Proceedings of the 21st Conference on Computer Science and Intelligence Systems (FedCSIS) pp. 263–271
DOI: 10.15439/2026F3041
ISSN 2300-5963 ACSIS, Vol. 47

Two-Step Occupation Coding

Alexander M. Esser
0000-0002-5974-2637
Federal Institute for Vocational Education and Training, Bonn, Germany
University of Koblenz, Germany
Email: alexanderesser@uni-koblenz.de

Jens Dörpinghaus
0000-0003-0245-7752
Federal Institute for Vocational Education and Training, Bonn, Germany
University of Koblenz, Germany
Email: doerpinghaus@uni-koblenz.de

***Abstract*—Occupation coding links job titles in free text to occupational taxonomies and is a core task in labor market research. Existing approaches typically address this problem in a single end-to-end step, jointly identifying job titles and assigning occupational codes. This paper presents a novel two-step approach that separates these tasks. In the first step, a domain-specific Named Entity Recognition (NER) model identifies occupational titles in continuous text, even under noise such as OCR errors. In the second step, the extracted job titles are mapped to a taxonomy, enabling the classifier to focus exclusively on this mapping. We demonstrate that this separation improves accuracy, robustness, and interpretability compared to single-step approaches. The method has been developed for German documents but is transferable to other languages. We further introduce a margin-based confidence criterion for occupation coding, replacing common absolute thresholds. To support reproducibility, we publish the source code and evaluation scripts.**

I. Introduction

OCCUPATION CODING (OC) is the process of linking a given job title to a predefined occupational taxonomy. It can be understood as a text classification problem in which a textual sequence is assigned to one of potentially thousands of classes. Automatically classifying job titles plays a central role in labor market research [1–4]. Job titles are important for job seekers when searching for relevant job postings, for HR departments when filling new positions, and for vocational institutions when aligning their programs with labor market demands [5]. Although job titles generally follow a certain structure, they are often freely entered as unstructured text [6].

Usually, for occupation coding, end-to-end approaches are used, which simultaneously identify job titles and map them to standardized occupational taxonomies. Such single-step approaches, however, mix two conceptually distinct tasks: (i) identify whether a given text span is a job title at all, and (ii) classify a recognized title according to a taxonomy. We propose a two-step occupation coding framework that deliberately separates these tasks.

In the first step, a domain-specific NER model identifies job titles in continuous text, ensuring that only legitimate job title mentions are passed to the subsequent classification stage. This step builds upon our previous work [7] on Named Entity Recognition (NER) in the field of Vocational Education and Training (VET), which provides an NER model for detecting occupational titles and other entity types in noisy textual contexts. In the second step, the identified job titles are mapped to a taxonomy. This division of tasks yields a noise-reduced input for occupation coding, allowing the classifier to focus on the title-taxonomy alignment rather than first filtering out non-occupational phrases. From a conceptual point of view, this improves accuracy by reducing the risk of false positive matches.

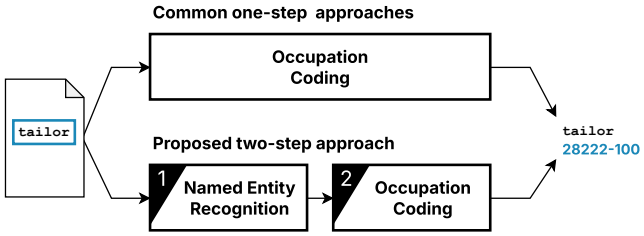


Figure 1: One-step and two-step occupation coding approaches.

Figure 1 illustrates the core idea: Within a VET document, the job title “tailor” is recognized. The corresponding job ID is retrieved from a taxonomy. While common approaches perform identification and classification as one single end-to-end task, the proposed method follows a two-step strategy, first applying NER, and then matching all recognized job titles against a taxonomy. The proposed methods, in particular the components for NER and occupation coding, have been developed for German documents but are transferable to other languages.

In Germany, the reference taxonomy for occupational titles is the *German Classification System of Occupations* (Klassifikation der Berufe; KldB), used by the Federal Employment Agency (BA) and its research institute (IAB). Here, occupations are structured at task level. The latest version is the 2020 revision of KldB 2010. It has been developed for compatibility with the *International Standard Classification of Occupations*[1] (ISCO) [8, 9]. ISCO occupations are structured by skill level and linked to the *European Skills, Competences, Qualifications and Occupations* (ESCO) ontology [10, 11].

The German Classification System of Occupations KldB [12] defines a five-level hierarchy of groups, represented by five-digit codes. The first digit denotes the general occupational area (*Berufsbereich*), down to the fifth digit, which specifies

[1]See https://www.ilo.org/public/english/bureau/stat/isco/isco08/.

the detailed occupational activity (*Berufsgattung*). Each job title is allocated to the lowest of these five groups, e. g., the profession of *pharmaceutical technical assistant* (*Pharmazeutisch-technischer Assistent*; code 81822-105) is assigned to group 81822. Occupation coding approaches typically predict the 5-digit codes, i. e., the occupational group at the lowest level.

While pipeline decomposition is common in information extraction, this paper is one of the first to apply this idea to occupation coding, presenting a two-step approach. Our method demonstrates that splitting occupation coding into two distinct tasks – separately identifying and matching job titles – measurably improves accuracy, robustness, and interpretability.

The main contributions of this paper are:

- We propose a novel *two-step occupation coding approach* that explicitly separates job title identification from taxonomy-based occupation classification, in contrast to existing single-step, end-to-end approaches.
- For *NER*, we developed a domain-specific model, which is trained using a noise-aware, multi-step fine-tuning strategy integrating occupational taxonomies as additional training resources.
- For *occupation coding*, we retrain the model to focus exclusively on matching previously identified job titles against the taxonomy. We systematically compare multiple classifiers, including Support Vector Machines, logistic regression, and XGBoost. We demonstrate that a two-step approach substantially improves accuracy, robustness, and interpretability.
- Furthermore, we introduce a *margin-based confidence criterion* for occupation coding as an alternative to commonly used absolute confidence thresholds.

To support the reproducibility of scientific results, we publicly provide the source code and evaluation scripts for our two-step occupation coding approach.[2]

The remainder is structured as follows: Section II discusses relevant literature in the fields of NER and Occupation Coding. Section III presents the proposed approach. Section IV evaluates the results. Section V addresses limitations and directions for future work, and concludes this study.

II. Related Work

In this section, existing approaches in the fields of NER and occupation coding are reviewed.

A. Named Entity Recognition

The aim of NER is to identify named entities in continuous text according to a set of predefined categories [13]. It can be regarded as a particular instance of the sequence labeling problem, which concerns assigning labels to tokens in an input sequence [14]. A common distinction can be made between classical *heuristic-based* and modern *deep learning-based* NER approaches.

[2]See https://github.com/TM4VETR/two_step_occupation_coding.

a) Heuristic-based Approaches: Within heuristic-based approaches, a further distinction can be made between *rule-based* and *statistical* methods: Rule-based approaches, such as pattern matching techniques and regular expressions in particular, rely on manually crafted rules to detect entities [15]. These methods typically depend on curated lists of known entities. Statistical methods, including Hidden Markov Models (HMMs) [16] and Conditional Random Fields (CRFs) [17], formulate NER as a sequence labeling task using hand-engineered features [18].

Hidden Markov Models (HMMs) model sequences by transition matrices, which represent the joint probability of transitions between hidden input states and observable output states [19]. Conditional Random Fields (CRFs) are undirected graphical models that estimate the conditional probability of a sequence of labels given a corresponding input sequence [20]. For both HMMs and CRFs, algorithms such as the Viterbi algorithm [21] are commonly employed to infer the most probable sequence of states or labels.

b) Deep Learning-based Approaches: In contrast, deep learning-based methods apply modern learning techniques based on Deep Neural Networks (DNNs). These approaches can be further categorized according to the underlying architecture: *LSTM-based models*, in particular bi-directional LSTMs, are frequently used to model contextual information and are often combined with CRFs for structured sequence decoding [22, 23]. Such models typically rely on word- or character-level embeddings, including GloVe, FastText, or FLAIR [24–27].

Transformer-based models, such as BERT and its numerous variants, currently define the state of the art in NER [28]. Transformer architectures are composed of stacked self-attention layers that capture pairwise dependencies between tokens. This enables each layer to access the full input sequence while remaining parallelizable during training [29]. These models do not require explicit dictionaries of known entities, but instead rely on contextual representations learned from large corpora.

Application to VET: A limited number of publications have addressed NER specifically for job titles in VET data. [30] detected and classified occupations in German texts, comparing a rule-based approach with a language model-based approach. [31] fine-tuned BERT and GPT-3 for the automated extraction of German occupations. [5] proposed JobBERT, a deep learning-based model for recognizing job titles and linking them to ESCO classes. Starting from BERT as a pre-trained model, the recognition is based on skill information, showing that skill descriptions are an essential component for job title recognition.

B. Occupation Coding

Occupation coding describes the task of mapping a given job title to a predefined occupational taxonomy, such as the German KldB. It can be viewed as a text classification problem in which a textual input is mapped to one of potentially hundreds of possible classes [2–4].

Automatically classifying job titles plays a central role in labor market research, for instance in the analysis of surveys or administrative data [1].

Research on occupation coding has a long tradition and dates back to a time before the era of computers [32]. For many years, occupational assignments were performed manually. Automation promised lower processing costs, faster production of statistics, and higher consistency in coding decisions [1, 4].

In the field of occupation coding, the work of Malte Schierholz is noteworthy. His contributions include a Master's thesis [3], a PhD thesis [33], and numerous further publications [4, 34–36]. Starting from an early prototype for coding during interviews, these works iteratively improved the methodology and introduced new approaches. [4] also developed the *occupationCoding* R package[3]. This package has been used as a reference for one-step approaches. For integration into Python-based code, we developed a Python wrapper[4].

Analogous to NER, occupation coding approaches can be grouped into *rule-based methods* and *learning-based methods*.

Rule-based methods rely on predefined dictionaries of occupational titles and apply pattern matching or handcrafted rules to assign codes. While these methods are transparent and often work well with standardized terminology, they tend to perform poorly on noisy text (e. g., OCR errors or misspellings), creative job titles (e. g., in Online Job Advertisements (OJAs)), or historical language. Moreover, they require ongoing manual maintenance, as labor market terminology changes over time.

Learning-based approaches apply statistical or machine learning techniques for classification. Classical techniques include Support Vector Machines (SVMs), k-nearest neighbors, logistic regression, naive Bayes, and random forests [2, 37–40]. A comparative evaluation of several of these models was provided by [1]. More recently, deep learning methods – typically based on Transformer architectures or Convolutional Neural Networks (CNNs) – have been applied for occupation coding. In [41], *Carotene* was proposed, a cascade of an SVM and k-nearest neighbor. In [42], a CNN, *DeepCarotene*, was applied for classifying job titles in online advertisements.

While all of the above methods regarded occupation coding as an end-to-end approach, our approach considers occupation coding as two conceptually distinct tasks.

III. Proposed Method

In this section, the proposed two-step occupation coding approach is presented. In the first step, a domain-specific NER model is applied. In the second step, the extracted job titles are mapped to the KldB taxonomy, enabling the classifier to focus exclusively on the title-taxonomy matching.

A. Named Entity Recognition

One key challenge for NER is noisy input data. Noise introduced upstream, such as Optical Character Recognition (OCR) errors or spelling mistakes, can substantially affect NER. Therefore, we applied a Noise-aware Training (NAT) approach that improves the robustness of NER systems in the presence of OCR and spelling errors.

[3]See https://github.com/malsch/occupationCoding.

[4]See https://github.com/TM4VETR/occupation_coding_python.

Table I: Amount of annotated data used for NER and OC.

NER		**OC**
tokens	entities	occupations
62 379 520	1 761 261	1 641 486

A second challenge arises from the variability of job titles, as they evolve over time. The historical VET documents originate from different decades and from two German states, the Federal Republic of Germany (FRG) and the German Democratic Republic (GDR), with different socio-political systems. Additional variations arise from language-specific characteristics. [30] discussed the unique challenges in the German language. One source of variation is gender-specific forms of job titles.

Given these challenges, rule-based approaches are not promising. Instead, a Transformer-based model has been trained, applying transfer learning, NAT and multi-stage fine-tuning, specifically adapted for job titles in German VET data.

The entity OCCUPATION refers to either a specific job title or a higher-level group of occupations. In previous work [7, 43], a finer distinction was made between *job titles* and higher-level *job title groups*. However, confusion analysis revealed difficulties in distinguishing between these entities. Ideally, entity classes should be distinct; therefore, both were merged into the single entity class OCCUPATION. Very general terms (e. g., the term "occupation" itself) should not be recognized as OCCUPATION.

1) Training Data: The training data consist of *historical* and *contemporary* data. These data sources differ in origin, noise characteristics, and their role in the proposed two-step approach.

The historical data consist of VET documents from East and West Germany as well as reunified Germany, especially training regulations and systematic listings of job titles. They cover the period since 1976 and comprise 159 document pages in total. Since these documents were digitized from historical sources, they may contain OCR errors.

The contemporary data were crawled from X (formerly Twitter) [44, 45]. This dataset is born-digital and, therefore, does not contain OCR errors. However, spelling variations and errors may occur. Moreover, job titles frequently appear in non-standardized forms, posing challenges for both entity identification and occupation coding.

The two data sources play different roles in the proposed approach. For NER, both the historical and contemporary data were used as training data. For occupation coding, only the contemporary data were used, since this dataset contains occupations annotated according to the KldB taxonomy. Thus, both datasets contribute to entity identification, while the KldB-annotated contemporary data provide the basis for the subsequent occupation-classification step.

Table I reports the amount of annotated data used for NER (final model within the multi-stage pipeline) and for OC. The NER data are counted at token level, including "O" tokens,

while the OC data are counted at the level of occupational entities.

2) Additional Training Data: To increase the accuracy, additional training data have been integrated. These include word lists of search terms provided by the Federal Employment Agency[5] as well as the KldB taxonomy [12, 46] with its hierarchical structure. For the KldB, both the 2020 and the 1988 versions were integrated in order to account for both contemporary and historical occupational titles.

However, it is important to note that all these additional data differ from real-world data in two important respects: Every word belongs to an entity; the data contain no "O" labels stating that the corresponding token is not part of any entity. Besides, the data contain no OCR noise.

Therefore, the additional training data cannot be used directly for the final model. Instead, we applied multi-stage fine-tuning, introducing an intermediate pre-training stage trained on this data, as described later.

3) Transfer Learning: Transfer learning is a common concept in which pre-trained models, which have been trained on large datasets, are fine-tuned for target tasks on smaller, domain-specific datasets. [47] demonstrated that a backbone model (in their case, for OCR) can be significantly improved by adding only a small amount of data.

As base model, we use a German, cased BERT variant, `dbmdz/bert-base-german-cased`[6], released by the Bavarian State Library, with weights initialized with the original BERT values.

4) Noise-Aware Training: An in-depth analysis of robustness in document processing pipelines is provided by [48]. The study shows that *domain-specific retraining*, i. e., fine-tuning an existing model on a collection of domain-specific documents, often already leads to considerable performance gains. A more sophisticated strategy is the application of *NAT*.

Noise-aware Training (NAT) refers to a set of methods aiming to make downstream Natural Language Processing (NLP) systems like NER robust to noise introduced upstream, e. g., OCR errors. The core idea of NAT is to inject realistic errors into the training data so that the model learns representations that are less sensitive to such perturbations [49, 50]. In practice, this involves injecting typical OCR or spelling errors.

NAT is closely related to *Empirical Error Modeling*, which empirically analyzes typical OCR errors, in order to subsequently inject realistic perturbations [51]. For the proposed method, NAT has been used as an online approach, dynamically injecting synthetic OCR errors (insertions, deletions, substitutions) on-the-fly during training, generating a new noisy variant of each input instance before tokenization.

5) Multi-Stage Fine-Tuning: To integrate the additional training data, we introduce an intermediate pre-training stage, applying *Domain-adaptive Pre-training (DAPT)* [52].

Figure 2 shows the multi-stage training process, with BERT as base model, the intermediate stage pre-trained on the additional data, and the final stage trained on the real-world VET data.

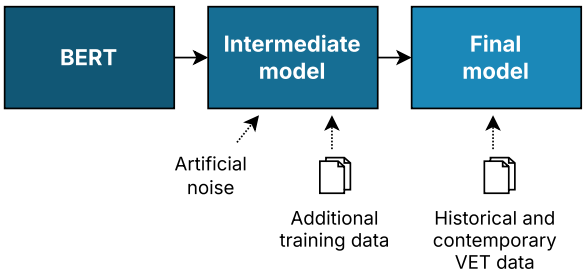


Figure 2: Multi-stage training process.

6) Implementation Details: The framework for training the NER models was built on PyTorch [53] and the Hugging Face Transformers library [54].

a) Bias: In the present setting, different sources of bias may arise: pre-trained model bias, data bias, and a classification bias. When using pre-trained models, these may be subject to bias present in their training data and affecting their predictions. This bias is carried over to the fine-tuned model [55]. Moreover, in the data, certain occupations, time periods, or document types may be over- or underrepresented, which can influence both entity recognition and subsequent occupation classification. Finally, the classification step depends on existing occupational taxonomies, which themselves reflect institutional classification decisions. While a systematic bias analysis is beyond the scope of this paper, the presented model can serve as a basis for future evaluations to identify potential imbalances.

b) Hyperparameters: When training the intermediate and the final model, we searched for the lowest validation loss using the following hyperparameters: During a warm-up phase (10 % of the steps), the learning rate is gradually increased up to the target rate of $2 \cdot 10^{-5}$ (learning rate scheduling) and subsequently decreased as training progresses. Especially when fine-tuning pre-trained models, smaller learning rates allow for more refined adjustments. The best-performing model over all epochs, measured by the F_1 score (micro-averaged on entity level), was saved.

c) Class Weighting and Oversampling: To compensate for the significant imbalance, *class weighting* and *oversampling* have been applied. The majority of labels is "O", stating that the corresponding token is not part of any entity. The loss function would be dominated by these labels, and the model could degenerate into predicting only "O" (class collapse). A certain proportion of "O" tokens was deliberately removed, as the model can learn little from these tokens. The weights for class weighting have then been normalized and clamped to avoid extreme values. Oversampling, during training, replicates token windows that contain positive labels (non-"O") and thereby increases the relative frequency of positive examples presented to the model. This artificially rebalances the class distribution and strengthens the loss signal for rare entities.

[5] See https://www.arbeitsagentur.de/institutionen/dkz-downloadportal.
[6] See https://huggingface.co/dbmdz/bert-base-german-cased.

For pre-training the intermediate model, slightly different hyperparameters and settings have been used: The additional training data for the intermediate model contain no "O" labels. Therefore, no oversampling, weight balancing, or clamping has been applied.

d) Annotations: For annotating entities, we used the *BIO* format, in which each word (token) is assigned a tag. The tag B-OCCUPATION indicates that the token marks the beginning of an entity span, i. e. of a job title or a higher-level occupational group; I-OCCUPATION indicates that the token lies inside an entity span; "O", outside any entity span.

B. Occupation Coding

When applying a two-step occupation coding approach, the classifier needs to be retrained, as it no longer needs to identify job titles in free text. Instead, it can focus exclusively on matching a given job title to an occupational taxonomy, starting from a set of previously identified job titles.

1) Classifiers: For retraining, we systematically compared three types of classifiers, an SVM, XGBoost as a tree-based approach, and logistic regression. For training and evaluating these classifiers, the `scikit-learn` Python package [56] was employed. All three classifiers require input data in the form of numerical vectors. To generate these embeddings, a TF-IDF vectorizer was used. We deliberately focused on classical classifiers to ensure comparability with the evaluated one-step occupation coding approach and to isolate the effect of task separation. Training an advanced classifier, such as a Transformer-based model, is left for future work.

From a theoretical perspective, one might assume that decision trees can best reflect the hierarchical structure of the KldB taxonomy. However, XGBoost does not automatically exploit label hierarchies. It still treats all classes as flat categories. The results show that the SVM achieves the best performance among the evaluated classifiers (see Section IV-A2).

A linear SVM represents each sample as a point in a high-dimensional feature space. It then learns linear decision boundaries (hyperplanes) that split each space into two half-spaces. To effectively discriminate among many classes, as in the KldB taxonomy, a high number of dimensions is necessary.

In this use case, the SVM yields better performance than a tree-based method. Therefore, we selected an SVM as classifier for subsequent steps.

C. Confidence Thresholding

The central idea of two-step occupation coding is to allow the classifier to focus exclusively on matching a previously identified job title to a taxonomy. The confidence[7] associated with this decision is a critical parameter.

The combined confidence *conf* for the overall process is computed as the product of the NER and OC confidence:

$$\mathrm{conf} = \mathrm{conf}_{\mathrm{NER}} \cdot \mathrm{conf}_{\mathrm{OC}}\,. \tag{1}$$

[7]The resulting probabilities are not confidences in a strict sense, but they are, after calibration, a suitable measure for comparing thresholding approaches.

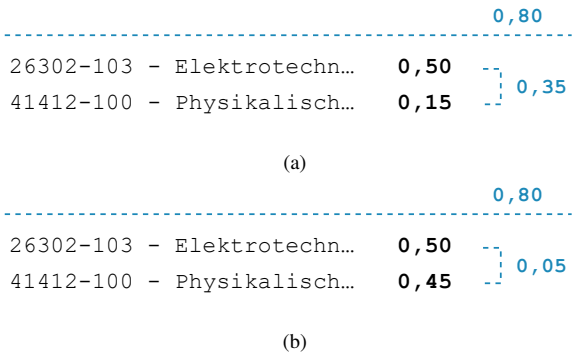


Figure 3: Absolute threshold and confidence margin – two examples.

As an additional improvement, besides the two-step approach, we introduced a confidence margin approach for occupation coding. Common approaches, including the *occupationCoding* R package [4], the *Carotene* classifier [41], and the *DeepCarotene* CNN [42], typically rely on an absolute threshold to determine whether the algorithm is sufficiently confident about a prediction. Only if the confidence score (i. e., the predicted probability assigned by the classifier) is above this threshold is the job ID assigned to the job title.

We, in contrast, propose a margin-based approach, as commonly used in other application domains, such as computer vision. The key idea is to replace the absolute threshold with a margin defined as the difference between the highest and the second-highest predicted confidence score. Rather than asking whether a prediction exceeds a fixed threshold, the margin-based approach evaluates how confident the algorithm is about its top prediction relative to the second-highest one.

Figure 3 illustrates this idea in two examples. In both cases, the job ID with the highest confidence is "26302-103 – Elektrotechnische/r Assistent/in" (*electrical engineering assistant*), and the second-highest is "41412-100 – Physikalisch-technische/r Assistent/in" (*physical–technical assistant*). In neither example does the confidence exceed the threshold of 0.8. Consequently, using an absolute threshold approach, both examples would be classified as "not recognized".

However, in Figure 3(a), there is a large difference between the two confidence scores, whereas in Figure 3(b) this difference is small. In the first example, a margin-based approach, with a suitably chosen margin, would return the top job ID. In contrast, in the second example, no job ID would be recognized, since the algorithm is not sufficiently confident in distinguishing between the two most likely candidates. The recognition rates obtained using both approaches are reported in Section IV-A4.

IV. Evaluation

In this section, the proposed method is evaluated, demonstrating that task separation allows error sources to be attributed more clearly to either identification or classification.

Standard evaluation metrics for NER and occupation coding include *precision*, *recall*, and the F_1 *score*. These metrics are

Table II: Accuracy for Named Entity Recognition.

	Precision	Recall	F_1 score
NER	79.70 ± 2.41 %	92.95 ± 1.11 %	85.79 ± 1.33 %

Table III: Accuracy of different classifiers.

Classifier	F_1 score
SVM	94.92 %
XGBoost	82.71 %
logistic regression	62.59 %

derived from the numbers of *true positives* TP, *false positives* FP, and *false negatives* FN:

$$\begin{aligned} \mathrm{Pr} &= \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} \\ \mathrm{Re} &= \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} \\ F_1 &= \frac{2\cdot \mathrm{Pr}\cdot \mathrm{Re}}{\mathrm{Pr}+\mathrm{Re}} \end{aligned} \tag{2}$$

A. Evaluation Results

The annotated data were split into a training, test, and validation set using a ratio of 70:20:10. Splitting was performed at segment level, not at token level, to avoid breaking multi-word entities.

1) NER Accuracy: Table II reports the results of the separate NER step, evaluated on the historical and contemporary real-world data. The NER model contains multiple stochastic components. To ensure statistical robustness, the model was trained five times using different random seeds and the standard deviation of the evaluation metrics was reported.

2) Comparing Classifiers: As mentioned in Section III-B1, three classifiers – an SVM, XGBoost, and logistic regression – have been systematically compared. Table III reports the F_1 scores of each of these classifiers, evaluated on a test set which was split from the standardized word lists, with the SVM achieving the highest accuracy (F_1 score: 94.92 %). Note that the accuracy on standardized word lists is significantly higher than when the model is later evaluated on real-world data, as reported in the following tables.

3) One or Two Steps: Table IV presents the central outcome of this study: A two-step occupation coding approach leads to substantially higher recognition rates than one-step approaches.

The model for the two-step approach has been trained using the same training data as the one-step approach implemented in the *occupationCoding* R package, namely the list of occupations provided by the Federal Employment Agency. This demonstrates that a two-step approach offers conceptual advantages and yields higher recognition rates (F_1 score: 57.26 % compared to 46.94 %). The improvement in F_1 score is mainly driven by a higher precision, while recall remains almost unchanged. This indicates that the advantage of the two-step approach lies in reducing false positives, as conceptually expected: In the one-step setting, occupation coding is applied directly to free text and has to solve entity identification and classification simultaneously. As a result, non-occupational fragments, OCR noise, or context words may be incorrectly matched to occupations. In the two-step setting, the NER model first restricts the input of the occupation classifier to occupational entities. This task separation reduces the number of irrelevant candidate strings and thereby improves precision. The number of false positives is considerably lower for the two-step approach.

Table IV: Accuracy for one-step and two-step occupation coding.

Approach	Precision	Recall	F_1 score
one-step	38.23 %	60.78 %	46.94 %
two-step	54.16 %	60.75 %	57.26 %

Table V: Confidence thresholding approaches.

Thresholding approach	F_1 score
Baseline	34.07 %
Absolute threshold	43.89 %
Confidence margin	43.96 %
Combined (absolute & margin)	43.96 %

4) Confidence Thresholding: Table V lists the F_1 scores for the baseline approach (i. e. accepting all predictions without any threshold), the absolute threshold approach, the confidence margin approach, and the combination of both methods.

Both thresholding strategies outperform the baseline approach. Evaluated on historical and contemporary real-world data, the confidence margin approach achieves a slightly higher accuracy (F_1 score: 43.96 %) than the absolute threshold approach (F_1 score: 43.89 %). The combination of both approaches, in this use case, achieves an identical F_1 score as the confidence margin approach. The comparison is intended to illustrate the intuition behind both thresholding strategies and to introduce confidence-margin thresholding as an alternative to common absolute thresholding. Statistically robust conclusions would require a broader evaluation across multiple data splits and repeated runs.

5) Benchmark Dataset: In the field of occupation coding, there is no standard benchmark dataset. However, [5] published a dataset[8] to evaluate their JobBERT approach. This test set is a subset of the ESCO dataset, containing 15 462 occupation instances and 1 369 unique occupations. We evaluated the proposed two-step approach on this JobBERT dataset.

It should be emphasized that the different approaches are only partially comparable, as they rely on different methodologies. JobBERT [5] and the Sentence-BERT (SBERT) approach [57] are based on *contextual information* obtained through semantic analysis of skill-related data. In contrast, the proposed method is based on the recognition and classification of specific entities. Conceptually, semantic-based approaches are superior to entity-based ones. However, they must be properly trained; otherwise, entity-based approaches can indeed outperform semantic-based ones.

[8]See https://github.com/jensjorisdecorte/JobBERT-evaluation-dataset.

Table VI: Accuracy on the JobBERT dataset.

Method	Concept	Language	Source	Recall@1	F_1 score
SBERT [57]	contextual information	EN	[5]	19.30 %	
JobBERT [5]	contextual information	EN	[5]	19.19 %	
Fine-tuned JobBERT [5]	contextual information	EN	[5]	22.48 %	
Proposed method (two-step, combined threshold)	classification	DE			44.20 %

To evaluate the proposed two-step approach, we performed a document-level evaluation on the original ESCO HTML pages[9]. The goal is to find the ground truth occupation with the corresponding job ID on the HTML page. If multiple occupations were found, the most frequent job ID, weighted by the confidence, was selected as the final prediction for the page (weighted majority vote). A prediction is counted as correct only if the occupation has been recognized during NER and the classification result is correct. In cases where the classifier predicted multiple job IDs, all candidates were considered and contributed equally to the final decision. Samples without a valid ground truth job ID were excluded from the evaluation.

For confidence thresholding, a combination of an absolute threshold (empirically chosen margin: $t = 0.6$) and a confidence margin approach (empirically chosen margin: $m = \text{top}_1 - \text{top}_2 = 0.3$) was applied.

The JobBERT test data are in English. In contrast, the proposed two-step approach has been trained on German data. However, since ESCO is multilingual, a mapping between English and German occupations could be established. The two-step approach was evaluated on the same JobBERT data, only in German instead of English. In this work, the focus was to demonstrate the fundamental conceptual advantages of a two-step approach. In future work, we plan to extend our approach by training multilingual models.

Table VI reports the accuracy of three established approaches, SBERT [57], JobBERT, and a fine-tuned JobBERT variant [5], as well as the proposed two-step approach. The evaluation results for the first three methods are taken from [5]. The reported *micro Recall@1* (i. e., the top-ranked prediction) most closely corresponds to the F_1 score used for the two-step approach. Although the results are only partially comparable, they nonetheless indicate that the proposed method achieves at least a comparable level of accuracy, highlighting the potential of the approach. Establishing a standardized benchmark dataset for occupation recognition and classification remains an avenue for future work.

For the proposed method, recall (82.86%) is substantially higher than precision (30.14%). From a theoretical perspective, a high recall indicates that the system is highly sensitive and captures most job titles, resulting in few false negatives. Such a coverage-oriented system identifies most of the potential job titles for subsequent analysis, albeit at the cost of reduced precision, including some incorrect predictions.

Regarding the effect of multi-stage fine-tuning and NAT on the recognition of vocational entities, a detailed analysis is provided in our prior work [7, 43].

[9]See https://github.com/alexander-esser/jobbert_html_dataset.

V. Conclusion

This paper presented a two-step occupation coding approach that deliberately separates occupation coding into two distinct tasks: (i) detecting job titles in continuous text and (ii) mapping the identified titles to an occupational taxonomy.

In the first NER step, multi-stage fine-tuning was employed to incorporate additional training data and NAT to improve the robustness against typical sources of noise, such as spelling errors. In the second OC step, several classifiers were evaluated, with an SVM yielding the strongest performance. It was demonstrated that using a confidence margin, rather than a fixed absolute confidence threshold, leads to additional small improvements in recognition accuracy.

Limitations and Future Work

For occupation coding, we used classical machine learning classifiers to maintain comparability with one-step approaches and focus on demonstrating the conceptual benefits of task separation. For practical applications, training an advanced classifier, such as a Transformer-based model, promises to further enhance recognition rates.

Currently, the approach relies on explicitly mentioned job titles, which is error-prone, especially in contemporary social media data where occupations are frequently paraphrased rather than explicitly stated [58]. Future work will, therefore, focus on recognizing occupations based on semantic context, in a way similar to JobBERT or SBERT, but as a two-step approach. The NER model introduced in [7, 43] already supports the recognition of multiple entity types, such as skills, working activities, or work equipment. This provides a foundation for a more comprehensive semantic understanding and for further developing the method towards practical application.

From a practical perspective, the results should be interpreted as a methodological improvement rather than a ready-to-use occupation coding system. For deploying the application in practice, further improvements in contextual recognition and classification accuracy are required.

Another goal of future work is to extend the system to English and other languages. The KldB taxonomy is aligned with ISCO, making it possible to automatically generate annotated ground truth data and to train multilingual models.

In summary, this study demonstrated that the proposed two-step approach achieves substantial gains in accuracy, robustness, and interpretability compared to one-step occupation coding methods.

ACKNOWLEDGMENTS

During the preparation of this manuscript, the authors used generative AI tools (ChatGPT, DeepL) for spelling and grammar checks and wording suggestions. The authors reviewed and edited all generated content and take full responsibility for the final text.

In addition, during the development of software components used in this study, generative AI tools (ChatGPT and GitHub Copilot) were used for assisted coding. All generated code was reviewed, tested, and integrated by the authors, who take full responsibility for the implemented software.

REFERENCES

[1] R. Dorau, K. Hein, J. Dörpinghaus, and M. Tiemann, "Automated classification of German job titles according to KldB: Challenges and novel methods," in *INFORMATIK*, 2025. DOI: 10.18420/inf2025_86

[2] H. Gweon, M. Schonlau, L. Kaczmirek, M. Blohm, and S. Steiner, "Three methods for occupation coding based on statistical learning," *Journal of Official Statistics*, vol. 33, no. 1, 2017. DOI: 10.1515/jos-2017-0006

[3] M. Schierholz, "Automating survey coding for occupation," M.S. thesis, Ludwig-Maximilians-Universität München, 2014.

[4] M. Schierholz and M. Schonlau, "Machine learning for occupation coding – a comparison study," *JSSAM*, vol. 9, no. 5, 2021. DOI: 10.1093/jssam/smaa023

[5] J.-J. Decorte, J. Van Hautte, T. Demeester, and C. Develder, "JobBERT: Understanding job titles through skills," *arXiv preprint arXiv:2109.09605*, 2021.

[6] J. Van Hautte, V. Schelstraete, and M. Wornoo, "Leveraging the inherent hierarchy of vacancy titles for automated job ontology expansion," *arXiv preprint arXiv:2004.02814*, 2020.

[7] A. M. Esser and J. Dörpinghaus, "Noise-aware named entity recognition for historical VET documents," in *VISAPP*, 2026.

[8] A. Fischer and J. Dörpinghaus, "Web mining of online resources for German labor market research and education: Finding the ground truth?" *Knowledge*, vol. 4, no. 1, 2024. DOI: 10.3390/knowledge4010003

[9] J. Dörpinghaus, J. Binnewitt, S. Winnige, K. Hein, and K. Krüger, "Towards a German labor market ontology: Challenges and applications," *Applied Ontology*, vol. 18, no. 4, 2023. DOI: 10.3233/AO-230027

[10] T. Reiser, J. Dörpinghaus, P. Steiner, and M. Tiemann, "Towards a dataset of digitalized historical German VET and CVET regulations," *Data*, vol. 9, no. 11, 2024. DOI: 10.3390/data9110128

[11] R. Dorau and K. Hein, "Towards the automated classification of German job titles according to KldB," in *FedCSIS*, 2025. DOI: 10.15439/2025F4081

[12] Bundesagentur für Arbeit, *Klassifikation der Berufe 2010 – überarbeitete Fassung 2020*, 2020.

[13] D. Nadeau and S. Sekine, "A survey of named entity recognition and classification," *Lingvisticae Investigationes*, vol. 30, no. 1, 2007. DOI: 10.1075/li.30.1.03nad

[14] E. F. Tjong Kim Sang and F. De Meulder, "Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition," in *CoNLL*, 2003. DOI: 10.3115/1119176.1119195

[15] T. Eftimov, B. Koroušić Seljak, and P. Korošec, "A rule-based named-entity recognition method for knowledge extraction of evidence-based dietary recommendations," *PLOS ONE*, vol. 12, no. 6, 2017. DOI: 10.1371/journal.pone.0179488

[16] D. M. Bikel, R. Schwartz, and R. M. Weischedel, "An algorithm that learns what's in a name," *Machine Learning*, vol. 34, no. 1, 1999. DOI: 10.1023/A:1007558221122

[17] A. McCallum and W. Li, "Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons," in *CoNLL*, 2003. DOI: 10.3115/1119176.1119206

[18] D. Jurafsky and J. H. Martin, *Speech and Language Processing*. 2025.

[19] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *IEEE*, vol. 77, no. 2, 1989. DOI: 10.1109/5.18626

[20] J. D. Lafferty, A. McCallum, and F. C. N. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *ICML*, 2001.

[21] A. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Transactions on Information Theory*, vol. 13, no. 2, 1967. DOI: 10.1109/TIT.1967.1054010

[22] Z. Huang, W. Xu, and K. Yu, "Bidirectional LSTM-CRF models for sequence tagging," *arXiv preprint arXiv:1508.01991*, 2015.

[23] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer, "Neural architectures for Named Entity Recognition," in *NAACL-HLT*, 2016. DOI: 10.18653/v1/N16-1030

[24] J. Pennington, R. Socher, and C. Manning, "GloVe: Global vectors for word representation," in *EMNLP*, 2014. DOI: 10.3115/v1/D14-1162

[25] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Transactions of the Association for Computational Linguistics*, vol. 5, 2017. DOI: 10.1162/tacl_a_00051

[26] E. Grave, P. Bojanowski, P. Gupta, A. Joulin, and T. Mikolov, "Learning word vectors for 157 languages," *arXiv preprint arXiv:1802.06893*, 2018.

[27] A. Akbik, D. Blythe, and R. Vollgraf, "Contextual string embeddings for sequence labeling," in *COLING*, 2018.

[28] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *NAACL-HLT*, 2019. DOI: 10.18653/v1/N19-1423

[29] A. Vaswani et al., "Attention is all you need," *NeurIPS*, 2017.

[30] T. Reiser, J. Dörpinghaus, and M. Tiemann, "Detecting occupations in German texts: Challenges and data," in *AISEER*, 2025.

[31] P. Safikhani, H. Avetisyan, D. Föste-Eggers, and D. Broneske, "Automated occupation coding with hierarchical features: A data-centric approach to classification with pre-trained language models," *Discover Artificial Intelligence*, vol. 3, no. 1, 2023. DOI: 10.1007/s44163-023-00050-y

[32] National Research Council, *Work, Jobs, and Occupations: A Critical Review of the Dictionary of Occupational Titles*. 1980.

[33] M. Schierholz, "New methods for job and occupation classification," Ph.D. dissertation, Universität Mannheim, 2019.

[34] M. Schierholz, M. Gensicke, N. Tschersich, and F. Kreuter, "Occupation coding during the interview," *J. R. Stat. Soc. Ser. A*, vol. 181, no. 2, 2018. DOI: 10.1111/rssa.12297

[35] M. Schierholz, "Eine Hilfsklassifikation mit Tätigkeitsbeschreibungen für Zwecke der Berufskodierung," *AStA Wirtschafts- und Sozialstatistisches Archiv*, vol. 12, no. 3, 2018. DOI: 10.1007/s11943-018-0231-2

[36] A. Bethmann, M. Schierholz, K. Wenzig, and M. Zielonka, "Automatic coding of occupations using machine learning algorithms for occupation coding in several German panel surveys," in *Statistics Canada Symposium*, 2014.

[37] R. H. Creecy, B. M. Masand, S. J. Smith, and D. L. Waltz, "Trading MIPS and memory for knowledge engineering," *Communications of the ACM*, vol. 35, no. 8, 1992. DOI: 10.1145/135226.135228

[38] D. E. Russ et al., "Computer-based coding of free-text job descriptions to efficiently identify occupations in epidemiological studies," *Occupational and Environmental Medicine*, vol. 73, no. 6, 2016. DOI: 10.1136/oemed-2015-103152

[39] K. Takahashi, H. Taki, S. Tanabe, and W. Li, "An automatic coding system with a three-grade confidence level corresponding to the national/international occupation and industry standard: Open to the public on the web," in *KEOD*, 2014. DOI: 10.5220/0005131703690375

[40] A. Ikudo, J. I. Lane, J. Staudt, and B. A. Weinberg, "Occupational classifications: A machine learning approach," *Journal of Economic and Social Measurement*, vol. 44, no. 2-3, 2019. DOI: 10.3233/JEM-190463

[41] F. Javed, Q. Luo, M. McNair, F. Jacob, M. Zhao, and T. Kang, "Carotene: A job title classification system for the online recruitment domain," in *BigDataService*, IEEE, 2015. DOI: 10.1109/BigDataService.2015.61

[42] J. Wang, K. Abdelfatah, M. Korayem, and J. Balaji, "DeepCarotene: Job title classification with multi-stream convolutional neural network," in *Big Data*, IEEE, 2019. DOI: 10.1109/BigData47090.2019.9005673

[43] A. M. Esser and J. Dörpinghaus, "Noise-aware Named Entity Recognition for historical VET documents," 2026, Extended version of the VISAPP 2026 paper. arXiv: 2601.00488 [cs.CL].

[44] K. Hein, M. Tiemann, and J. Dörpinghaus, "Talking about tasks or just sharing job offers? a case study on job-related tweets," in *INFORMATIK*, 2025. DOI: 10.18420/inf2025_88

[45] M. Tiemann and J. Dörpinghaus, "Occupations and education in X data: How representative is the data?" In *INFORMATIK*, 2025. DOI: 10.18420/inf2025_90

[46] Bundesagentur für Arbeit, *Klassifikation der Berufe*, 1988.

[47] M. Namysł and I. Konya, "Efficient, lexicon-free OCR using deep learning," in *ICDAR*, 2019. DOI: 10.1109/ICDAR.2019.00055

[48] M. Namysł, "Robust information extraction from unstructured documents," Ph.D. dissertation, Rheinische Friedrich-Wilhelms-Universität Bonn, 2023.

[49] M. Namysł, S. Behnke, and J. Köhler, "NAT: Noise-aware training for robust neural sequence labeling," in *ACL*, 2020. DOI: 10.18653/v1/2020.acl-main.138

[50] G. Xu, W. Ding, W. Fu, Z. Wu, and Z. Liu, "Robust learning for text classification with multi-source noise simulation and hard example mining," in *ECML PKDD*, 2021. DOI: 10.1007/978-3-030-86517-7_18

[51] M. Namysł, S. Behnke, and J. Köhler, "Empirical error modeling improves robustness of noisy neural sequence labeling," in *ACL-IJCNLP Findings*, 2021. DOI: 10.18653/v1/2021.findings-acl.27

[52] S. Gururangan et al., "Don't stop pretraining: Adapt language models to domains and tasks," in *ACL*, 2020. DOI: 10.18653/v1/2020.acl-main.740

[53] A. Paszke et al., "PyTorch: An imperative style, high-performance deep learning library," *NeurIPS*, vol. 32, 2019.

[54] T. Wolf et al., *Transformers*, 2022.

[55] A. Wang and O. Russakovsky, "Overwriting pretrained bias with finetuning data," in *ICCV*, 2023. DOI: 10.1109/ICCV51070.2023.00366

[56] F. Pedregosa et al., "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, 2011.

[57] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using siamese BERT-networks," in *EMNLP-IJCNLP*, Association for Computational Linguistics, 2019. DOI: 10.18653/v1/D19-1410

[58] A. M. Esser and T. Reiser, "Processing and linking vocational documents," *mimeo*, 2026.