A Semi-supervised Model for Automated Classification of AI-related Job Tasks using Bloom's Taxonomy

Christophe Mbounang Fongang
Ruhr West University of Applied Sciences
Mülheim an der Ruhr, Germany
cmbounang2@gmail.com

Michael Vogelsang
Ruhr West University of Applied Sciences
Mülheim an der Ruhr, Germany
michael.vogelsang@hs-ruhrwest.de

Timm Eichenberg
University of Applied Sciences Weserbergland
Hameln, Germany
eichenberg@hsw-hameln.de

Britta Rüschoff
FOM University of Applied Sciences for Economics and Management
Düsseldorf / Münster, Germany
britta.rueschoff@fom.de

Anne Stockem Novo
Ruhr West University of Applied Sciences
Mülheim an der Ruhr, Germany
anne.stockem-novo@hs-ruhrwest.de

***Abstract*—Recent developments in artificial intelligence have been giving rise to new job profiles comprising several tasks which require different technical and cognitive skills. These tasks, most of which are new to human resources staff and job seekers, make it difficult to assess the complexity of a job and therefore challenging to select the right candidate profile. In this paper, we present an automated classification scheme based on Bloom's taxonomy, a hierarchical model of educational objectives, which is applied on online AI-related job postings. The main goals are an improvement of the prediction accuracy of the classification model as well as an analysis of requirements for AI-related jobs. The modeling relies on a pre-trained BERT model which is fine-tuned on our dataset. In a two-step process, a semi-supervised approach is used in order to benefit from a large amount of unlabeled data. The model-generated pseudo-labels have been evaluated by the experts. Taking advantage of the now available larger correctly labeled dataset, a fully supervised training is done on the enhanced dataset and compared to the semi-supervised approach. Our results show that both models can classify AI-related tasks with good performance, supporting the use of semi-supervised training. The performance limitation rather lies in the subjectivity of expert labeling which is addressed in more detail in the paper. Moreover, we observe the model being more accurate at classifying tasks at higher levels of Bloom's taxonomy than at lower levels.**

***Index Terms*—Artificial Intelligence, Bloom's Taxonomy, semi-supervised learning, BERT, online job postings**

I. Introduction

Over the last decade, artificial intelligence (AI) has seen rapid progress thanks to the availability of large-scale datasets and new machine learning techniques. This technological advance leads mainly to automation, which on the one hand contributes to the change of usual jobs, and on the other hand to the creation of new jobs and new tasks [1], [2]. As a result, professional profiles are changing, with AI-related jobs becoming increasingly important. The positive aspect of AI on employment was also demonstrated by Damioli et al. (2023), as it fosters the development of new business sectors and offers a wide variety of new jobs [3]. These new jobs, with their new tasks, most of which are new to human resources staff and job seekers, make it difficult to assess the complexity of a job and therefore challenging to select the right candidate profile.

Bloom's Taxonomy is a hierarchical model of cognitive skills which is used to classify learning objectives into six progressively complex levels. Though commonly used in educational settings to guide the development of curricula and assessment tools, the taxonomy provides a general framework to evaluate the complexity of a task, which in the context of job-related tasks can serve as an estimate of job complexity. It classifies cognitive tasks into 6 levels, from the least to the most challenging: "Remember", "Understand", "Apply", "Analyze", "Evaluate", and "Create" [4]. Each level is characterized by a number of keywords that identify the tasks assigned to them. However, the context in which these keywords are used also has to be considered, making task classification non-trivial and requiring expert knowledge. Shaikh et al. (2021) applied the keyword approach to automatically classify learning outcomes using Bloom's Taxonomy, achieving an accuracy of just 55% [5].

In this study, we present an automated classification scheme based on Bloom's Taxonomy which is trained and tested on a small expert-labeled dataset of AI-related job tasks. The modeling relies on a pre-trained BERT model which is fine-

979-8-3503-5409-6/24/$31.00 ©2024 IEEE
DOI 10.1109/CAI59869.2024.00019

tuned on our dataset. In a two-step process, a semi-supervised approach is used in order to benefit from a large amount of unlabeled data. The model-generated pseudo-labels have been evaluated by the experts. Taking advantage of the now available larger correctly labeled dataset, a fully supervised training is done on the enhanced dataset and compared to the semi-supervised approach.

The paper is structured as follows. After briefly reviewing previous work in Section 2, we present our semi-supervised learning approach and the BERT model setup in Section 3. Afterwards, in section 4, we present and discuss the results. Section 5 concludes the study and suggests possible future work.

II. Relevant work

A. AI-related jobs

Analysis of the required job skills has been carried out in a number of studies to promote the use and development of new AI-related technologies. Samek et al. (2021) analyzed the required skills in AI-related jobs and found that next to technical skills, social and emotional skills are almost as important in AI-related jobs [6]. More specifically they show a high need in two groups of skills, one related to the application of AI and the other to the development or creation of AI programs, and show a correlation between these two groups. They also found that AI-related jobs require competencies linked to Big Data. According to De Mauro et al. (2018), there are 4 different job groups linked to Big Data: “Business Analysts, Data Scientists, Big Data Developers, and Big Data Engineers.” These job groups also require skills linked to development for the more technically oriented roles and skills linked to analysis and application for the business-oriented roles [7].

B. Bloom’s Taxonomy

In 1956, Bloom et al. [8] introduced a taxonomy of educational objectives in the cognitive domain, categorizing different learning levels based on cognitive processes. Anderson and Krathwohl [4] later revised this taxonomy in 2001 by changing terminology from nouns to verbs and introducing a two-dimensional structure that incorporated different types of knowledge and cognitive processes, resulting in a taxonomy with six levels: Remember, Understand, Apply, Analyze, Evaluate, and Create. This paper will use the revised taxonomy by Anderson and Krathwohl.

Several studies have been using Bloom’s Taxonomy to automatically categorize assignments in educational content. Yahya and Osman (2011) used Support Vector Machines (SVM) to classify questions in the field of E-learning, based on Bloom’s cognitive levels. The machine learning model was trained on a dataset of 272 questions taken from the internet, resulting in a precision of 85.83% and a recall of 29.14%. The authors highlight the need for a large quantity of labeled data to improve the model’s performance [9]. This observation was confirmed by Zhang et al. (2021). They trained a BERT model on a dataset of 504 questions related to computer science, manually labeled with Bloom’s Taxonomy levels. This dataset was unbalanced, resulting in an accuracy of 59.2% for the six classes. The dataset was merged for the least represented classes to train the model on four classes, yielding in an accuracy of 68.52%. Finally, further model improvement was obtained by removing these three classes and training the model only on the three remaining, i.e. lowest-hierarchical, classes. This increased the accuracy to 82.61% [10].

In a study from 2021, Shaikh et al. (2021) used Bloom’s Taxonomy to automatically classify manually labeled datasets of 829 learning course outcomes and 600 assignments. They compared the keyword-based classification approach with the machine learning approach. Using an LSTM neural network, an increased accuracy of 74% was achieved over 55% for the keyword-based approach [5]. Other studies using Bloom’s Taxonomy have been pursued to automatically classify learning objectives. On a dataset of 21,380 manually labeled objectives, Li et al. (2022) implemented and compared two classification methods, namely the multi-class multi-label classification aiming to recognize all levels at the same time and the multiple binary classifiers allowing to train a binary model for each level. They trained different machine learning and deep learning models including Random Forest, XGBoost, logistic regression, SVM, naive Bayes, and BERT. The results showed that BERT outperformed all other classifiers and the multiple binary classifier approach gave better results. The authors also recommend the use of semi-supervised learning to achieve better performance [11].

C. Semi-supervised Learning

Semi-supervised learning (SSL) based on pseudo-labeling was introduced by Lee (2013). It involves training a model on the available data first, then using the trained model to generate pseudo-labels of the unlabeled data. The generation of pseudo-labels is thresholded with a class probability to select the labels most likely to be correct [12].

Nowadays, there are several ways of integrating unlabeled data into the learning process. Lee et al. (2019) trained a semi-supervised neural network with different proportions of labeled and unlabeled data and compared its performance with the supervised approach. They confirmed that SSL outperforms supervised learning (SL) and that the performance of the SSL model scales with the amount of available data. However, they showed that the uncontrolled addition of unlabeled data to the labeled data can sometimes weaken the performance of the model and therefore recommend more care to select appropriate unlabeled data [13]. In line with this recommendation, Ghosh and Desarkar (2020) successively integrated unlabeled datasets with higher confidence with those labeled by self-learning. They defined this integration process through two criteria, both specific to each class of dataset. On the one hand, by setting a threshold for each class based on the initial performance obtained on the model trained by supervised learning. On the other hand, they defined the number of pseudo labels to be added by each iteration in each class according to the number of labeled records initially present in that class.

This allowed them to manage the eventual problem of the unbalanced dataset [14].

In 2021, a study highlighted the limitations of a fixed threshold for selecting unlabeled data with high confidence and proposed an SSL approach with a variable threshold [15]. Since the model is likely to improve its performance after each iteration, they progressively upgrade the value of the threshold, allowing the model to be increasingly selective.

The majority of the work on SSL with pseudo-labels uses a threshold to incorporate unlabeled data into the learning process. Kumar et al., (2021) noted a limitation of this approach in the sense that it is difficult to choose the appropriate threshold, and an inappropriate threshold can significantly degrade the performance of the model. They proposed an approach that avoids this difficulty when generating pseudo-labels by training a binary classifier for each class in a "one-vs-all" strategy. Each binary classifier assigns the pseudo-label of its corresponding class, by classifying unlabeled data as positive. Data classified as negative are passed on to other binary classifiers. For image, sound, and text classification, they demonstrated the superiority of their approach over several other SSL approaches [16].

III. Method

A. Data Collection and Labeling

The dataset used in this experiment consists of 1966 AI-related job offers in the German language extracted from three major job portals for the period from March to September 2022. The key search term was "Künstliche Intelligenz" [engl.: Artificial Intelligence]. Online job postings differ in their layout, their content and the language they use, as job- or company-specific layouts and terminologies may be used. Nevertheless, each job posting included a job title, information on the company's offers to potential candidates (e.g., benefits), a description of the candidate's profile, and the tasks associated with the job [17]. A sample of 466 tasks was randomly selected and labeled by two domain experts according to the different levels of Bloom's Taxonomy. Since the lowest level Remember("Erinnern") was never assigned by the expert labelers, the labeling range starts at the second level of Bloom's Taxonomy, meaning Understand ("Verstehen"), Apply ("Anwenden"), Analyse ("Analysiere"), Evaluate ("Bewerten"), Create ("Entwickeln"). For ease of reading, we will refer to these levels as L2, L3, L4, L5, and L6, respectively. Figure 1 shows the distribution of these labeled tasks, which is [33, 208, 36, 49, 140] for levels [L2, L3, L4, L5, L6], respectively. In a semi-supervised manner, these tasks will be used to train the models which will generate the pseudo-labels for 1500 unlabeled tasks.

B. Semi-Supervised Learning Approach

Inspired by the work of Kumar et al. (2021), which makes it possible to implement semi-supervised learning without the use of thresholds [16], we have developed a new approach for producing pseudo-labels. Instead of training a binary classifier for each class, we successively group classes (levels) with

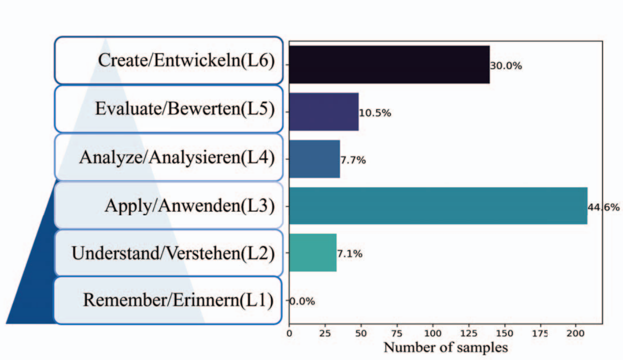

Fig. 1. Distribution of the labeled data on each level of the cognitive domain of Bloom's Taxonomy.

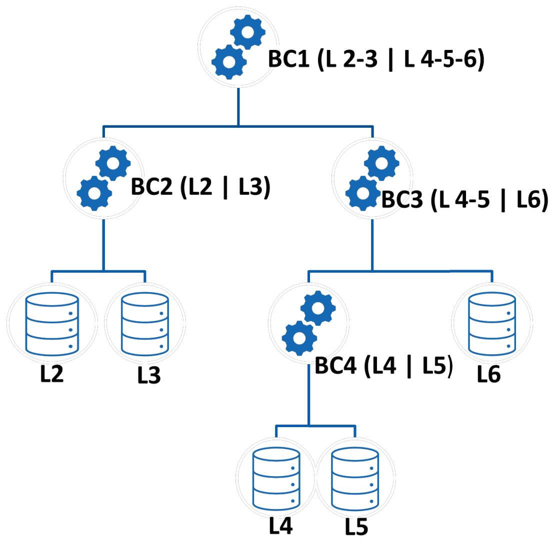

Fig. 2. Filter tree for generating pseudo-labels. Four binary classifiers are trained, BCn with n = [1...4].

similar samples until we obtain two class groups, which are then used to train the first binary classifier, reducing the problem of class imbalance. The process is then repeated in each class group until pseudo labels are assigned to the unlabeled data. A total of n-1 binary classifications will be trained to produce pseudo-labels for n classes.

The process is presented in detail in Algorithm 1: First select the least represented class (level), then merge it with the one with which it is most similar. Repeat the process until two groups of classes are obtained, which are then used to train a binary classifier. In each of the class groups obtained, algorithm 1 is reapplied.

Recursive application of Algorithm 1 produces a filter tree as shown in Figure 2, which should be read from top to bottom and from left to right.

The grouping of the levels was done based on the cosine similarity of their vector representations. The vector representations were derived by generating BERT sentence embeddings for each task description of a level and calculating their mean. The cosine similarity is given by the formula:

$$\cos(L_i, L_j) = \frac{L_i \cdot L_j}{\|L_i\| \cdot \|L_j\|} \tag{1}$$

where L_i, L_j are the vector representation of tasks at level i, j. With i, j in {2,3,4,5}.

Algorithm 1 binary Partition

Input: multi_classes: list of tasks for n levels ($n \geq 2$)
Output: binary_Classes: list of tasks for 2 groups of levels
BC: trained binary classifier

$weak\_class \leftarrow minClass(multi\_classes)$ ▷ select the least represented class
$rest_classes \leftarrow$multi_classes – $weak\_class$
for $<class \in rest\_classes>$ **do**
    Compute cosine similarity($class, weak\_class$)
    Choose most_similar_class
end for
merge_classes = merge($weak\_class, most\_similar\_class$)
remove $weak\_class$ and $most\_similar\_class$ from $multi\_classes$
insert $merge\_classes$ into $multi\_classes$
if len($multi\_classes \geq 2$ **then**
    goto 1
end if
$binary\_classes \leftarrow multi\_classes$
train($BC, binary\_classes$) ▷ train the binary classifier BC
savemodel($trainedBC$)
return $binary\_classes, BC$

We trained four binary classifiers BCn with n = [1...4] to produce pseudo-labels. In the first experiment (Exp1), we merged these pseudo-labels with the original labels to train a multi-class classification model in a semi-supervised manner. In the second experiment (Exp2), the generated pseudo-labels were evaluated by experts and then a fully supervised training is performed on the enhanced dataset.

C. Training BERT Model

For hyper-parameter tuning, data from the binary classifiers, as well as the final multi-class classifier (MCC), were divided into training, validation, and test data.

For binary classifiers involving several levels in a class, we want to make each level stand out proportionally in the test set for a better analysis. This is the case for BC1 in which we took 20% of each level for the test set, giving: Test set **BC1** = 49 (7 L2 + 42 L3) + 46 (7 L4 + 10 L5 + 29 L6). This is also the case for BC3 where we have 25% of each level for the test set, leading to: test set **BC3** = 22 (9 L4 and 13 L5) + 35 (L6).
For other binary classifiers, a balanced test set corresponding to 30% of the least represented level is used once to assess the classifier at the end of training (this corresponds to 10 tasks per level in the multi-class classifications). To address the problem of class imbalance in the training data, we used three operations of "Easy data augmentation" (EDA) [18], namely "Synonym Replacement" (SR), which consists of randomly choosing a word and replacing it with one of its synonyms, "Random Insertion" (RI), which consists of inserting a synonym of a randomly chosen word in a random position, and finally "Random Swap" (RS), which consists of swapping the positions of two randomly chosen words [18]. Synonyms of German words have been randomly selected from OpenThesaurus[1]. Usually, applying EDA to a text results in a paraphrase of that text.
In addition, we selected a maximum value of 181 tasks corresponding to the median value of the number of tasks in the pseudo-labels obtained, to merge with the original labels in order to train the final multi-class classifier.
The augmented training was finally divided into 20% for validation and 80% for the final training set. Table I summarizes the splitting process.

TABLE I
DATASET SPLITTING: NUMER OF SAMPLES IN TRAIN, VALIDATION AND TEST SET FOR THE DIFFERENT BINARY CLASSIFIERS AND THE FINAL MULTI-CLASS CLASSIFIER (MCC). THE TRAIN AND VALIDATION DATASET WERE AUGMENTED WITH "EASY DATA AUGMENTATION" (EDA) [18].

Models	Test Data	Train data + EDA	
		Train	Validation
BC1	95	307	77
BC2	20	318	80
BC3	57	165	41
BC4	22	160	40
MCCs	50	Train + Pseudo labels + EDA	
		1516	379

For fine-tuning of the BERT sequence classification model, All combinations of the following values were tested to optimize the hyperparameters: Batch size = [2,4], number of epochs = [2,3,4], Evaluation strategy = "epoch" to see the performance at each epoch, and Learning rate = [1e-4, 1e-5, 5e-5] with Adam optimizer and cross-entropy loss.

D. Evaluation Metrics

We have evaluated the performance of classification models by calculating standard performance measures: precision, which measures the correctness of positive predictions; recall or true-positive rate, which is the rate of positive instances successfully recognized by the classifier; F1 score, which is the harmonic mean of precision and recall; and accuracy, which gives the rate of correct predictions.

IV. RESULTS AND DISCUSSION

A. Binary Classifiers: Pseudo-labels

The performance of the four binary classifiers trained to produce pseudo labels is shown in Table II. The first binary

[1]https://www.openthesaurus.de/

classifier (BC1) involving the aggregation of levels L2-3 on the one hand and levels L4-5-6 on the other has an accuracy of 81%. Tasks classified as L2-3 are subjected to the second binary classifier BC2, which has an accuracy of 60%. We assume that this poor performance is due to the fact that the model has difficulty differentiating between tasks in L2 and those in L3 which is further enhanced by the problem of class imbalance. Although the application of EDA allows us to produce additional tasks, it obviously does not completely solve the problem of class imbalance, as the tasks produced lack feature variety. The model is, therefore, able to learn sufficiently the characteristics linked to tasks at the over-represented level L3, but is unable to learn those of tasks at the least-represented level L2. This justifies a high recall against a low precision at level L3, as well as a high precision against a low recall at level L2.

Tasks classified as L4-5-6 by BC1 are then passed through the binary classifier BC3 to separate L4-5 tasks from L6 tasks. BC3 has an accuracy of 91%. The L4-5 tasks are finally classified by the binary classifier BC4. This has an accuracy of 81%. Generally, we can observe that tasks at higher levels are easier to classify than those at lower levels of Bloom's Taxonomy.

TABLE II
CLASSIFICATION REPORT OF THE BINARY CLASSIFIERS

Model		Evaluation metrics on test set			
		Precision	Recall	F1-Score	Accuracy
BC1	2-3	0.84	0.78	0.81	**0.81**
	4-5-6	0.78	0.85	0.81	
BC2	2	0.75	0.30	0.43	**0.60**
	3	0.56	0.90	0.69	
BC3	6	0.92	0.94	0.93	**0.91**
	4-5	0.90	0.86	0.88	
BC4	4	0.99	0.64	0.84	**0.82**
	5	0.73	0.99	0.85	

B. Multi-Class Classifiers

In the first experiment based on semi-supervised learning, the model was trained on a large dataset comprising both data with correct labels and data with pseudo-labels. It achieved an accuracy of 76% as shown in Table III. Levels L3 "Apply" and L2 "Understand" are the lowest in terms of precision and recall respectively. This is certainly justified by the poor performance of the binary classifier (BC2) used to generate the pseudo-labels for these two levels.

In the second experiment, in a further iterative step the pseudo-labels were evaluated by experts. Of the 1500 pseudo-labeled data, 81 were assessed as unclassifiable with Bloom's Taxonomy, and of the remaining 1419, 1249 were correctly labeled with the filter tree, yielding an effective accuracy of 88%. This demonstrates the effectiveness of our approach based on the successive binary classification described in the previous section. The revised pseudo-labeled data was then merged with the original data and a fully supervised learning was performed on the enhanced dataset. The model achieved an accuracy of 78% as shown in Table IV. This result is very close to that obtained with the semi-supervised approach, which justifies the use of semi-supervised learning. As in the first experiment Exp1, the levels "Apply" and "Understand" always have the lowest precision and recall, respectively.

TABLE III
CLASSIFICATION REPORT EXP1.

	Precision	Recall	F1Score	Support
Create	0.73	0.80	0.76	10
Evaluate	0.73	0.80	0.76	10
Analyse	0.86	0.60	0.71	10
Apply	0.67	1.00	0.80	10
Understand	1.00	0.60	0.75	10
Accuracy			0.76	50
Macro avg	0.80	0.76	0.76	50

TABLE IV
CLASSIFICATION REPORT EXP2.

	Precision	Recall	F1Score	Support
Create	0.82	0.90	0.86	10
Evaluate	0.78	0.70	0.74	10
Analyse	0.78	0.70	0.74	10
Apply	0.64	0.90	0.75	10
Understand	1.00	0.70	0.82	10
Accuracy			0.78	50
Macro avg	0.80	0.78	0.78	50

C. Evaluation of the classification model

The performance of both, the final fully-supervised model and the semi-supervised approach, was adversely affected by the absence of a large number of tasks at level L2 "Understand" and by the models' difficulty in distinguishing tasks at level L3 "Apply" from other tasks. Indeed, the confusion matrices for the Exp1 (Fig. 3) and Exp2 (Fig. 4) show that all tasks at level 3 were misclassified with the other levels at least once, and tasks at level 2 are often confused with other levels. This result was to be expected, given the poor performance of the binary classifiers in classifying these levels.

The implementation of the proposed semi-supervised method is very time-consuming. It took around 20 hours longer than the supervised method. This is due to the filter tree, which requires the training of 4 binary classifiers to produce the pseudo-labels. This method would therefore be optimal for large quantities of unlabeled data. For small quantities of unlabeled data, manual labeling would be more beneficial.

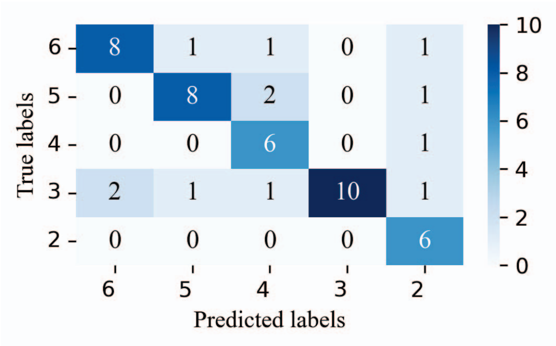

Fig. 3. Confusion matrix for MCC Exp1.

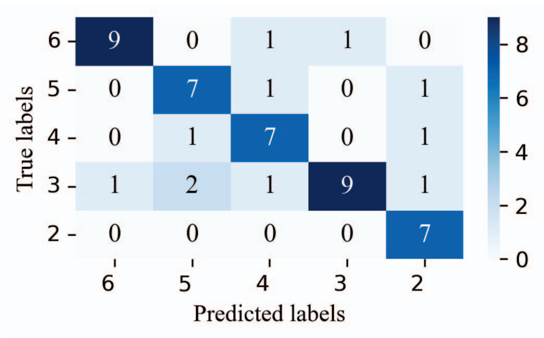

Fig. 4. Confusion matrix for MCC Exp2.

Fig. 5. Confusion matrix for Pseudo-labels

The slight outperformance of the fully supervised approach over the semi-supervised approach was expected, as all labels were verified by experts in the fully supervised approach. However, the performance increase is only marginal. The result of this verification shows that tasks at level L2 were the most misclassified by the filter tree (s. Fig. 5). This could justify the improvement of the F1-score at level L2 from 75% to 82% in particular, and the overall improvement of the model in Exp2.

D. Analysis of AI-related job requirements

As with the models, the two domain experts struggled to classify the tasks, since most of them were not clearly defined. This has been very time-consuming, sometimes requiring the involvement of an additional expert to corroborate the classification. The final analysis and manual classification show that almost none of the AI-related jobs were assigned to the lower levels of Bloom's taxonomy (levels 1 and 2). Instead, the majority of tasks corresponded to level 3 “Apply”, followed by level 6 “Create”. This suggests that AI-related jobs require fairly high cognitive skills that involve practice, either using tools and algorithms that are already available, or developing one's own tools to solve specific problems.

V. Conclusion

The goals of this work have been an automatic classification of tasks in AI-related job offers using Bloom's Taxonomy and an analysis of their requirements. Due to the lack of expert-labeled data, we first implemented a semi-supervised learning approach, which was then compared to a fully supervised training on the dataset obtained by combining the small amount of initial data and pseudo-labeled data proced by the first method and reviewed by experts. The results of these two experiments show that both models can classify AI-related tasks with encouraging performance (up to 76% accuracy with semi-supervised learning), supporting the use of semi-supervised learning. Upon closer examination, it was evident that the models struggled to differentiate between tasks at the lower levels of Bloom's Taxonomy, specifically ”Apply” and ”Understand”, which had a negative impact on the overall performance of the multi-class classifier. Another aspect that had an impact on the models was the problem of class imbalance. This was partially solved by paraphrasing the available text data, but the results show that a larger dataset with a roughly even distribution would have produced better results. Nevertheless, the results of our research show that it is possible to use deep learning together with semi-supervised learning to automatically classify tasks in AI-related job offers using Bloom's Taxonomy.

Future work could examine other deep learning models such as LSTM and CNN for the classification of tasks in AI-related job offers, or implement a multi-label classification, as some tasks may contain subtasks belonging to different levels of Bloom's Taxonomy. Furthermore, the effect of an enlarged dataset should be investigated containing better quality labels at the lower levels of Bloom's Taxonomy.

References

[1] Mariagrazia Squicciarini and Jacopo Staccioli. “Labour-saving technologies and employment levels: Are robots really making workers redundant? ”. In: OECD Science, Technology and Industry Policy Papers, no. 124, (Jan. 2022). DOI: https://doi.org/https://doi.org/10.1787/9ce86ca5-en.

[2] Daron Acemoglu and Pascual Restrepo. “Artificial Intelligence, Automation and Work” In: The Economics of Artificial Intelligence: An Agenda, Eds. Ajay Agrawal and Joshua Gans and Avi Goldfarb, 2019. URL: https://www.nber.org/books-and-chapters/economics-artificial-intelligence-agenda.

[3] Giacomo Damioli et al. “AI technologies and employment: micro evidence from the supply side”. In: Applied Economics Letters 30.6 (2023), pp. 816–821. DOI: 10.1080/13504851.2021.2024129

[4] Anderson Lorin W. et al. Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom’s Taxonomy of Educational Objectives. Ed. by Anderson Lorin W. and Krathwohl David R. Addison Wesley Longman, 2001.

[5] Sarang Shaikh, Sher Muhammad Daudpotta, and Ali Shariq Imran. “Bloom’s Learning Outcomes’ Automatic Classification Using LSTM and Pretrained Word Embeddings”. In: IEEE Access 9 (2021), pp. 117887-117909. DOI: 10.1109/ACCESS.2021.3106443

[6] Lea Samek, Mariagrazia Squicciarini, and Emile Cammeraat. “The human capital behind AI: Jobs and skills demand from online job postings”. In: OECD Science, Technology and Industry Policy Papers, no. 120, (Sept. 2021). URL: https://www.oecd-ilibrary.org/content/paper/2e278150-en.

[7] Andrea De Mauro et al. “Human resources for Big Data professions: A systematic classification of job roles and required skill sets”. In: Information Processing & Management (2018). In (Big) Data we trust: Value creation in knowledge organizations, pp. 807–817. DOI: https://doi.org/10.1016/j.ipm.2017.05.004.

[8] Benjamin S. Bloom et al. Taxonomy of educational objectives. The classification of educational goals. Handbook 1: Cognitive domain. New York: David Mckay, 1956.

[9] Anwar Yahya and Addin Osman. “Automatic Classification of Questions into Bloom’s Cognitive Levels using Support Vector Machines”. Dec. 2011.

[10] James Zhang et al. “Automated Classification of Computing Education Questions Using Bloom’s Taxonomy”. In: Proceedings of the 23rd Australasian Computing Education Conference. Virtual, SA, Australia: Association for Computing Machinery, 2021, pp. 58–65. URL: https://doi.org/10.1145/3441636.3442305.

[11] Yuheng Li et al. “Automatic classification of learning objectives based on Bloom’s taxonomy”. In: Proceedings of the 15th International Conference on Educational Data Mining. Ed. by Antonija Mitrovic and Nigel Bosch. International Educational Data Mining Society, 2022, pp. 530–537. DOI: 10.5281/zenodo.6853191.

[12] Dong-Hyun Lee. “Pseudo-Label : The Simple and Efficient Semi-Supervised Learning Method for Deep Neural Networks”. In: ICML 2013 Workshop : Challenges in Representation Learning (WREPL) (July 2013).

[13] Vivian Lay Shan Lee et al. “Semi-supervised Learning for Sentiment Classification using Small Number of Labeled Data”. In: Procedia Computer Science 161 (2019). The Fifth Information Systems International Conference, 23-24 July 2019, Surabaya, Indonesia, pp. 577–584. URL: https://www.sciencedirect.com/science/article/pii/S1877050919318708.

[14] Samujjwal Ghosh and Maunendra Sankar Desarkar. “Semi-Supervised Granular Classification Framework for Resource Constrained Short-Texts: Towards Retrieving Situational Information During Disaster Events”. In: Proceedings of the 12th ACM Conference on Web Science. Association for Computing Machinery, 2020, pp. 29–38. URL: https://doi.org/10.1145/3394231.3397892.

[15] Yi Xu et al. “Dash: Semi-Supervised Learning with Dynamic Thresholding”. In: Proceedings of the 38th International Conference on Machine Learning. Ed. by Marina Meila and Tong Zhang. Proceedings of Machine Learning Research. PMLR, 2021, pp. 11525–11536. URL: https://proceedings.mlr.press/v139/xu21e.html.

[16] Teerath Kumar et al. “Binary-Classifiers-Enabled Filters for Semi-Supervised Learning”. In: IEEE Access 9 (2021), pp. 167663–167673. DOI: 10.1109/ACCESS.2021.3124200.

[17] Timm Eichenberg et al. “Developing personas of ideal-type candidates in AI-related jobs: An exploratory study based on the analysis of online job postings”, unpublished.

[18] Jason Wei and Kai Zou. “EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks”. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 6382–6388. URL: https://aclanthology.org/D19-1670

ACKNOWLEDGEMENT

The authors are grateful to Ulrich Stockem for expert labeling support.