities to people seeking change. This requires they identify key features in thousands of badly entitled, loosely-defined and often long job descriptions to compare job offers. However, an ever-increasing number of profiles and many similar-looking job opportunities combined with a quickly evolving job market make this task significantly harder and require advanced automatic tools.

In this work, we tackle these problems by building a job embedding space called *job2vec* and a title predictor from a description. To do so, we propose the first Job-Oriented Asymmetrical Pairing System, JOAPS which on one hand uses a title encoder to create the *job2vec* and on the other hand uses a description encoder to predict a job title from a job description. The JOAPS architecture is based on two different sub-neural networks connected at their outputs using a novel weighted ranking pairwise loss, that we named Weighted On-the-fly Chunk Ranking (WOCR) loss, to back-propagate simultaneously to both sub-networks.

First, we compare the *job2vec* built by JOAPS to a word2vec-based *job2vec*. Then we compare the JOAPS description encoder to a Sequence to Sequence baseline model. Our experiments show that JOAPS *job2vec* incorpore HR key features, whereas word2vec-based *job2vec* only combines features extracted from job title words. Thus JOAPS *job2vec* has more suitable properties to compare jobs, even with never-seen-before job titles. Futhermore, we show that JOAPS description encoder outperforms the Seq2Seq baseline method in title prediction.

1 Introduction

Convolutional Neural Networks (CNN) can model complex data by learning and combining higher-level features made of a combination of lower-level features. Such neural networks have historically been used for computer vision where their ability to detect simple features and combine them into more complex patterns is a useful property for image recognition [22, 24, 33]. The ability of CNNs to extract latent structures can also be applied to textual data: just as a CNN can capture features in images, it can extract and combine features about each word of a document [23]. This approach has produced strong results in various Natural Language Processing (NLP) tasks such as part-of-speech tagging, named entity recognition or sentence classification [10, 11, 21, 23, 39].

In NLP, sentences can also be modelled as temporal sequences of words. Recurrent Neural Network (RNN) architectures [14, 19, 29] are well fitted for such sequences. The main ability of a RNN is to capture essential information along sequences using previous information. RNNs have been very successful in many NLP tasks such as speech recognition, text generation and recently machine translation [15, 16, 25, 34, 35].

Word representation is a major research topic as words are the main input of NLP tasks. First processed as discrete atomic symbols, they have been later modeled as vectors, called word embeddings [6, 28, 30, 32], which have the ability to store useful linguistic and semantic information. As semantically-related words map to similar vectors, a well-built vector space can improve the performance and generalization abilities of downstream tasks [5, 11].

In our application domain, Human Resources (HR) professionals suggest job opportunities to employees looking for new challenges. They build relationships between jobs and profiles by looking at HR key features such as skills, activities, industries, tools or experience levels. However, considering the very large number of profiles on one side and the massive, quickly evolving job market on the other side makes this task significantly harder. Machine learning and NLP techniques provide new tools to tackle this challenge [3, 13, 20, 27, 40, 41, 42]. One of the main tasks in HR is to compare an employee's career with a job opportunity so as to know if the opportunity fits the employee. HR professionals compare an employee's career with a job opportunity by looking into the employee's experiences and match HR key features extracted from both experiences and opportunity descriptions. With some experience, HR professionals carry out this task by first - and even only - looking into the job titles without reading their descriptions.

To reproduce this behaviour, we aim to build a vector space that embeds jobs not only with contextual meaning but also upon key features similar to the ones defined by HR professionals. This job embedding space, called *job2vec*, allows us to tackle the challenge of comparing experiences of an employee with a job title. [3, 40] already investigate on building a job embedding space. Another challenge is that job opportunities or experiences can have meaningless titles or no title at all. Therefore, we also need to train a predictor that could suggest a job title from a job description.

Both challenges are restricted by production constraints that are: a limited amount of RAM, no use of GPU because of prohibitive long term costs and the need to process client data rapidly. As written above, jobs are evolving quickly, new jobs including new descrip-

[1] Artificial Intelligence for Human Resources (HaiR) 365Talents, email: firstname.lastname@365talents.com
[2] Univ. Lyon, Univ. St-Etienne, Laboratoire Hubert Curien CNRS UMR 5516, email: christophe.gravier@univ-st-etienne.fr
[3] Univ. Lyon, INSA Lyon, LIRIS CNRS UMR 5205, email: frederique.laforest@insa-lyon.fr

tions are often created, therefore the model needs to be updated frequently; large language models like BERT [12] are then not suitable.

In this paper, we describe a new architecture called Job Oriented Asymmetrical Pairing System (JOAPS) inspired by `#TagSpace` [37, 38] for the purpose of jointly creating a *job2vec* and a title predictor.

JOAPS is built on two asymmetrical sub-neural networks which can be seen as a Siamese Neural Network [8] with two different sub-networks. We designed the first sub-network as a job title encoder and the second as a job descriptions encoder. The two sub-neural networks are connected at their outputs by a novel weigthed pairwise ranking loss [36], the Weighted On-the-fly Chunk Ranking (WOCR) loss. This loss is used to optimize simultaneously the two sub-networks.

Using HR professionals evaluation, we show that *job2vec* from JOAPS is more consistent in terms of HR key features than a *job2vec* based on word2vec. We also show that JOAPS outperforms a Sequence to Sequence (Seq2Seq) baseline to predict a title from its description. We point out that JOAPS focuses on HR components in a description that are useful to predict title. Finally, JOAPS has less parameters and processes data faster than a Seq2Seq baseline.

In the following, section 2 defines the architecture of JOAPS by detailing the title and description encoders along with the WOCR loss. In Section 3, we introduce our dataset containing French job opportunities and formalize the evaluation setup. Finally, Section 4 reports experimental results.

2 JOAPS: Job Oriented Asymmetrical Pairing System

In this section we describe the architecture of JOAPS and present the novel weighted pairwise ranking loss.

2.1 JOAPS System Architecture

As preprocessing, We split each title t and then pad them to length $m = \max_t(length(t))$. We embed each word of t a word2vec, named W_{in}, pre-trained on large HR dataset (more than 16M profiles and offers), containing more than 1.4 million words in its vocabulary. W_{in} output a matrix of size $m \times e$ with e being the vector size of W_{in}. We apply the same process to the descriptions d with padding length $n = \max_d(length(d))$. The architecture of JOAPS is built on two asymmetrical sub-networks, as shown in Figure 1.

The first sub-network, the title encoder, is based on a bidirectional recurrent neural network with a Gated Recurrent Unit [9, 17, 29, 35]. The RNN encodes each phrase (word by word) into a vector of size h.

The second sub-network is the description encoder: it encodes words from a job description into a vector using a CNN [11]. The use of a CNN ensures a fast processing [1] to extract HR key feat ures such as skills, activities, industries, tools or experience levels within a job description. The architecture of the description encoder is described in figure 2.

The $n \times e$ matrix is fed to the description encoder consisting of a 1-D convolutional layer of f filters with filters size of s words. The CNN outputs a $n \times f$ matrix. We then apply tangent activation and max-pooling over each filter [11] to output a vector of size f. At the end, a fully connected layer outputs an encoded vector of size h. Finally, we use a novel weighted pairwise ranking loss, called WOCR loss for Weighted On-the-fly Chunck Ranking loss, between the two output vectors to jointly optimize both encoders. Which is

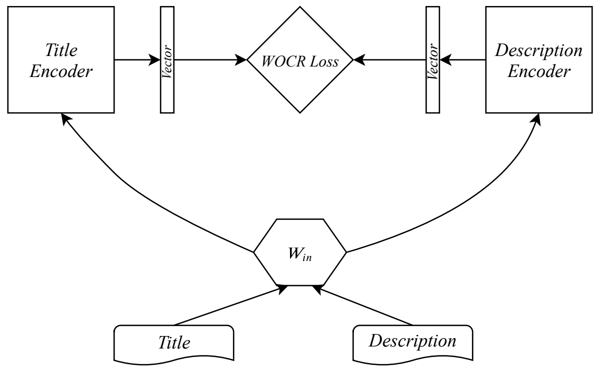

Figure 1: JOAPS training architecture.

explained in the next subsection. Once training is complete, we create *job2vec* by encoding job titles through the title encoder. Our architecture allows us to process job titles never seen before (during training), which is a must in a field where new jobs are constantly being created.

2.2 Weighted On-the-fly Chunck Ranking Loss

We use a novel weighted pairwise ranking loss to jointly backpropagate in both sub-networks [36]. Such a weighted ranking loss is particularly useful in the construction of ranked space in information retrieval tasks.

A weighted ranking pairwise optimization works as follow. Let $\mathcal{D}$ be the set of inputs and $\mathcal{T}$ the set of labels. Given an input $d \in \mathcal{D}$ and its unique true label $t \in \mathcal{T}$, let $\mathcal{N}_t$ be the set of negatives label of t (thus $\mathcal{N}_t \subset \mathcal{T} \setminus \{t\}$). Let $dist$ be a distance function. To compute a ranking pairwise loss $\mathcal{L}$ for a couple of description and title (d, t), we look for negatives $t_j \in \mathcal{N}_t$ that are ranked higher than t and refer to them as violating negatives v_j i.e when equation (1) is satisfied :

$$dist(d, t_j) < dist(d, t) \tag{1}$$

The model is fully trained when it is impossible to find any violating negative for each couple $(d, t) \in \mathcal{D} \times \mathcal{T}$. When there are still some violating negatives, we compute the ranking pairwise loss as follows:

$$\mathcal{L}(d, t, t_j) = dist(d, t) - dist(d, v_j) \tag{2}$$

To speed up convergence, one usually uses weighted ranking loss [36]. It features a weight ω that differs among applications.

$$\mathcal{L}(d, t, t_j) = \omega\,(dist(d, t) - dist(d, v_j)) \tag{3}$$

As it is, the ranking pairwise loss is time consuming. Indeed for each couple $(d, t) \in \mathcal{D} \times \mathcal{T}$, we need to compute equation 2 for each negative t_j and then retropropagate for each violating negatives v_j.

Weston et al. suggest an optimisation of the weighted pairwise ranking loss to speed up the training in order to correctly rank hashtags for descriptions. They call their loss the Weighted Approximate-Rank Pairwise (WARP) loss [37, 38]. For weighted ranking loss, as training progresses and the model is tuned, each description needs more and more negatives in order to find a violating negative and then trigger a backpropagation. This makes each iteration longer than the previous one. Therefore, they suppose that the violating negatives are equally distributed among the negatives and propose an approximation where they stop browsing when the first violating negative v_j is

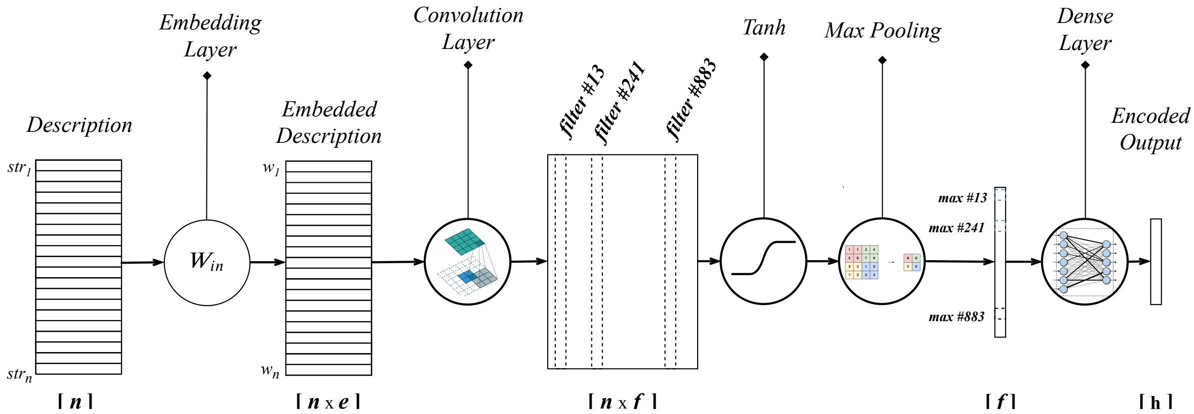

Figure 2: Architecture of the Description Encoder.

found at rank j (equation 1). They compute the weight $\omega = \sum_{i=1}^{r} \frac{1}{i}$ with $r = \left[\frac{k}{j}\right]$ where $k = Card(\mathcal{N}_t)$ is the number of negatives and j is the rank of v_j. In order to be more restrictive, they add a margin μ in the search for the first violating negative:

$$\begin{aligned} \mathcal{L}(d,t,v^j) &= \sum_{i=1}^{r} \frac{1}{i} \max(l, 0) \\ l &= dist(d,t) - dist(d,v^j) - \mu \end{aligned} \tag{4}$$

To parallelized the WARP loss over description d on a batch b with size 2^p (with $p \geq 1$), it requires to keep a batch of 2^p couples of encoded descriptions and titles but also $2^p \times k$ samples of encoded negatives $t^j \in \mathcal{N}_t$ and finally retropropagate 2^p times for each (d, t). Not to have to search among too many negatives, the size k of $\mathcal{N}_t$ is set to 1000.

In this work, we propose a novel weighted pairwise ranking loss called Weighted On-the-fly Chunk Ranking (WOCR) loss. The main idea of our WOCR loss is to set the ensemble of negatives $\mathcal{N}_t$ to be an adaptative chunk during training.

For a batch b of encoded descriptions and titles (d, t), we set the chunck $\mathcal{N}_t$ to be all the job titles in the batch. Then, we count all violating negatives v_j in the chunk of negatives $\mathcal{N}_t$. We retropropagate on each (d, t, v_j) with ω as the piecewise ratio of violating negatives.

$$\begin{aligned} \mathcal{L}(d,t,t_j) &= \omega \left(dist(d,t) - dist(d,t_j) \right) \\ \omega &= \frac{\sum_{\mathcal{N}_t} I\left(dist(d,t) > dist(d,t_j) \right)}{2^p} \end{aligned} \tag{5}$$

It is worth noticing that ω will differ for each couple (d, t) in batch b. $I(\cdot)$ is the characteristic function. In practice, we need to keep a batch b of couple of encoded descriptions and titles (d, t), 2^p encoded job titles and retropropate a maximum of $2^p \times (2^p - 1)$ times. Nevertheless, this loss is constrained by the batch size. Thus, we adjust batch size $b = 2^p$ on the fly. If the ratio of violating negatives is not high enough, we increase $b = 2^{p+1}$; if there are too many violating negatives, we decrease $b = 2^{p-1}$. It should be noted that when the model converges, b increases. A summary of the differences between the WARP loss and the WOCR loss is given in Table 1.

We have tested both losses on a set of $(d, t) \in \mathcal{D} \times \mathcal{T}$ with $Card(\mathcal{D}) = 500\,000$ and then computed an average time per couple. The WARP loss was five times longer than the WOCR loss (times are reported in Table 1). We found absolutely no result differences between the models produced by the two losses. The difference between both losses is mainly a significant difference in convergence time: WARP loss is basically not tractable on commodity servers. Thus, we choose the WOCR loss as the weighted ranking loss for our ranking optimization.

	WARP	**WOCR**
Weight	Approximate rank of v_j	Ratio of v_j
Stop browsing negatives	First	Never
Number of negatives	1000	8 to 256
Mean time per couple	0.1s	0.02s

Table 1: Differences between WARP loss and WOCR loss.

3 Dataset and Experimental Setup

3.1 Dataset

We use a private dataset of job offers as input training for the model. Each job offer is a (*title*, *description*) pair. In this real life dataset, different descriptions can have the same title. We apply linguistic preprocessing techniques such as lemmatisation, POS tagging and stop words removal using the library SpaCy version 2.0.11 [18]. To reduce noise introduced by under-represented tokens in the word embedding space, we discard words that appear less than 10 times. It is worth noticing that the job offers dataset follows the Zipf law: few job titles appear frequently and most of the titles appear only few times. Therefore, discarding non frequent words allows to reduce non frequent titles by re-adjusting them on a more commun form. After this preprocessing, we split the dataset into training and evaluation.

Table 2 summarizes various information about the datasets.

	Training	**Evaluation**
Offers	500 000	60 000
Titles	124 056	25 463
Longest description	343	337
Longest title	13	11

Table 2: Datasets summary. This table displays the number of unique items found in the corresponding dataset.

3.2 Experimental setup

The word2vec-based *job2vec* is built using W_{in}. To obtain a job title vector t, we sum all of its word vector w_i.

$$t = \sum_i w_i$$

The baseline title predictor method consist of a Seq2Seq model with 256 Gated Reccurent Unit and Bahdanau attention [4, 26, 35]. It takes an embedded job description as input and outputs a sequence of word index probabilities.

We train the Seq2Seq model to retrieve the right job title with teacher forcing method. We use cross-entropy loss to optimize index generation. To keep consistency in the comparison, we apply the same preprocessed data and train it on the same machine as JOAPS. In the remainder of this section, we detail hyper-parameters of JOAPS and Seq2Seq baseline method. W_{in} embedding size is set to 300, i.e., $e = 300$. The convolutional layer of the description encoder has filter size of 3 words, 4 words and 5 words with 100 filters each i.e., $f = 300$ and $s = 3, 4, 5$, striding word by word. The number of units in the last dense layer is set to 100 i.e., $h = 100$. The title encoder outputs must also match the output of the description encoder, thus we choose a GRU RNN with 100 units. The total number of trainable parameters in JOAPS is 508 800. The WARP loss margin μ is set to 0.1. To keep the number of possible violating negatives reasonable, batchsize starts at $b = 8$ (56 maximum violating negatives) and is capped at 128 (16 256 maximum of violating negatives). We choose to increase the size of the batch when the ratio of violating negatives is under 0.2 and to decrease it when the ratio of violating negatives is over 0.8. We programm JOAPS and the Seq2Seq baseline method under TensorFlow [2] running on 40 CPUs with 157 Gb RAM. It allows to follow production constraints even for training, which is aligned to our production and industrial constraints in this ever-evolving domain.

3.3 Evaluation tasks

The evaluation of JOAPS is performed on two tasks: job title nearest neighbours and title prediction.

Job title nearest neighbours task. We first evaluate the scenario where HR professionals create opportunities with job titles. The objective is to suggest related jobs based on a job title query: for a person targeting a job, what are the jobs which can be opportunities for him/her. This requires to correctly structure the job titles embedding space.

For this task, we extract from the evaluation dataset job titles never seen during the training phase. We randomly sample 100 job titles for evaluation. For each of these 100 job titles, we asked a pool of 10 HR professionals to evaluate the 10 closest neighbours of both *job2vec* spaces, using the cosine distance as the similarity measure. This qualitative evaluation is done as follows: we display job titles with their closest neighbours using principal component analysis (PCA) with two components on two sides: one side for JOAPS and the other for word2vec, where sides are randomly swapped. We ask the HR professionals to choose the best PCA among left, right, both, or none.

Title prediction task. The second task consists of predicting a job title given a job description. This corresponds to the task where a HR professionnal must label a job opportunity with a consistent job title. For this task, the number of unique job titles (the job title vocabulary) is made of all unique job titles from the testing dataset ($25\ 463$ job titles). The test set for this task is composed of $60\ 000$ never-seen-before job descriptions.

In order to evaluate the prediction made, we proceed as follows. We encode a job description with the description encoder and look for the 10 closest vectors in *job2vec*, using the cosine distance as the similarity measure. We then check if the corresponding job title belongs to the Top#1, Top#5 and Top#10 of closest predicted job titles thereby providing as many accuracy measures.

Nevertheless, as the title encoder projects sequences of words into an embedding space, the predicted sequence may not perfectly match the labelled title. Therefore, we provide two additional metrics for this task. First, to relax the perfect string matching constraint of the previous accuracy metric, we compute the BLEU score [31] for Top#1, Top#5 and Top#10 of closest predicted job titles.

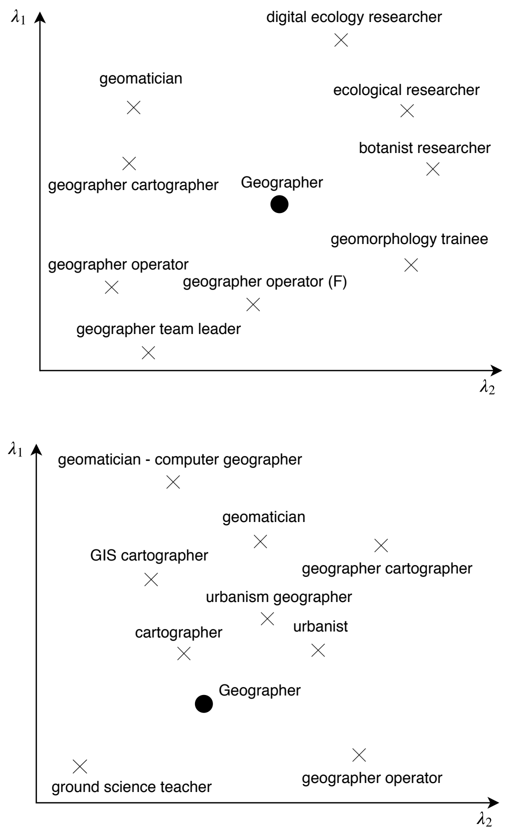

Figure 3: Closest neighbours from JOAPS (up) and word2vec (down) with PCA dimension reduction in *job2vec* spaces for the job *geographer*

Second, we compute a human accuracy score by choosing 100 job descriptions among the ones that got wrong predicted job titles. Then, for each chosen description, we ask our pool of 10 HR professionals to pick the right title among a shuffled set composed of the correct job title and its 10 closest neighbours. Ultimately, as a qualitative analysis, we also extract the CNN filters for JOAPS and the attention of Seq2Seq in order to discuss the differences between extracted features from JOAPS and Seq2Seq.

Parameters and processing time. As we are restricted by the production constraints such as computer memory and processing time, we analyse the number of trainable parameters and the training and inference times.

4 Results and Discussion

Job titles nearest neighbours results. The HR evaluation of the projection of the 100 test job titles yields an excellent preference of 70% for JOAPS, while word2vec-based *job2vec* scored 12.5%.

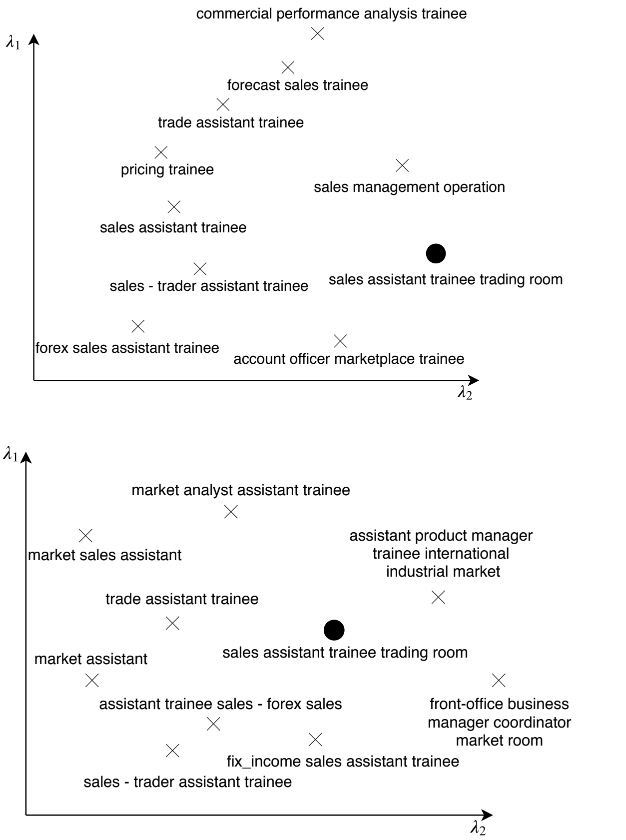

Figure 4: Closest neighbours from JOAPS (up) and word2vec (down) with PCA dimension reduction in the *job2vec* spaces for the job *sales assistant trainee trading room*

This result corroborates the idea that JOAPS *job2vec* clusters job titles correctly even with never-seen-before job titles.

A job title size effect also tends to emerge. Figure 3 and 4 show dimension reduction for examples of two types of job titles: short titles (only one word) on Figure 3 and long titles (more than four words) on Figure 4. The chosen job titles are *geographer* and *assistant trainee trading room*. For short titles, at first sight JOAPS clusters similar jobs together. Indeed we can see that a generic job title like *geographer* is at equal distance to three clusters. One cluster denotes cartography-oriented jobs containing *geographer cartographer, geomatician* and *geographer urbanism*.

A second cluster contains research-oriented jobs such as *ecological researcher* and a last one is composed of jobs for operators. The word2vec-based *job2vec* surrounds the short title job with jobs containing the target job title, or with short job titles already close in the word2vec original embedding space. In contrast to JOAPS, it does not embed the HR key features (such as skills, activities, industries, tools or experience levels) of word combinations. For example, *geographer team leader* is not in the neighbourhood of *geographer* in the word2vec-based*job2vec*. The meaning of *team leader* taking the emphasize over the meaning *geographer*.

For the long title example, it first seems that both *job2vec* are coherent. The word2vec-based method constructs the *job2vec* by summing words vectors, so job titles with multiple common words tend to be close to each other. JOAPS clusters long job titles with a greater diversity. All neighbours are close by the skills, activities, industries, tools or experience levels. Those key features are the ones the HR professionals extract by reading the job title, which is the aim of our work. These results illustrate that the use of a joint loss as WOCR in JOAPS is also able to capture semantic relatedness pulled from job descriptions. The following results use the *job2vec* built by the JOAPS title encoder.

Title prediction results. Table 3 shows accuracy of the title prediction task on the 60 000 never seen job descriptions within *job2vec* (25 463 job titles as shown in Table 2).

Accuracy	**Top#1**	**Top#5**	**Top#10**
JOAPS	29	**55**	**64**
Seq2Seq	**41**	54	58
BLEU	**Top#1**	**Top#5**	**Top#10**
JOAPS	**0.34**	**0.49**	**0.55**
Seq2Seq	0.31	0.41	0.45

Table 3: Accuracies (%) and mean BLEU for the title predictors.

Labelled title	Predicted title
Cartographer draftswoman	FTTH Draftsman FTTH Designer Optic fibre projector
Team leader	Call center manager Call center team manager Call center supervisor
Helicopter mechanic licence part 66 B1-3	Aeronautical sector mechanic Experienced Aeronautical mechanic Plane mechanic

Table 4: First closest neighbours in *job2vec* (25K job titles) for false prediction of job titles.

The JOAPS title predictor finds the right title in the 10 closest neighbours with an accuracy of 29% and Seq2Seq with an accuracy of 41%. However, JOAPS takes over on Top#5 (55% vs 54% resp) and Top#10 (64% vs 58% resp). Notice that the same jobs written in different ways score 0 in accuracy whereas they can have similar words in common (ex: *trade assistant trainee* and *trainee assistant sales*). Furthermore, as shown in Table 4 the *JOAPS* title predictor manages to find more specific titles regarding the description. It also predicts more comprehensive job titles.

That is why we also computed the average maximum BLEU score (Table 3) for both job title predictors (for example *trade assistant trainee* and *trainee assistant sales* have 0.75 BLEU-score).

The JOAPS title predictor achieves the best BLEU score on Top#1, Top#5 and Top#10. Given the 25,463 samples tested, the t-test yields a very significant p-value greater than 0.99 for both evaluations. This assert that averages are different and the differences are significant.

The explanation stands in the training. With its ranking loss, JOAPS is optimized to embed the description such as the output vector is in the neighbourhood of the true title, while Seq2Seq is trained to generate the exact title.

HR professionnals indicate that in 62% cases, the ten first JOAPS predictions suit the description better than the true title, while Seq2Seq performs poorly with 25%.

About models attention on job offers. To deeply understand the description encoder mechanics for both models and their aptitude to act as a job title predictor, we extract CNN max-pooling filters from JOAPS and attention vector from Seq2Seq baseline model, and evaluate which key features are used to predict the job title (Table 5).

Table 5-A and B show two very similar job descriptions which contain target job titles *refrigerator electromechanic* and *geographer*. This kind of job descriptions is one of the main problems in our domain. Indeed, lots of job descriptions from the same company look identical and are difficult to differentiate. Both of them are able to give the right job title, despite their resemblance. Notice that the job title is quoted in the description. The Seq2Seq model attention mainly focuses on those words to predict the correct title while CNN filters also highlight HR key features.

Table 5-C shows a job description which does not contain its job title *Human Resources Apprentice*. In this case JOAPS predicts *HR Mission Leader Apprentice* from *part-time working*, *HR* and *administration*. The Seq2Seq model predits "HR Manager" from the words "HR Manager" but from the wrong context. Seq2Seq focuses only on job titles and does not highlight useful skills to find the job title. This also emphasizes that JOAPS predicts embedded sequences of words in relation with extracted HR key features.

While these experiments demonstrate how JOAPS outperforms a standard Seq2Seq baseline on the two main tasks performed by HR professionals, it is also important to note that it also allows to decrease the training and inference times with respect to that baseline (Table 6).

Trainable parameters	
JOAPS	**505,000**
Seq2Seq	1,200,000
Prediction time for 25k descriptions	
JOAPS	**7m41s**
Seq2Seq	31m52s
Prediction time for 1 description	
JOAPS	**0.02s**
Seq2Seq	0.08s
Time to compute 500k examples	
JOAPS	**7h24m**
Seq2Seq	9h22m

Table 6: Number of trainable parameters, prediction times, and training times for both models.

5 Conclusion

In this paper, we tackled two challenges: create a consistent job embedding space, *job2vec*, in order to compare job titles and retrieve a suitable job title from a job description. To do so, we introduce a solution based on a novel architecture, the first Job Oriented Asymmetrical Pairing System, JOAPS, which jointly constructs a *job2vec* and trains a job title predictor from a description. This architecture is based on asymmetrical sub-neural networks, a title encoder and a description encoder that are connected at their outputs and optimized through an Weighted On-the-fly Chunck Ranking (WOCR) loss. We have evaluated and compared our new architecture on two tasks. JOAPS *job2vec* is compared to a *job2vec* build from a HR oriented word2vec and JOAPS title predictor is compared to a Seq2Seq baseline. Thanks to the joint optimization, JOAPS is able to create a *job2vec* that proved to be consistent with the objective of comparing two jobs titles based on HR key features extracted from job descriptions, such as skills, industries or experience level. In addition, HR professionals found *job2vec* from JOAPS more relevant than the word2vec-based model in 70% cases. For the second challenge, the JOAPS title predictor predicts a job title for a never-seen-before description with an accuracy of 29% for Top#1 up to 64% for Top#10. As sometimes a predicted job title can be correct even if it does not exactly match the same sequence, we also analyze the dataset of unseen job descriptions with an average maximum BLEU score. JOAPS title predictor achieves a BLEU score of 0.34 for Top#1 up to 0.55 for Top#10. To finalize the title predictor analysis, we performed human scoring with the help of HR professionals and highlight that the title predictor is accurate at 50% for Top#5 on 100 false predicted descriptions. Additionally, we discuss the fact that the JOAPS title predictor is able to find a job title more suitable than the given one.

We can also imagine other uses for the Asymmetrical Pairing System to construct any *title2vec* from any input couples (title, description). While we use job titles and job descriptions as training input to construct a *job2vec* with the help of HR key features, one could apply Asymmetrical Pairing System on disease names and diagnoses (resp. dish names and recipes) as training input to construct a *ill2vec* (resp. *dish2vec*) with the help of key features such as symptoms or medicines (reps. cooking methods or ingredients). In our application domain, such a tool can support HR professionals to find bridges between jobs and help them suggest relevant job opportunities to the appropriate employees. According to Dell, 85% of the jobs that will exist in 2030 have not been invented yet [7]. In this near future, a tool like JOAPS will also play an important role helping HR professionals to keep pace with the transformations ahead.

ACKNOWLEDGEMENTS

We would like to thank colleagues who helped us to proofread this paper. A last special thank for our team members at 365Talents in Lyon and Paris. This work was sponsored by 365Talents.

Model	Description
JOAPS	**A)** The Army is made up of 110,000 servicemen and women, serving in regiments, staff and administrative bodies, in mainland France and overseas. Operational commitment is the sole purpose of the Army. The Army recruits over 10,000 soldiers each year, to serve as a soldier through more than 100 jobs in nearly 300 jobs. Every soldier is a fighter, and at the same time a specialist. Qualified or not, you will have to learn the trade of soldier. We are not born as soldiers, we become soldiers. The extraordinary professional career that the Army proposes, revolves around the triptych: recruitment - training - retraining. To engage as: Engaging as Refrigerator Electromechanic M / F is to participate in the smooth running of the soldier's daily by ensuring that the equipment for restoration and cooling works perfectly. Your missions: After initial general training and specialist training, you will be in charge of the installation, maintenance and repair of the cooking equipment, conservation and air conditioning of the Army in the field during exercises or in external operations. Soldier first and foremost, you will develop your technical and tactical know-how and your physical condition in order to perfect yourself and to fulfil the common missions of the Army that can go up to engagement in the fight. Required profile : You are skilled and autonomous? Do you show initiative and have a taste for a job well done? These are qualities that the Army expects from you to be Electromechanical Refrigerator M / F: M / F between 17.5 and 29 years old at most, French nationality, in good standing with French civil rights; Graduate (s) type CAP / BEP or Bac. Initial general training during three months to acquire the bases of the occupation of soldiers. Initial specialist training in regiment. Evolution throughout the course: graduation, acquisition of new qualifications, possibility of reorientation, acquisition of life skills in addition to know-how. Adaptation training or help of the Army to retraining possible as from 4 years. **B)** The Army is made up of 110,000 service men and women, serving in regiments, staffs and administrative bodies, in mainland France and overseas. Operational commitment is the sole purpose of the Army. The Army recruits more than 10,000 soldiers each year to work as a soldier through more than 100 jobs in nearly 300 jobs. Every soldier is a fighter, at the same time a specialist. Qualified or not, you will have to learn the trade of soldier. We are not born a soldier, we become one. The extraordinary professional career that the Army proposes, revolves around the triptych: recruitment - training - retraining. To engage as: To become a Geographer M / F is to carry out topographic surveys in the field and to work on maps that will enable military operations to be carried out. Your missions: The topographer must carry out the measurements and plans in the field, at all times and sometimes in time limits, which will allow the constitution of a cartographic document. The cartographer who executes paper drawings or computer-assisted drawings from topographer surveys. In both cases, you mainly work on a computer and often go outside. Soldier first and foremost, you will develop your technical and tactical know-how and your physical condition in order to perfect yourself and to fulfil the common missions of the Army that can go up to the engagement in the fight. Required profile : Do you have a taste for new technologies? You are dynamic, methodical and rigorous? These are qualities that the Army expects from you to be a Geographer: M / F between 17.5 and 29 years old at most, French nationality, in good standing with French civil rights. Graduate (s) type CAP / BEP or Bac appreciated (but not mandatory). Type of contract / status / remuneration: Training / evolution / reconversion: Initial general training of three months to acquire the bases of the occupation of soldier. The geographer is assigned to the 28th geographic group, but may have to join regiments in France or overseas during exercises or external operations. **C)** Graduated with a bac+2 you wish to continue your training in human resources with a diploma certified by the state with a periodic training. As a member of a subsidiary of a large group, your mission will be to participate in the implementation of a plan training, needs gathering, setting up opca relationship files, creating a GPEC, writing a business guide, skills referential, mobility assessment, recruitment, sourcing, administration, job board. Pre-qualification by telephone interview with a HR manager, good written and oral expression required, teamwork skills, discretion, adaptability, rigour and organisation.
Seq2Seq	**A)** The Army is made up of 110,000 servicemen and women, serving in regiments, staff and administrative bodies, in mainland France and overseas. Operational commitment is the sole purpose of the Army. The Army recruits over 10,000 soldiers each year, to serve as a soldier through more than 100 jobs in nearly 300 jobs. Every soldier is a fighter, and at the same time a specialist. Qualified or not, you will have to learn the trade of soldier. We are not born as soldiers, we become soldiers. The extraordinary professional career that the Army proposes, revolves around the triptych: recruitment - training - retraining. To engage as: Engaging as Refrigerator Electromechanic M / F is to participate in the smooth running of the soldier's daily by ensuring that the equipment for restoration and cooling works perfectly. Your missions: After initial general training and specialist training, you will be in charge of the installation, maintenance and repair of the cooking equipment, conservation and air conditioning of the Army in the field during exercises or in external operations. Soldier first and foremost, you will develop your technical and tactical know-how and your physical condition in order to perfect yourself and to fulfil the common missions of the Army that can go up to engagement in the fight. Required profile : You are skilled and autonomous? Do you show initiative and have a taste for a job well done? These are qualities that the Army expects from you to be Electromechanical Refrigerator M / F: M / F between 17.5 and 29 years old at most, French nationality, in good standing with French civil rights; Graduate (s) type CAP / BEP or Bac. Initial general training during three months to acquire the bases of the occupation of soldiers. Initial specialist training in regiment. Evolution throughout the course: graduation, acquisition of new qualifications, possibility of reorientation, acquisition of life skills in addition to know-how. Adaptation training or help of the Army to retraining possible as from 4 years. **B)** The Army is made up of 110,000 service men and women, serving in regiments, staffs and administrative bodies, in mainland France and overseas. Operational commitment is the sole purpose of the Army. The Army recruits more than 10,000 soldiers each year to work as a soldier through more than 100 jobs in nearly 300 jobs. Every soldier is a fighter, at the same time a specialist. Qualified or not, you will have to learn the trade of soldier. We are not born a soldier, we become one. The extraordinary professional career that the Army proposes, revolves around the triptych: recruitment - training - retraining. To engage as: To become a Geographer M / F is to carry out topographic surveys in the field and to work on maps that will enable military operations to be carried out. Your missions: The topographer must carry out the measurements and plans in the field, at all times and sometimes in time limits, which will allow the constitution of a cartographic document. The cartographer who executes paper drawings or computer-assisted drawings from topographer surveys. In both cases, you mainly work on a computer and often go outside. Soldier first and foremost, you will develop your technical and tactical know-how and your physical condition in order to perfect yourself and to fulfil the common missions of the Army that can go up to the engagement in the fight. Required profile : Do you have a taste for new technologies? You are dynamic, methodical and rigorous? These are qualities that the Army expects from you to be a Geographer: M / F between 17.5 and 29 years old at most, French nationality, in good standing with French civil rights. Graduate (s) type CAP / BEP or Bac appreciated (but not mandatory). Type of contract / status / remuneration: Training / evolution / reconversion: Initial general training of three months to acquire the bases of the occupation of soldier. The geographer is assigned to the 28th geographic group, but may have to join regiments in France or overseas during exercises or external operations. **C)** Graduated with a bac+2 you wish to continue your training in human resources with a diploma certified by the state with a periodic training. As a member of a subsidiary of a large group, your mission will be to participate in the implementation of a plan training, needs gathering, setting up opca relationship files, creating a GPEC, writing a business guide, skills referential, mobility assessment, recruitment, sourcing, administration, job board. Pre-qualification by telephone interview with a HR manager, good written and oral expression required, teamwork skills, discretion, adaptability, rigour and organisation.

Table 5: Relevant words as detected by the CNN filters and Attention while predicting job title A) *Refrigerator Electromechanic*, B) *geographer*, C) *Human Ressources Manager*. Words detected as relevant are highlighted; The darker the highlight, the more important the word is.

REFERENCES

[1] Report on text classification using cnn, rnn han. https://medium.com/jatana/report-on-text-classification-using-cnn-rnn-han-f0e887214d5f. Accessed: 2018-07-17.

[2] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, and Zhifeng Chen et al. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[3] Maurya Abhinav, 'Understanding Job-Skill Relationships using Big Data and Neural Networks', Bloomberg Data for Good Exchange Conference, (2018).

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

[5] Marco Baroni, Georgiana Dinu, and Germán Kruszewski, 'Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors', in Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 238–247, Baltimore, Maryland, (June 2014). Association for Computational Linguistics.

[6] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov, 'Enriching word vectors with subword information', arXiv preprint arXiv:1607.04606, (2016).

[7] Ian Bork, Ming-Li Chai, Devin Fidler, Eri Gentry, and Deborah Johnson et al., Emerging Technologies Impact on Society and Work in 2030, Jul 2017.

[8] Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah, 'Signature verification using a" siamese" time delay neural network', in Advances in Neural Information Processing Systems, pp. 737–744, (1994).

[9] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio, 'Learning phrase representations using rnn encoder-decoder for statistical machine translation', arXiv preprint arXiv:1406.1078, (2014).

[10] Ronan Collobert and Jason Weston, 'A unified architecture for natural language processing: Deep neural networks with multitask learning', in Proceedings of the 25th international conference on Machine learning, pp. 160–167. ACM, (2008).

[11] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa, 'Natural language processing (almost) from scratch', Journal of Machine Learning Research, **12**(Aug), 2493–2537, (2011).

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, 'Bert: Pre-training of deep bidirectional transformers for language understanding', arXiv preprint arXiv:1810.04805, (2018).

[13] Charles-Emmanuel Dias, Vincent Guigue, and Patrick Gallinari, 'Passé, présent, futurs: induction de carrières professionnelles à partir de cv.', in CORIA, pp. 281–296, (2017).

[14] Jeffrey L Elman, 'Finding structure in time', Cognitive science, **14**(2), 179–211, (1990).

[15] Alex Graves, 'Generating sequences with recurrent neural networks', arXiv preprint arXiv:1308.0850, (2013).

[16] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton, 'Speech recognition with deep recurrent neural networks', in Acoustics, speech and signal processing (icassp), 2013 ieee international conference on, pp. 6645–6649. IEEE, (2013).

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

[18] Matthew Honnibal and Ines Montani, 'spacy 2: Natural language understanding with bloom embeddings, convolutional neural networks and incremental parsing', To appear, (2017).

[19] L. C. Jain and L. R. Medsker, Recurrent Neural Networks: Design and Applications, CRC Press, Inc., Boca Raton, FL, USA, 1st edn., 1999.

[20] Faizan Javed, Qinlong Luo, Matt McNair, and Jacob Ferosh et al., 'Carotene: A job title classification system for the online recruitment domain', 286–293, (2015).

[21] Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom, 'A convolutional neural network for modelling sentences', arXiv preprint arXiv:1404.2188, (2014).

[22] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei, 'Large-scale video classification with convolutional neural networks', in Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, pp. 1725–1732, (2014).

[23] Yoon Kim, 'Convolutional neural networks for sentence classification', arXiv preprint arXiv:1408.5882, (2014).

[24] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, 'Imagenet classification with deep convolutional neural networks', in Advances in neural information processing systems, pp. 1097–1105, (2012).

[25] Konstantin Lopyrev, 'Generating news headlines with recurrent neural networks', arXiv preprint arXiv:1512.01712, (2015).

[26] Thang Luong, Hieu Pham, and Christopher D. Manning, 'Effective approaches to attention-based neural machine translation', in Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp. 1412–1421, Lisbon, Portugal, (September 2015). Association for Computational Linguistics.

[27] Emmanuel Malherbe, Mario Cataldi, and Andrea Ballatore, 'Bringing order to the job market: Efficient job offer categorization in e-recruitment', in Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, (2015).

[28] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean, 'Efficient estimation of word representations in vector space', arXiv preprint arXiv:1301.3781, (2013).

[29] Tomáš Mikolov, Martin Karafiát, Lukáš Burget, Jan Černockỳ, and Sanjeev Khudanpur, 'Recurrent neural network based language model', in Eleventh Annual Conference of the International Speech Communication Association, (2010).

[30] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean, 'Distributed representations of words and phrases and their compositionality', in Advances in neural information processing systems, pp. 3111–3119, (2013).

[31] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu, 'Bleu: A method for automatic evaluation of machine translation', in Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, ACL '02, pp. 311–318, Stroudsburg, PA, USA, (2002). Association for Computational Linguistics.

[32] Jeffrey Pennington, Richard Socher, and Christopher D. Manning, 'Glove: Global vectors for word representation', in Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543, (2014).

[33] Karen Simonyan and Andrew Zisserman, 'Very deep convolutional networks for large-scale image recognition', arXiv preprint arXiv:1409.1556, (2014).

[34] Ilya Sutskever, James Martens, and Geoffrey E Hinton, 'Generating text with recurrent neural networks', in Proceedings of the 28th International Conference on Machine Learning (ICML-11), pp. 1017–1024, (2011).

[35] Ilya Sutskever, Oriol Vinyals, and Quoc V Le, 'Sequence to sequence learning with neural networks', in Advances in neural information processing systems, pp. 3104–3112, (2014).

[36] Nicolas Usunier, David Buffoni, and Patrick Gallinari, 'Ranking with ordered weighted pairwise classification', in ICML, pp. 1057–1064, (2009).

[37] Jason Weston, Samy Bengio, and Nicolas Usunier, 'Wsabie: Scaling up to large vocabulary image annotation', in IJCAI, volume 11, pp. 2764–2770, (2011).

[38] Jason Weston, Sumit Chopra, and Keith Adams, '# tagspace: Semantic embeddings from hashtags', in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1822–1827, (2014).

[39] Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, and Jun Zhao, 'Relation classification via convolutional deep neural network', in Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, pp. 2335–2344, (2014).

[40] Denghui Zhang, Junming Liu, Hengshu Zhu, Yanchi Liu, Lichen Wang, Pengyang Wang, and Hui Xiong, 'Job2vec: Job title benchmarking with collective multi-view representation learning', in Proceedings of the 28th ACM International Conference on Information and Knowledge Management, CIKM '19, pp. 2763–2771, New York, NY, USA, (2019). ACM.

[41] Yun Zhu, Faizan Javed, and Ozgur Ozturk, 'Semantic similarity strategies for job title classification', arXiv preprint arXiv:1609.06268, (2016).

[42] Yun Zhu, Faizan Javed, and Ozgur Ozturk, 'Document embedding strategies for job title classification', (2017).