Word Embeddings vs Word Types for Sequence Labeling: the Curious Case of CV Parsing

Melanie Tosik[†] **Carsten L. Hansen** [‡] **Gerard Goossen** [‡] **Mihai Rotaru** [‡]

[†]Department of Linguistics
University of Potsdam
tosik@uni-potsdam.de

[‡]Textkernel B.V.
Amsterdam, Netherlands
{hansen,goossen,rotaru}@textkernel.nl

Abstract

We explore new methods of improving Curriculum Vitæ (CV) parsing for German documents by applying recent research on the application of word embeddings in Natural Language Processing (NLP). Our approach integrates the word embeddings as input features for a probabilistic sequence labeling model that relies on the Conditional Random Field (CRF) framework. Best-performing word embeddings are generated from a large sample of German CVs. The best results on the extraction task are obtained by the model which integrates the word embeddings together with a number of hand-crafted features. The improvements are consistent throughout different sections of the target documents. The effect of the word embeddings is strongest on semi-structured, out-of-sample data.

1 Introduction

Curriculum Vitæ (CV) parsing refers to the task of processing and transforming the relevant information contained in a given CV. The goal is to produce structured output detailing the information presented in the document, including personal information, education items, work experience, or further skills.

CV Parsing is used in multiple real world scenarios. Nowadays, job seekers are frequently presented with the option of simply uploading the required documents into an application system, which then automatically processes the data and directly uploads the candidate information into the corresponding databases. Given structured information on the candidate, recruiters are able to quickly search for potential matches, and systems are enabled to generate personalized recommendations that meet the candidate's specific skill set.

CV Parsing poses an interesting challenge to modern Natural Language Processing (NLP) techniques, because the documents consist of a mixture of semi-structured and free form text with a high degree of variance in the data. The semi-structured text often takes the shape of attribute-value pairs. Typical examples with regard to personal information would be

Name: John Doe, or
Phone: 212 / 123-5678.

A considerable portion of CVs contain personal information without any left context, e.g.

John Doe, MD
900 Main Street
New York, NY

Free form text is most often encountered in work experience items, such as

2006–2008 *Software Developer*
Eastman Kodak Company
Rochester, NY

Led a small team, investigated current systems, and created applications.

High variance in the data stems from the fact that we are dealing with CVs from all possible industries and locations, and on any possible skill level. As a result, there will always be unknown words in the

Proceedings of NAACL-HLT 2015, pages 123–128,
Denver, Colorado, May 31 – June 5, 2015.

entities we seek to extract, most commonly when processing names, addresses, jobs, or companies.

While CV Parsing combines many different NLP components, in this paper we will focus on one task in particular: the extraction of two different types of entities from pre-segmented sections, namely the section containing the personal information of the applicant, and his or her work related information.

More precisely, we investigate the contribution of word embeddings versus word type (or *one-hot*) representations as input feature for a sequence labeling model based on Conditional Random Fields (CRF). By using word embeddings instead of word types, the model is able to utilize large amounts of unlabeled data to supplement the supervised training.

We show that using word embeddings as additional input feature to the CRF model greatly improves the overall model performance. Word embeddings also enhance model performance on out-of-sample data, since the model no longer relies on only the fixed observations in the training data.

2 Related work

To our knowledge, the availability of prior research on CV parsing is very limited. Yu et al. (2005) design a cascaded Information Extraction (IE) framework for CV extraction, comparing flat models based on Hidden Markov Models (HMM) and Support Vector Machines (SVM) with a hierarchical hybrid model.

Over the past decade, there has been an increasing research interest in the application of word embeddings to complex tasks in language processing. As input features for different CRF models, word embeddings are already effectively used in a wide range of NLP systems, including Named Entity Recognition (Demir and Ozgur, 2014), chunking and Part-of-Speech Tagging (Huang and Yates, 2009). Turian et al. (2010) evaluate different techniques for inducing word representations and detail significant improvements for supervised NER and chunking systems when also incorporating word embeddings. Wang and Manning (2013) suggest that linear model architectures benefit from a high-dimensional, discrete feature space. Guo et al. (2014) investigate different approaches on transforming skip-gram embeddings (Mikolov et al., 2013) correspondingly, and report higher performance than directly using the word embeddings with supervised NER as evaluation task. We extend previous work by exploring a novel task in NER, as well as directly comparing the effect of using word types versus word embeddings and how this affects the robustness of the model.

3 Task

As indicated above, our task is to extract structured information from the personal and the experience sections of a diverse set of German input CVs. We solve this extraction task by treating it as a conventional NER problem. Unlike most previous NER work that focuses on extracting the standard name/organization/location/other entities, our domain has an extended set of entities.

For personal information, we extract 6 different entities, specifically the full name of the candidate, the contact address, birthday, phone number, nationality, and email address. From the work experience section, we extract 3 entities, namely the job title, job duration, as well as the company and location. Since experience sections of CVs usually contain multiple previous job descriptions, the task is to extract the given information for each of these jobs.

4 Methodology

We first discuss word embeddings in Section 4.1, before we move on to a formal description of the CRF architecture in Section 4.2.

4.1 Word Embeddings

Word embeddings are continuous vector representations induced from unlabeled input text of arbitrary length. Each dimension of the word embedding represents a latent feature of the word. Intuitively, this kind of meaning representation captures useful properties of the word, both semantically and syntactically (Mikolov et al., 2013).

Word embeddings are typically learned using neural networks (Collobert and Weston, 2008) or clustering as underlying predictive model. Turian et al. (2010) provide a comparison of multiple approaches. Recently, Mikolov et al. (2013) proposed a simple and computationally efficient way to learn word embeddings. In the skip-gram model architecture, the hidden layer is replaced by a shared pro-

jection layer, and a window of size c surrounding words $w_{t-c}, .., w_{t-1}, w_{t+1}, .., w_{t+c}$ from word w_t is predicted. The training objective is to learn word embeddings which are good predictors for the surrounding words. This is done by maximizing the average log probability over the data:

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\leq j\leq c, j\neq 0} log\ p(w_{t+j}|w_t)$$

In order to avoid a costly computation proportional to the size of the vocabulary, $p(w_{t+j}|w_t)$ is computed using the hierarchical softmax function as an approximation of the softmax functions. Increasing the window size c can improve accuracy at the expense of training time, since it results in more training examples.

4.2 Conditional Random Fields

The Conditional Random Fields (CRF) model is a state-of-the-art sequence labeling method first introduced by Lafferty et al. (2001).

CRFs are a undirected, graphical model trained to maximize a conditional probability distribution given a set of features. The most common graphical structure used with CRF is linear chain. Let $Y = (y_1, ..., y_T)$ denote a sequence of labels and $X = (x_1, ..., x_T)$ denote the corresponding observations sequence. The sequence of labels is the concept we wish to predict (e.g. target phrases, named-entity, POS, etc.). The observations are the words in the input string. Given a linear chain CRF, the conditional probability $p(Y|X)$ is computed as follows:

$$p(Y|X) = \frac{1}{Z_X}\prod_{t=1}^{T} exp\left\{\sum_{k=1}^{K}\lambda_k f_k(y_t, y_{t-1}, x_t)\right\}$$

Z_X is a normalizing constant such that all the terms normalize to one, f_k is a feature function, and λ_k is a feature weight. CRF offers an advantage over generative approaches by relaxing the conditional independence assumption and allowing for arbitrary features in the observation.

For all our experiments we use *CRFsuite*[1], an implementation of CRF for labeling sequential data provided by Okazaki (2007). We choose an appropriate learning algorithm based on accuracy on the development set and use Limited-memory BFGS optimization (Nocedal, 1980).

[1] http://www.chokkan.org/software/crfsuite/

5 Experimental setup

We start by describing our data sets in Section 5.1. Section 5.2 details the feature set implemented in the models. Section 5.3 provides details on the generating of the word embeddings, and Section 5.4 specifies the model evaluation.

5.1 Data

We use two separate data sets for evaluation: the main set, and an additional out-of-sample set. Table 1 provides an overview of the number of documents and section-specific entities contained in the main set and the out-of-sample dataset.

	Main set			OOS
	Train	Dev	Test	Test
#Docs	1010	233	214	25
#Pers	6736	1634	1388	n/a
#Exp	20687	4569	4410	356

Table 1: Distribution of documents and personal and experience entities over main set and out-of-sample (OOS) dataset.

In total, the main set is comprised of a sample of 1457 annotated German documents. This sample was randomly split into training (1010 documents), development (233 documents), and test partition (214 documents). All sequence labeling models are trained on the training partition. The test partition is used to evaluate the model performance of previously unseen but similar data.

In addition, we evaluate the performance of the same model on an out-of-sample dataset. This is done in order to measure how well the model generalizes to unseen data from an inherently different sample, i.e. CVs from a new domain not included in the original sample. The out-of-sample set is comprised of a sample of 25 annotated German CVs. These documents are only annotated for experience

entities, but since each document contains an average of 4.4 work experience items, it provides us with approximately 115 examples of each entity.

Personal information entities usually occur only once per document. Experience entities occur at most once per work experience item. Each document contains 5.9 work experience items on average.

5.2 Features

As indicated in Section 4.2, the CRF model learns based on a number of predefined features. The hand-crafted features include mostly simple orthographic features that account for the beginning and end of a line, unknown words, digits, single characters, multi spaces, capitalization, as well as the first and last token of each line. In addition, high frequency token features encode the 200 most frequent words in the training data in a one-hot binary vector. This is done separately for personal and work experience section.

Similarly, we implement a one-hot representation of word types incorporating all tokens that occur at least twice in the training data. Most importantly, we also implement word embeddings of any given word type as one feature per dimension. We use a BIO encoding (Ramshaw and Marcus, 1995) for labels, resulting in 13 labels for the personal section, and 7 labels for the experience section. Each label spans entire tokens.

5.3 Generating Word Embeddings

To generate the word embeddings, we use the open source *word2vec*[2] toolkit. We conduct a number of experiments to determine most suitable parameters settings. We tune the number of latent dimensions on the development set and find 150 dimensions to give us the best results. Except for vector size, we use default parameters. Applying the skip-gram architecture has proven to be robust across trials.

We experiment with various data sources, including the German Wikipedia, different batches of sample CVs, and spidered job postings. Overall best word embeddings for the information extraction tasks are generated from a set of 200K German sample CVs containing approximately 145.5M tokens.

[2] https://code.google.com/p/word2vec/

5.4 Evaluation

We evaluate five models based on three different groups of features (cf. Table 2). The first baseline model uses only the hand-crafted features. We compare this baseline to two models which incorporate either a feature vector for the word types, or a feature vector for the word embeddings, respectively. Finally, we combine the hand-crafted features with word types or word embeddings for two additional models.

Character-based overlap scores are computed for averaged precision, recall, and F1 scores to evaluate the performance of the models on personal and experience sections. We use character-based overlap instead of token-based overlap scores to penalize the incorrect labeling of longer tokens. Recall that our labeling always spans entire tokens.

6 Results

The macro-averaged precision, recall, and F1 scores for the entities in the personal and experience sections, for the different phrase models on the main test partition, are shown in Table 2.

For the personal section, the models using only word types or only word embeddings give the lowest performance. This is due to the fact that personal sections have a semi-structured layout and content words, which are already well captured when using the hand-crafted orthographic features together with the high frequency token feature.

On less structured experience sections, the effect of the word embeddings is much stronger. By using only word embeddings as features for the model, we outperform the hand-crafted feature baseline by 3.9% on average. The best performance is achieved by combining word embeddings and hand-crafted features, resulting in 96.0% averaged F1 score on the personal section, and 84.0% F1 score on the experience section.

We compare the performance of experience entity extraction on the main test partition with its performance on the out-of-sample data. The results are presented in Table 3. Since word embeddings are learned from large amounts of unlabeled data, we verify that word embeddings also enhance the model performance on the out-of-sample data. Indeed, we observe a 10.1% increase in recall on the out-of-

Model	Personal section			[%]	Experience section		
	Prec (avg.)	Rec (avg.)	F1 (avg.)		Prec (avg.)	Rec (avg.)	F1 (avg.)
Hand-crafted features	94.5	94.0	94.3		84.7	69.8	76.4
Word types	94.7	91.2	92.3		85.3	67.7	75.3
Word embeddings	94.9	93.1	93.9		87.0	74.6	80.3
Word types + features	95.2	95.0	95.1		88.4	74.3	80.6
Word embeddings + features	96.3	95.7	96.0		89.6	79.2	84.0

Table 2: Macro-averaged precision, recall, and F1 scores of the phrase models on the main test partition.

Model	Test set			[%]	Out-of-sample set		
	Prec (avg.)	Rec (avg.)	F1 (avg.)		Prec (avg.)	Rec (avg.)	F1 (avg.)
Word types + features	88.4	74.3	80.6		82.3	57.0	65.6
Word embeddings + features	89.6	79.2	84.0		83.3	67.1	73.8

Table 3: Experience phrase model performance on test partition and out-of-sample dataset.

sample data when using word embeddings instead of word types. Using word embeddings also leads to a greater improvement in F1 score on the out-of-sample set (+8.2%) compared with the main test partition (+3.5%). This suggests that the word embeddings increase the robustness of the model towards the lexical variety comprised in CVs from additional industries.

The results support our hypothesis that the observed improvements are mostly due to the Distributional Hypothesis (Firth, 1957), and the enhanced handling of out-of-vocabulary words: by using word embeddings rather than one-hot representations, the models are able to more accurately predict labels on words that did not occur in the training data.

7 Future Work

Based on the limited sets of sample documents at hand, we currently learn word embeddings from much less data than has been suggested in previous related work. Thus, we are planning on investigating the impact of data source and amount of data for word embedding generation.

Since the focus of the work presented was on German documents, we would additionally like to verify that the results generalize to other languages. First initial test runs on Portuguese indicate that similar improvements can be reproduced easily.

It would also be interesting to move beyond the CRF architecture by comparing performances of different sequence labeling methods on the given task.

8 Conclusion

We describe how word embeddings can be successfully applied to the task of CV parsing. Using the skip-gram architecture, we learn word embeddings from a large set of unlabeled German CVs, and implement them as additional feature to our CRF based sequence labeling model.

Results on the personal section show that neither word types, nor word embeddings alone perform well enough to beat the baseline model based on hand-crafted features only. When combining word types or word embeddings with the hand-crafted features, word embeddings outperform the word types.

We observe that the improvements from the word embeddings combined with hand-crafted features carry over to semi-structured and free form work experience text. Applying word embeddings together with hand-crafted features additionally greatly improves the performance on an out-of-sample dataset.

Acknowledgments

We are thankful to Stephen Roller and the anonymous reviewers for their helpful comments and suggestions. We acknowledge valuable discussions with other researchers at Textkernel: Florence Berbain, Lena Bayeva, Chao Li, and Jakub Zavrel.

References

Collobert, R. and J. Weston (2008). A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, New York, NY, USA, pp. 160–167. ACM.

Demir, H. and A. Ozgur (2014). Improving Named Entity Recognition for Morphologically Rich Languages Using Word Embeddings. In *13th International Conference on Machine Learning and Applications, ICMLA 2014, Detroit, MI, USA, December 3-6, 2014*, pp. 117–122.

Firth, J. (1957). A synopsis of linguistic theory 1930-1955. pp. 1–32.

Guo, J., W. Che, H. Wang, and T. Liu (2014). Revisiting Embedding Features for Simple Semi-supervised Learning. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pp. 110–120.

Huang, F. and A. Yates (2009). Distributional Representations for Handling Sparsity in Supervised Sequence-labeling. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1 - Volume 1*, ACL '09, Stroudsburg, PA, USA, pp. 495–503. Association for Computational Linguistics.

Lafferty, J. D., A. McCallum, and F. C. N. Pereira (2001). Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, San Francisco, CA, USA, pp. 282–289. Morgan Kaufmann Publishers Inc.

Mikolov, T., K. Chen, G. Corrado, and J. Dean (2013). Efficient Estimation of Word Representations in Vector Space. *CoRR abs/1301.3781*.

Mikolov, T., W. Yih, and G. Zweig (2013). Linguistic Regularities in Continuous Space Word Representations. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 9-14, 2013, Westin Peachtree Plaza Hotel, Atlanta, Georgia, USA*, pp. 746–751.

Nocedal, J. (1980). Updating quasi-newton matrices with limited storage. *Mathematics of computation 35*(151), 773–782.

Okazaki, N. (2007). CRFsuite: a fast implementation of Conditional Random Fields (CRFs).

Ramshaw, L. A. and M. P. Marcus (1995). Text Chunking using Transformation-Based Learning. *CoRR cmp-lg/9505040*.

Turian, J. P., L. Ratinov, and Y. Bengio (2010). Word Representations: A Simple and General Method for Semi-Supervised Learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, July 11-16, 2010, Uppsala, Sweden*, pp. 384–394.

Wang, M. and D. C. Manning (2013). Effect of Non-linear Deep Architecture in Sequence Labeling. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pp. 1285–1291. Asian Federation of Natural Language Processing.

Yu, K., G. Guan, and M. Zhou (2005). Resume Information Extraction with Cascaded Hybrid Model. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, Stroudsburg, PA, USA, pp. 499–506. Association for Computational Linguistics.