Using RobBERT and eXtreme Multi-Label Classification to Extract Implicit and Explicit Skills From Dutch Job Descriptions

NINANDE VERMEER, University of Amsterdam, Netherlands
VERA PROVATOROVA, University of Amsterdam, Netherlands
DAVID GRAUS, Randstad Groep Nederland, Netherlands
THILINA RAJAPAKSE, University of Amsterdam, Netherlands
SEPIDEH MESBAH, Randstad Groep Nederland, Netherlands

Randstad processes large amounts of vacancies and resumes to find matches between jobs and people. An important aspect of this process is skill extraction, since overlap in skills between vacancies and resumes could indicate a match. However, skill-extraction, especially when the skills are not explicitly mentioned in the text (i.e., implicit skills), is a challenging task. While ontology-based techniques show promising results for extracting implicit skills, they are expensive to create and maintain for different languages. A more recent research exploits the context of the texts with Bidirectional Encoder Representations from Transformers (BERT) and eXtreme Multi-Label Classification (XMLC) to tackle the problem of explicit and implicit skill extraction from English-language vacancies. Inspired by the aforementioned technique, in this project we explore a Dutch-language alternative: RobBERT-XMLC with the recently developed pre-trained RobBERT. The aim of this project is therefore to answer the following research question: To what extent can a RobBERT-XMLC model be used to extract both explicit and implicit skills from Dutch job descriptions? The evaluation results demonstrate the effectiveness of our approach (i.e., Recall@100 83.12, nDCG@100 73.97 and MRR 90.53) in extracting skills from vacancies.

Additional Key Words and Phrases: BERT, RobBERT, XMLC, Skill-extraction, Implicit skills

ACM Reference Format:
Ninande Vermeer, Vera Provatorova, David Graus, Thilina Rajapakse, and Sepideh Mesbah . 2022. Using RobBERT and eXtreme Multi-Label Classification to Extract Implicit and Explicit Skills From Dutch Job Descriptions. In *compjobs '22: Computational Jobs Marketplace, Feb 25, 2022.* ACM, virtual, 5 pages.

1 INTRODUCTION

Randstad[1] processes large amounts of vacancies and resumes to find matches between jobs and people. An important aspect of this process is the extraction of skills from those documents, since a similarity in skills of a vacancy and a resume could indicate a match. Skill extraction is a challenge in general, but it is even more difficult when skills are not explicitly mentioned in the text. These "implicit skills" are relevant for the job position but are not explicitly mentioned in the text. For example, if a job description states that a suitable candidate needs to have experience in analyzing and modelling unstructured data, this implicitly means that the talent is required to have knowledge in the machine learning area.

While skill-extraction has been a focus of several recent studies ([2, 5, 10, 12]), extracting implicit skills remains an under-explored topic. The most common approach to implicit skills extraction from job descriptions relies on ontologies [3], which are expensive to create and maintain. More recently, researchers have started to investigate hybrid techniques, such as combining dictionary-based methods with Named Entity Recognition (NER), Word2vec and

[1] https://www.randstad.nl/

2022. Manuscript submitted to ACM

Doc2vec [8] to extract explicit and implicit skill sets from a job description. Bhola et al. [4] on the other hand developed a context-based approach to extract both implicit and explicit skills that shows promising results. They created the BERT-XMLC architecture, a model that combines the BERT language model [7] with eXtreme Multi-Label Classification (XMLC).

Our goal is to develop a context-based skill extraction method that only exploits the context of the text in order to extract skills from vacancy texts. In this way our technique can be easily extended to new languages. Based on this intuition and inspired by Bhola et al. [4], we approach our problem as a context-dependent eXtreme multi-label text classification task. Unlike the English-language vacancies used in [4], the vacancies of Randstad are in Dutch. Therefore, this study aims at investigating the effectiveness of using RobBERT, a Dutch BERT model presented in 2020 [6], and answering the research question: *To what extent can a RobBERT-XMLC model be used to extract both explicit and implicit skills from Dutch job descriptions?* We make the following contributions: (1) we introduce a technique which combines RobBERT with XMLC to extract explicit and implicit skills from vacancy texts; (2) we make the source code publicly available which can be easily extended to different languages and any other multi-label classification task[2]; (3) we provide a detailed analysis of the performance of our model using English and Dutch datasets and different model setups.

2 METHOD

Given a job description and a large set of skills, our goal is to calculate the probability of each skill being relevant to the job description. First, the words of the job descriptions are encoded by a pre-trained BERT model. The encoding of the [CLS] token is then used as representation of the job description. For the Dutch language several monolingual BERT-based language models are available: BERT-NL, BERTje and RobBERT. Since RobBERT outperformed other BERT-like models on a variety of Dutch language tasks [6], we turn to RobBERT. Next, we process the representations of job descriptions by a bottleneck layer (i.e., an added linear layer before the output layer, like in [11]) to prevent overfitting [4]. The last layer treats every skill as a binary classification problem, so for each skill it calculates the probability that the skill is associated with the job description. Since there are many skill labels, 3,789 to be precise, this is considered an XMLC task.

2.1 Data Selection and Pre-processing

Our dataset contains 20,000 vacancies which were scraped from the internet by Jobdigger,[3] and the skills labeled by Textkernel[4]. Each vacancy consists of a job description, **explicit skills** extracted by Textkernel (T), and a "JDCO-code." JDCO stands for "Jobdigger Classification of Occupations" and indicates to which of the 3,800 occupation classes the job role of the vacancy belongs [9]. There is a list of most relevant skills associated with each JDCO code (J). Since our dataset does not include implicit skills, and extracting implicit skills for the training data is time-consuming and expensive, we consider the skills in ($T_c \cup J$) as **implicit skills** and add those to the explicit skills to obtain our complete skill list. Note that we cannot claim that all the implicit skills are indeed relevant to each vacancy with the same JDCO code, and our implicit skill list can be noisy. There are, on average, 20.61 skills per vacancy including implicit skills and 7.13 skills per vacancy excluding implicit skills. We pre-process job descriptions by removing code (e.g. HTML and JavaScript), special characters, and stopwords. Since RobBERT is case-sensitive, we do not lower-case the job descriptions. The average length of job descriptions is 202 words after pre-processing.

[2] Source code available at: https://github.com/NinandeVermeer/MasterThesis

[3] https://www.jobdigger.nl/

[4] https://www.textkernel.com/nl/

The dataset is split into a training (80%), validation (10%) and test set (10%), and the skill labels are binary encoded in a $1 \times S$-vector, where S is the number of unique skills. The tokenization is done within the model itself.

2.2 Implementation

We run our experiments in SageMaker Notebooks on Amazon Web Services on a `ml.p2.xlarge` GPU. The models are implemented using Simple Transformers[5]. We re-use the same hyperparameter values as Bhola et al. [4]. Furthermore, the models also use the output of the `[CLS]` token as representation of the job description, the AdamW optimizer and BCE with logits loss. Additionally, two (linear) layers are added of size 786 and 2,000. We only fine-tune these layers during the experiments, the rest of the architecture is frozen.

2.3 Evaluation

The evaluation is done similarly to Bhola et al. [4]. The model predicts for each skill label the probability of being relevant to the job description. We rank the predictions in descending order, and calculate MRR, Recall and nDCG based on the list of true labels. For Recall and nDCG, the denominator of the equation is based on the total ordered list of skills, and the nominator is based on the first k skills. Here k can be 5, 10, 30, 50 or 100. Consequently, the metric values should increase when k increases. We multiply all metrics by 100, so they can be more easily compared to each other.

3 EXPERIMENTS

We perform several experiments to compare the RobBERT-XMLC model to the original BERT-XMLC study, and to analyze the effects of different aspects of the model on its performance. These experiments can be categorised into three main categories: baseline experiments, our main experiment, and model setup experiments.

3.1 Baseline experiments

The baseline experiments are conducted so that we can compare the results of our main experiment, `RobBERT-XMLC`, to a baseline. Since we re-implemented the BERT-XMLC model to make it more easily adjustable for new languages (i.e., changing the BERT model or dataset), we performed our first experiment: `BERT-XMLC`. This experiment explores how well our version works compared to the original model of Bhola et al. [4]. From Table 1 we can see that our results do not completely align with the results of the original code (e.g., our results are Recall@100=80.09, nDCG@100=68.80 and MRR=82.49, compared to 90.26, 81.79 and 90.19 respectively). To the best of our knowledge, our reproduced model includes all the important components of [4], and therefore the gap is likely caused by a difference in the training conditions. Previous research has shown that reproducing deep learning models is a challenging task due to possible uncontrolled randomizations [1], which may explain the discrepancy.

Our next baseline experiments concern exploring the effects of multilingual embeddings on the performance. First, we replace $BERT_{BASE}$ with $BERT_{BASE} multilingual$ to conduct `mBERT-XMLC-EN` on the English vacancies provided in [4]. Comparing `mBERT-XMLC-EN` to `BERT-XMLC`, we see that the monolingual BERT model indeed outperforms its multilingual counterpart as was expected from literature [6]. Next, we replace the English dataset with our Dutch dataset in `mBERT-XMLC-NL`. The two multilingual models show slightly different results. Several factors could potentially cause this slight difference, e.g. the amount of unique skills in the dataset, how much the skills relate to their context in the descriptions, the skill occurrences and the representation of the language in the training set of the BERT model.

[5] https://simpletransformers.ai/

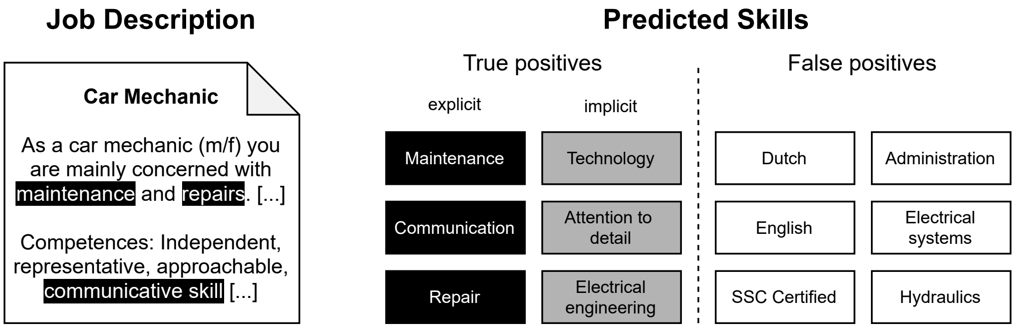

Fig. 1. Illustration of the system output for a vacancy text for a car mechanic. In this example, the model outputs 12 predicted skills, of which 3 are explicit ground truth skills (white text on black), 3 implicit ground truth skills (black text on grey) and 6 false positives (i.e., predicted but not in ground truth).

3.2 Main experiment

The baseline experiments show the added value of a monolingual BERT model. Hence, in our main experiment we explore how the Dutch RobBERT model influences the performance. We present this model as `RobBERT-XMLC`. Even though `RobBERT-XMLC` cannot reliably be compared to `mBERT-XMLC-NL` due to a difference in case-sensitivity, the results are also in line with the expectation: the monolingual model outperforms the multilingual model.

Furthermore, the increase in performance of `mBERT-XMLC-NL` > `RobBERT-XMLC` is larger compared to `mBERT-XMLC-EN` > `BERT-XMLC`. A possible explanation for this could be added value of the case-sensitivity of RobBERT. Based on an average of 20.61 skills per vacancy, about 4 skills were derived within the top 5 and 17 within the top 100 of derived skills. Figure 1 shows an example of the result of our model. Here we are especially interested in the relevance and implicitness of the retrieved implicit skills and false positives. We find the explicit and implicit skills underline the noisy nature of the labels; while maintenance and repair are clear required (explicit) skills for car mechanics, "technology" comes across as a very broad skill. In terms of false positives, we do note the vacancy explicitly mentions "good knowledge of electronics" as a required skill (explaining Electrical Systems), but overall many of the skills seem somewhat relevant yet not very distinctive to the occupation, e.g., language skills (Dutch and English) and Administration.

3.3 Model setup experiments

The model setup experiments are conducted to explore what conditions contribute to the performance of our method. We explore what happens when we omit the bottleneck layer (`RobBERT-XMLC-nobtlnk`) or when we exclude the implicit skills (`RobBERT-XMLC-noimpl`). The results without the bottleneck layer are lower compared to `RobBERT-XMLC`. This proves that the bottleneck layer enhances the model [11]. The added implicit skills seemed to improve the ranking of the skills, since the ranking metrics nDCG and MRR of `RobBERT-XMLC-noimpl` are clearly impacted negatively. This implies that adding implicit skills improves the predicted probability of the explicit skills (e.g. due to an increase in skill occurrences).

4 CONCLUSION & FUTURE WORK

In this study we introduced RobBERT-XMLC and conducted several experiments to show the effectiveness of this approach. The results of our model are promising; the model is able to extract both implicit and explicit skills. In addition, our experiments strengthened the statement that monolingual BERT models outperform their multilingual

Metric	BERT-XMLC	mBERT-XMLC-EN	mBERT-XMLC-NL	RobBERT-XMLC	RobBERT-XMLC-nobtlnk	RobBERT-XMLC-noimpl
Recall@5	15.86	14.23	15.95	18.80	17.15	**22.89**
Recall@10	26.92	23.87	24.77	32.47	28.52	**34.64**
Recall@30	53.08	45.61	42.12	**59.34**	50.99	57.70
Recall@50	65.57	56.75	51.76	**70.91**	61.07	68.57
Recall@100	80.09	70.97	64.95	**83.12**	73.75	80.85
nDCG@5	27.72	25.12	27.33	**31.90**	29.52	26.18
nDCG@10	37.81	33.91	35.43	**44.08**	39.80	33.04
nDCG@30	55.41	48.49	47.03	**62.08**	54.84	43.12
nDCG@50	62.09	54.43	52.14	**68.28**	60.17	46.86
nDCG@100	68.80	61.01	58.16	**73.97**	65.98	50.62
MRR	82.49	77.28	83.74	**90.53**	88.37	56.13

Table 1. The results per metric for each experiment.

equivalent, and showed that adding implicit skills is essential to improve the model's performance on ranking metrics. Moreover, the bottleneck layer proved to be effective in increasing performance. Nevertheless, the performance of the model can be further improved. Future work includes improving the quality of the dataset (e.g. by addressing class imbalance and improving (implicit) skill labeling), performing hyperparameter optimisation and solving the discrepancy between the original study and ours.

5 ACKNOWLEDGMENTS

We thank our colleagues in CDL and MI, specially Erik Volkers, Thomas De Vos and Bart Van Krimpen for their valuable inputs.

REFERENCES

[1] Saeed S Alahmari, Dmitry B Goldgof, Peter R Mouton, and Lawrence O Hall. 2020. Challenges for the Repeatability of Deep Learning Models. *IEEE Access* 8 (2020), 211860–211868.

[2] Dipika Baad et al. 2019. Automatic Job Skill Taxonomy Generation For Recruitment Systems. (2019).

[3] Yeshwanth Balachander and Teng-Sheng Moh. 2018. Ontology based similarity for information technology skills. In *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*. IEEE, 302–305.

[4] Akshay Bhola, Kishaloy Halder, Animesh Prasad, and Min-Yen Kan. 2020. Retrieving Skills from Job Descriptions: A Language Model Based Extreme Multi-label Classification Framework. In *Proceedings of the 28th International Conference on Computational Linguistics*. 5832–5842. https://doi.org/10.18653/v1/2020.coling-main.513

[5] Mariia Chernova. 2020. *Occupational skills extraction with FinBERT*. Master's thesis.

[6] Pieter Delobelle, Thomas Winters, and Bettina Berendt. 2020. Robbert: a dutch roberta-based language model. *arXiv preprint arXiv:2001.06286* (2020).

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

[8] Akshay Gugnani and Hemant Misra. 2020. Implicit Skills Extraction Using Document Embedding and Its Use in Job Recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 13286–13293.

[9] Jobdigger. [n. d.]. De Nederlandse arbeidsmarkt in detail dankzij specifieke JDCO-classificatie. Retrieved July 25, 2021 from https://www.jobdigger.nl/nieuws/de-nederlandse-arbeidsmarkt-in-detail-dankzij-specifieke-jdco-classificatie/

[10] Imane Khaouja, Ismail Kassou, and Mounir Ghogho. 2021. A Survey on Skill Identification From Online Job Ads. *IEEE Access* 9 (2021), 118134–118153.

[11] Jingzhou Liu, Wei-Cheng Chang, Yuexin Wu, and Yiming Yang. 2017. Deep learning for extreme multi-label text classification. In *Proceedings of the 40th international ACM SIGIR conference on research and development in information retrieval*. 115–124.

[12] Ellery Smith, Martin Braschler, Andreas Weiler, and Thomas Haberthuer. 2019. Syntax-based skill extractor for job advertisements. In *2019 6th Swiss Conference on Data Science (SDS)*. IEEE, 80–81.