*future internet*

MDPI

*Article*

# Anticipating Job Market Demands—A Deep Learning Approach to Determining the Future Readiness of Professional Skills

**Albert Weichselbraun [1,2,*], Norman Süsstrunk [1], Roger Waldvogel [1], André Glatzl [1], Adrian M. P. Braşoveanu [3,4] and Arno Scharl [2,3]**

[1] Swiss Institute for Information Research, University of Applied Sciences of the Grisons, Pulvermühlestrasse 57, 7000 Chur, Switzerland; norman.suesstrunk@fhgr.ch (N.S.); roger.waldvogel@fhgr.ch (R.W.); andre.glatzl@fhgr.ch (A.G.)
[2] webLyzard technology, Liechtensteinstrasse 41/26, 1090 Vienna, Austria; scharl@weblyzard.com
[3] Research Center of New Media Technology, Modul University Vienna, Am Kahlenberg 1, 1190 Vienna, Austria; adrian.brasoveanu@modul.ac.at or brasoveanu@modultech.eu
[4] Modul Technology, Am Kahlenberg 1, 1190 Vienna, Austria
* Correspondence: albert.weichselbraun@fhgr.ch; Tel.: +41-81-286-3727

**Abstract:** Anticipating the demand for professional job market skills needs to consider trends such as automation, offshoring, and the emerging Gig economy, as they significantly impact the future readiness of skills. This article draws on the scientific literature, expert assessments, and deep learning to estimate two indicators of high relevance for a skill's future readiness: its automatability and offshorability. Based on gold standard data, we evaluate the performance of Support Vector Machines (SVMs), Transformers, Large Language Models (LLMs), and a deep learning ensemble classifier for propagating expert and literature assessments on these indicators of yet unseen skills. The presented approach uses short bipartite skill labels that contain a skill topic (e.g., "Java") and a corresponding verb (e.g., "programming") to describe the skill. Classifiers thus need to base their judgments solely on these two input terms. Comprehensive experiments on skewed and balanced datasets show that, in this low-token setting, classifiers benefit from pre-training and fine-tuning and that increased classifier complexity does not yield further improvements.

**Keywords:** skill classification; deep learning; large language models; bipartite skill labels



**Citation:** Weichselbraun, A.; Süsstrunk, N.; Waldvogel, R.; Glatzl, A.; Braşoveanu, A.M.P.; Scharl, A. Anticipating Job Market Demands—A Deep Learning Approach to Determining the Future Readiness of Professional Skills. *Future Internet* **2024**, *16*, 144. https://doi.org/10.3390/fi16050144

Academic Editor: Filipe Portela

Received: 8 March 2024
Revised: 10 April 2024
Accepted: 15 April 2024
Published: 23 April 2024



## 1. Introduction

Automation, offshoring, and the emerging Gig economy instigate and intensify labor market disruptions. The ongoing trends of automating repetitive tasks and offshoring continue to reshape traditional job roles and workforce dynamics, changing skill requirements in the labor market. Research by Bick et al. [1] indicates that 60% of occupations have at least 30% work activities that could be automated.

Automation has triggered a growing demand for technical skills such as data analysis, artificial intelligence, and machine learning, while offshoring amplifies the relevance of intercultural communication and global collaboration skills. The rise of the Gig economy accentuates the significance of adaptability, self-management, and entrepreneurship, particularly as individuals navigate short-term projects and roles. Gig economy platforms like Fiverr, UpWork, and Freelancer, compel both employers and employees to adapt to more flexible working structures. However, as certain tasks become automated or outsourced, routine skills for repetitive operations witness declining demand. This underscores the need for continuous upskilling and reskilling in increasingly competitive job markets.

This article aims to provide decision-makers with insights into the future readiness of professional skills. The presented approach resonates with the objectives outlined in the United Nations Sustainable Development Goals (SDGs) (https://sdgs.un.org/goals, accessed on 1 March 2024) by (i) providing information on a skill's future readiness to

guide educational activities and increasing the number of people with relevant skills (SDG 4: Equitable Quality Education) and (ii) helping to better align skill supply and demand to promote sustainable economic growth and productive employment (SDG 8: Decent Work and Economic Growth; see Section 7).

Evaluating a skill's future readiness in terms of its resilience to automation and outsourcing requires the development of classifiers capable of automatically assessing the skill across these two dimensions. The presented work draws upon a skill ontology that characterizes skills with bipartite skill labels consisting of skill topic and skill verb (e.g., *Java*$^{[topic]}$ *Programming*$^{[verb]}$). These labels, more detailed than topic-only labels, still lack additional context and challenge classifiers due to their reliance on just two topics for predictions. This impacts the design and effectiveness of classifiers, as ensemble models might fail to improve the outcome with such limited input.

Building on previous model comparisons for assessing the future readiness of skills [2], the presented research has been conducted within the Future of Work project (https://semanticlab.net/future-of-work, accessed on 1 March 2024), which investigates the performance of multiple classification approaches (Support Vector Machines (SVMs), Transformers, Large Language Models (LLMs), and a deep learning ensemble classifier) toward reliably propagating expert and literature assessments on automatability and offshorability to yet unseen skills.

The remainder of this paper is structured as follows: Section 2 provides an overview of related work which is followed by a description of the methods applied in this study (Section 4). Section 5 presents the gold standard datasets, evaluation settings, and evaluation results, followed by the conclusion and outlook in Section 7.

## 2. State of the Art

The literature discussed in this section is focused on (i) anticipating job market demands; (ii) skill classification systems with a focus on skill bases such as taxonomies and ontologies, custom skill bases for automation and custom skill bases for offshorability; and (iii) a brief overview of recent deep learning models for classification, such as Transformer models and LLMs.

### 2.1. Anticipating Job Market Demands

Anticipating job market demands is a difficult proposition. The lockdown measures imposed during the COVID-19 pandemic, for instance, triggered declines in labor demand of up to 30% [3]. A 2023 study estimates that only 30.7% of current jobs are likely to remain unaffected by disruptions caused by generative artificial intelligence (AI) such as ChatGPT. The study's authors expect that a majority of jobs will either be fully or partially impacted by generative AI [4]. Another 2023 study states that Industry 5.0 requires highly skilled individuals with various soft skills (e.g., communication, teamwork, emotional intelligence) capable of collaborating with both humans and machines [5]. Such economic megatrends tend to disrupt and reshape job markets, often acting together (e.g., the twin impact of COVID-19 and AI [6]) and requiring both employers and workers to adapt.

A recent survey that used data mining techniques to predict student employability [7] identifies the following challenges: (i) focus on gender instead of psychometric attributes; (ii) unbalanced and incomplete datasets; (iii) studies based on heuristics; (iv) scalability is an issue as participants usually come from the same environment (e.g., universities); and (v) reproducibility since data and source code are not publicly available. Most of the studies investigated in the survey center on students and lack integration of online data sources, such as job advertisements or CVs. This tendency could potentially be attributed to the nature of the study cohort. For instance, there might be fewer online job opportunities directly targeting students. Notably, studies focusing on workers, as exemplified by [8], predominantly draw from online sources.

Khaouja and his group [8] survey skill identification technology [8], one of the pillars of predicting job market demands. They define the following objects: (i) online data

sources (e.g., job ads, academic curricula, CVs); (ii) skill bases developed by experts (e.g., public ontologies and taxonomies) and customized skill bases (e.g., manually built or based on embeddings); (iii) skill identification methodologies (e.g., skill count, topic modeling, embeddings, ML-based); (iv) evaluation metrics (e.g., precision, recall, f1 for binary classification tasks or MRR for multi-label classification tasks); (v) skill identification granularity (e.g., sentences, n-grams, sentences and n-grams); and (vi) industry sectors (e.g., IT, engineering, healthcare, multiple sectors). The second part of the survey focuses on the various applications of this technology, such as market analysis, curricula development, job recommendation engines, talent search, skill demand prediction and gender bias identification. The article ends with an overview of future work, which includes deep learning, graph embeddings, or generation of skill bases.

Most of the articles present global trends, but job demand forecasting is typically local. Some recent studies, for example, focused on analyzing the Norwegian IT market to improve the computing curriculum [9] or assessing skill demand in the Lithuanian market by analyzing the job ads with clustering and NLP techniques [10].

### 2.2. Skill Classification

Skill bases appeared due to the necessity of classifying jobs listed in statistical reports. The first classification schema focused on occupations, whereas the most recent ones are developed around skills.

Custom-built skill bases allow assessing skills regarding specific use cases, such as assessing their susceptibility to automation. They can be manually built (e.g., lists of predefined terms), extracted from word embeddings, or even generated from existing knowledge bases. The remainder of the section discusses popular skill bases as well as two types of customized skill bases that focus on automatability and offshorability.

#### 2.2.1. Skill Bases

Public skill classifications and skill bases developed by experts are good starting points for the discussion of skill classification [8]. Well-known classification schemes include:

- ESCO (European Skills, Competences, qualifications, and Occupations; https://esco.ec.europa.eu/en/classification, accessed on 1 March 2024), a fine-grained classification scheme that covers 14,295 skills in all the languages of the EU. It is the main classification scheme used in the European labor market.
- O*NET (https://www.onetonline.org, accessed on 1 March 2024), a coarse-grained classification schema which covers over 900 skills and also includes generic skills and is mainly used in the United States.
- ROME (https://www.francetravail.fr/employeur/vos-recrutements/le-rome-et-les-fiches-metiers.html, accessed on 1 March 2024), a coarse-grained classification developed by France Travail. It is available in French and only covers French territories.
- ISCO-08 (https://www.ilo.org/public/english/bureau/stat/isco/index.htm, accessed on 1 March 2024) is the latest version (December 2007) of a job classification scheme which was adopted as early as 1957.

These skill bases are built around ontologies and supported by organizations which are publicly funded by the US (e.g., O*NET), the European Union (e.g., ESCO), or national governments (e.g., ROME is supported by the French government).

Skill identification technology [8] has been a foundational technology for building these widely used skill bases. Nevertheless, special use cases such as forecasting and predicting the impact of trends such as automation and offshoring may require custom classification systems, such as the one discussed in the remainder of this section.

#### 2.2.2. Customized Skill Bases for Automation

Autor and Dorn [11] argue that jobs containing repetitive tasks are more likely to be automated. Josten and Lordan [12] identify a set of indicators that might impact automation, namely people (e.g., if the job requires daily interaction with people), brains

(e.g., if abstract reasoning is required), and brawn (e.g., if physical interaction with certain objects is needed). Josten and Lordan [13] classified O*NET professions based on their degree of automatability (e.g., full, non-, or partial). The main goal of their work was better alignment with the European labor force survey (https://ec.europa.eu/eurostat/web/microdata/european-union-labour-force-survey, accessed on 1 March 2024), which covers the period between 2013 and 2016. Based on a regression analysis, they established that occupations that require brains (e.g., abstract reasoning) are better protected from automation, especially when compared to occupations that require daily interaction (e.g., people skills) and physical interaction (e.g., brawn). Combining these factors can also decrease the likelihood of automation.

Eloundou et al. [14] investigated the impact of LLMs such as GPT and BLOOM [15] on the labor market. They note that routine and repetitive tasks have a high risk of technology-driven displacement. Brynjolfsson et al. [16] distinguish between the labor-augmenting and labor-displacing effects of automation. Eloundou et al. [14] use exposure as the main factor in their automation risk classification. They consider three cases: no exposure (e.g., minimal or no time reduction for completing a task using an LLM or software agent), direct exposure (e.g., LLMs reduce task completion times by 50 percent), and indirect exposure (e.g., if productivity rises and can be doubled with the help of a software agent). Routine tasks in automatable domains such as data processing, data science, or information processing exhibit a high chance of being displaced in contrast to agriculture, manufacturing, or mining tasks. ChatGPT shows promising results in a wide variety of programming tasks [17], supporting the hypothesis that programming and technology-related jobs will be disrupted by automation. However, the idea that people need domain knowledge to author software is also gaining traction, suggesting professional reconversion may be a solution to averting job losses [18].

Nevertheless, considering previous technological breakthroughs in areas such as agriculture and the industrial revolution leads to the conclusion that the overall impact of AI is difficult to assess at this point since it might take several decades to unfold, as it requires the development of new processes and business models [19,20].

#### 2.2.3. Customized Skill Bases for Offshorability

Wagner's work [21] on digital talent platforms (e.g., Freelancer, Upwork and Fiveer) provided important insights towards preparing the offshorable task gold standard. Digital talent platforms help employers meet unplanned needs for knowledge work services [22]; lower the need for permanent positions by hiring specialized workers for specific contracts [23]; and fill hiring gaps that are not addressed by traditional hiring strategies [24].

Dunn [25] classifies Gig economy platforms into: (i) low-skill (e.g., Uber, Bold, TaskRabbit) or high-skill location-dependent (e.g., Outschool or Tutoroo for private lessons) and (ii) low-skill (e.g., Amazon Mechanical Turk) or high-skill (e.g., Fiverr and Upwork) location-independent services. These services can be used as starting points for assessing offshorability, as all the tasks listed in their catalogs have already been offshored successfully.

While this article focuses on the effects of automation on offshoring (relocating manufacturing to other countries), the opposite strategy, reshoring (relocating manufacturing back home) is equally relevant. Pinheiro et al. [26] conducted a meta-regression analysis of the published research. They analyze four major automation trends (cost advantage, increased productivity, robots, and Industry 4.0) and their influence on offshoring and reshoring. After the pandemic, companies seem to favor reshoring, but they do use offshoring in cases where automation plays a central role (e.g., information technology). They argue that the offshoring vs. reshoring decision depends on internal decision-making chains rather than global trends.

### *2.3. Deep Learning for Text Classification*

The Transformer architecture [27] has changed the landscape of NLP, as almost all the classic NLP tasks (e.g., classification, dependency parsing, sentiment analysis, named

entity recognition, or question answering) can be implemented with it. Transformer-based language models (e.g., BERT [28], DistilBERT [29], or RoBERTa [30]) employ a self-attention mechanism to capture the context and relationships among words. This self-attention mechanism enables them to assess the significance of individual words within a sentence, prioritizing semantically meaningful tokens while filtering out irrelevant noise [31]. Bommasani [32] even names pre-trained Transformer foundation models because most NLP tasks can be designed around them. Still, they also require adaptation to domain-specific tasks (e.g., text classification, sentiment analysis, etc.).

A large-scale survey by Minaee et al. [33] presents most of the deep learning architectures widely used for text classification, from LSTMs to Transformers. A taxonomy of Transformers can be found in [27] and includes most models used for classification until late 2021. Some specialized surveys are also available. One survey [34] examines the role of embeddings in text classification. The last few years have also seen the rise of hybrid architectures that combine sequence-to-sequence or graph neural networks with Transformers, as described in Pham et al. [35]. Another recent survey [36] examines text classification models in the context of designing spam filters.

Large Language Models (LLMs) exceed Transformers in size (i.e., over 10 billion parameters [37]) and apply training strategies such as instruction tuning and adaptation tuning to enable instruction following and zero-shot capabilities. The GPT 3/4 models (https://chat.openai.com, accessed on 1 March 2024) [38] inhibit so-called emerging capabilities which further improve their capability to correctly interpret human language and, therefore, pave the way for even more advanced text classification systems [37].

Using AI tools for classification or related generative processes requires considering issues such as transparency and accountability. Since generative tools can easily reuse or remix text, code, or images on demand, it is important to know more details about the training process so that the generated artifacts can be traced back to the training corpora. An early review of LLM transparency and accountability can be found in [39].

## 3. Problem Statement

The recent literature highlights the absence of a unified skill framework that includes technical and soft skills and of a clear structure that covers skill gaps, shortages, and mismatches as gaps reported in the state of the art [40]. The presented work aims to address this gap with a method capable of assessing the future readiness of professional skills, considering the following two dimensions: (i) automatability (i.e., the extent to which tasks can be performed by automated systems) and offshorability (i.e., the feasibility of performing tasks off-site, often for cost-saving purposes). We therefore develop automatic classifiers that assess a skill's automatability and offshorability based on its label. The classifier then complements and significantly extends manual classifications provided in a gold standard dataset that has been assembled by domain experts based on their assessments and the scientific literature (Section 5.1.1). Automatic classification techniques provide significant cost and time savings compared to manual annotation processes.

Khaouja et al. [8] distinguish three different levels of skill label granularity: words (e.g., "Java"), multi-word phrases (e.g., "Java Programming"), and sentences (e.g., "Experience in software development, particularly the development of Java Web applications."). The presented approach draws upon German bipartite skill labels that are part of a proprietary occupation knowledge base which formalizes domain knowledge [41] in the human resources domain. These labels are multi-word but limited to skill topic and verb (e.g., $Java^{[topic]}$ $Programming^{[verb]}$) and do not convey any additional context information. Nevertheless, bipartite skill labels are still considerably more expressive than standard skill specifications which operate on word granularity, only comprise a topic, and therefore cannot distinguish nuances such as the difference between $Java^{[topic]}$ $Programming^{[verb]}$ and $Java^{[topic]}$ $Teaching^{[verb]}$. Table 1 lists example skills classified by their automatability and offshorability, taken from the customized skill bases.

**Table 1.** Bipartite skill labels, translations, and classification of their automatability and offshorability.

| **Skill Label (German)** | **Skill Label (English Translation)** | **Automatable** |
|---|---|---|
| Vorsorgestrategie erstellen | Develop pension strategy | Yes |
| Reden anpassen | Adapt speeches | Yes |
| Marketingkonzept planen | Plan marketing concept | Yes |
| Gebäude instandhalten | Maintain building | No |
| Audit beaufsichtigen | Supervise audit | No |
| Musicalproduction anleiten | Direct musical production | No |
| **Skill Label (German)** | **Skill Label (English Translation)** | **Offshorable** |
| Ausbildungsplan erstellen | Develop training plan | Yes |
| Marketingkonzept planen | Plan marketing concept | Yes |
| Sendung überwachen | Monitor broadcast | Yes |
| Blutentnahme vorbereiten | Prepare blood extraction | No |
| Gebäude instandhalten | Maintain building | No |
| Elektromotor montieren | Assemble electric motor | No |

The absence of supplemental context information presents a significant challenge since classifiers solely rely on the two terms used in the bipartite skill labels for predictions. This limitation directly impacts classifier design and performance (Section 5). Standard approaches for improving classification performance, such as ensemble models, cannot yield better results since their enhanced generalization capabilities do not translate into better outcomes if the model input is restricted to two terms.

## 4. Method

The presented research involves developing, fine-tuning, and evaluating four methods for classifying skills represented by bipartite skill labels in regard to their offshorability and automatability. The evaluation aims to find the optimal trade-off between method complexity and performance and considers the following methods:

- A Support Vector Machine (SVM; Section 4.1) serves as a competitive baseline approach;
- The more complex Transformer-based classifier (Section 4.2), which is expected to considerably benefit from pre-training;
- An approach that builds upon a Large Language Model (Section 4.3) by leveraging ChatGPT. This method draws upon few-shot learning and has the advantage of requiring a considerably lower number of training examples;
- An ensemble model (Section 4.4) which combines a Transformer with multiple fully connected neural networks. The ensemble then employs majority voting for overall skill assessment. This is the most complex model considered in the evaluation.

### *4.1. Baseline Classifier*

A Support Vector Machine in conjunction with FastText word embeddings, the *fasttext-wiki-news-subwords-300* model from Gensim, (https://pypi.org/project/gensim, accessed on 1 March 2024) acts as a baseline classifier. The input is tokenized with the Natural Language Toolkit (NLTK) library and converted to FastText embeddings to obtain the SVM input representation. The resulting NumPy arrays train the SVM (https://scikit-learn.org/stable/modules/svm.html, accessed on 1 March 2024). A four-fold cross-validation strategy is then used for training and evaluation.

### *4.2. Transformers*

We used three Transformer models (BERT, DistilBERT, and XLM-RoBERTa) from the Hugging Face library (https://huggingface.co, accessed on 1 March 2024) in the experimental section. The following section describes the pre-tests that have been conducted on a randomly selected subset of 434 bipartite labels from the random selection gold

standard dataset (Section 5.1.1) to obtain the optimal hyperparameter settings for the Transformer classifier.

The models were trained on a large-scale multilingual corpus to improve multilingual performance and are capable of handling complex German vocabulary, idioms, and syntactic structures. The final models are implemented in PyTorch, which seamlessly integrates with the Hugging Face library and optimizes through (i) domain adaptation, (ii) model-fine tuning, and (iii) automated hyperparameter optimization.

Adapting Transformers to a target domain can lead to increased robustness to noise or better feature alignment [42]. Exposure to domain-specific documents, before fine-tuning, enables the model to closely align itself with the target text corpus. This alignment usually improves the model's understanding of vocabulary, phrasing, and linguistic nuances and reduces the likelihood of semantic misinterpretations or mismatches. A dataset of 150,366 Swiss job postings was used for domain adaptation. The dataset covered diverse industries and job roles which have been converted to text with the Inscriptis library [43].

Assessing the performance of the fine-tuned models with and without domain adaptation allows evaluating the effectiveness of the adaptation process. Table 2 summarizes the effectiveness of domain adaptation for the offshorable classification task. Table 3 presents the corresponding results for classifying a skill's automatability.

**Table 2.** Classification performance for the "offshorable" indicator with/without domain adaptation (DA) and layer freeze (LF). The best results are indicated in bold.

| Model | DA | B | f1 | Precision | Recall | Accuracy |
|---|---|---|---|---|---|---|
| bert-base-multilingual-cased | | | 0.83 | 0.80 | 0.87 | 0.76 |
| bert-base-multilingual-cased | | ✓ | 0.83 | 0.76 | **0.91** | 0.73 |
| bert-base-multilingual-cased | ✓ | | 0.83 | 0.80 | 0.87 | 0.76 |
| bert-base-multilingual-cased | ✓ | ✓ | **0.84** | 0.78 | **0.91** | 0.76 |
| distilbert-base-multilingual-cased | | | **0.84** | **0.81** | 0.88 | 0.77 |
| distilbert-base-multilingual-cased | | ✓ | 0.80 | 0.76 | 0.86 | 0.70 |
| distilbert-base-multilingual-cased | ✓ | | 0.82 | 0.77 | 0.88 | 0.84 |
| distilbert-base-multilingual-cased | ✓ | ✓ | 0.80 | 0.73 | 0.88 | 0.81 |
| xlm-roberta-large | | | 0.83 | **0.81** | 0.86 | 0.76 |
| xlm-roberta-large | | ✓ | 0.81 | 0.78 | 0.84 | 0.72 |
| xlm-roberta-large | ✓ | | 0.77 | 0.69 | 0.88 | 0.79 |
| xlm-roberta-large | ✓ | ✓ | 0.75 | 0.69 | 0.83 | **0.82** |

**Table 3.** Classification performance for the "automatable" indicator with/without domain adaptation (DA) and layer freeze (LF). The best results are indicated in bold.

| Model | DA | LF | f1 | Precision | Recall | Accuracy |
|---|---|---|---|---|---|---|
| bert-base-multilingual-cased | | | 0.71 | 0.73 | 0.69 | 0.69 |
| bert-base-multilingual-cased | | ✓ | 0.73 | 0.72 | **0.79** | 0.68 |
| bert-base-multilingual-cased | ✓ | | **0.75** | **0.76** | 0.74 | 0.73 |
| bert-base-multilingual-cased | ✓ | ✓ | 0.73 | 0.72 | 0.74 | 0.69 |
| distilbert-base-multilingual-cased | | | 0.69 | 0.70 | 0.69 | 0.66 |
| distilbert-base-multilingual-cased | | ✓ | 0.69 | 0.68 | 0.72 | 0.65 |
| distilbert-base-multilingual-cased | ✓ | | 0.67 | 0.58 | 0.78 | 0.75 |
| distilbert-base-multilingual-cased | ✓ | ✓ | 0.68 | 0.60 | **0.79** | **0.76** |
| xlm-roberta-large | | | 0.74 | 0.75 | 0.75 | 0.72 |
| xlm-roberta-large | | ✓ | 0.69 | 0.66 | 0.74 | 0.65 |
| xlm-roberta-large | ✓ | | 0.67 | 0.62 | 0.74 | 0.73 |
| xlm-roberta-large | ✓ | ✓ | 0.67 | 0.64 | 0.71 | 0.71 |

Model fine-tuning followed the common approach of freezing certain layers [44] while the remaining layers were updated with task data. Tables 2 and 3 summarize the impact of layer freezing on the model performance.

For hyperparameter optimization, we used Optuna framework [45], one of the earliest tools that offered streamlined sampling and pruning algorithms. Adding Optuna improves the efficiency of our pipeline. Table 4 provides a summary of the Transformer hyperparameters used in the experimental section.

**Table 4.** Model specification and hyperparameters

| Base Language Model | Bert Base Multilingual | DistilBERT Base | XLM Roberta |
|---|---|---|---|
| Activation | Gelu | Gelu | Gelu |
| Attention dropout | 0.1 | 0.1 | 0.1 |
| Dimension | 768 | 3072 | 1024 |
| Dropout | 0.1 | 0.1 | 0.1 |
| Hidden layer dimensions | 12 | (n.a.) | 24 |
| Initializer range | 0.02 | 0.02 | 0.02 |
| Max position embeddings | 512 | 512 | 514 |
| Learning rate | AdamW | AdamW | AdamW |

The final experiments drew upon the DistilBERT classifier (without domain adaptation and layer freeze), which provided the best results for the offshorable indicator, a decent performance for the automatable label, and required the least resources for training, therefore making it the most efficient choice for our specific task.

### *4.3. Large Language Model with Heuristic Classifier*

The LLM-based approach builds upon the GPT API to classify the automatability and offshorability of skills (Figure 1). LLMs based on GPT models are considerably more complex than Transformer models and harder to adapt to specific domains [38]. They are useful for in-context learning from prompts, especially in few-shot settings, but they are sometimes saddled with fairness issues (e.g., stereotypes, biases, errors) [46].

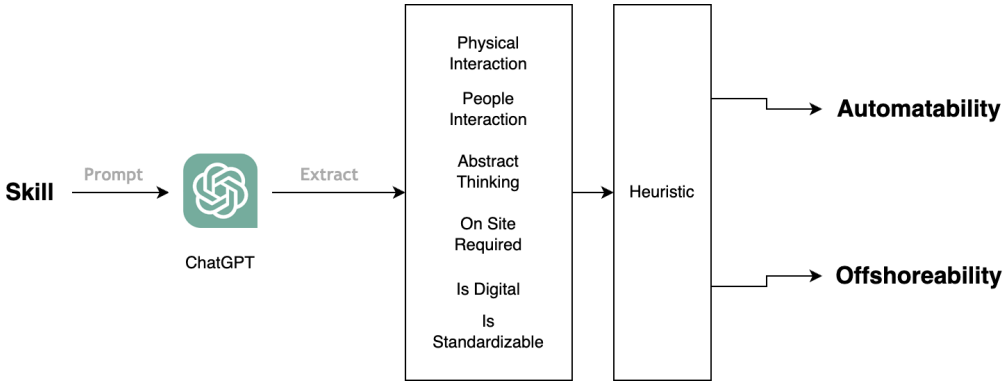



**Figure 1.** Obtaining details on skill automatability and offshorability via GPT and heuristics.

The LLM classifier described here queries a GPT model for assessments on a skill's basic characteristics as outlined by Josten and Lordan [12]: *brawn (B)*: physical interaction with objects is needed; *people (P)*: interactions with humans are required; *brain (T)*: abstract reasoning is needed; *location (L)*: the activity's location matters (e.g., customer's or own location); *digitalization (D)*: activity can be digitalized; *routine (R)*: activity can be standardized into subprocesses that are performed similarly around the world.

We use OpenAI's *gpt-3.5-turbo* model and a prompt that provides few-shot examples to contextualize the request—e.g., the next prompt helps us obtain a binary skill classification:

> Can the activity *[experiment planning]* necessarily only be carried out on-site at the customer's location? (answer only with Yes or No):
>
> - Examples/Yes: *[production order monitoring]*, *[building maintenance]*, *[object cleaning]*, *[tunnel measuring]*.
> - Examples/No: *[customer file updating]*, *[door producing]*, *[ensure analysis quality]*, *[coordinate program area]*.

The example prompt requests an assessment of whether the task *experiment planning* needs to be performed on-site and provides examples for skills with a positive (*monitor production order*, *maintain buildings*, *clean object*, and *measure tunnel*) and a negative (*update customer file*, *produce door*, *ensure analysis quality*, and *coordinate program area*) assessment.

Each prompt is built around a single indicator from Josten and Lordan's list [12], as combining the indicators was found to lead to worse results. The indicators were extracted for each skill using the prompts and then used to calculate with a heuristic whether the skill is automatable or offshorable.

We categorize a skill as automatable when the threshold value of 0.5 is surpassed for *automatable* according to the equation below:

$$automatable = 1 - (0.4 \cdot B + 0.3 \cdot P + 0.2 \cdot T + S - 0.4 \cdot D) \quad (1)$$

In this equation, all parameters ($B$, $P$, $T$, $S$, and $D$) are binary variables, taking values of either 0 or 1. A value of 0 indicates the absence of the characteristic, while a value of 1 signifies its presence. $B$ denotes the degree of physical interaction (brawn), $P$ reflects the degree of interaction with people, $T$ represents the level of abstract thinking required (brain), $D$ indicates the extent to which the task can be performed digitally (digitalization), and $L$ signifies the necessity of on-site presence (location). The weights (0.4, 0.3, 0.2, and $-0.4$) have been determined empirically.

As outlined in Equation (2), a task is considered offshorable if it does not have to be performed on-site (location $L = 0$) and can either be digitalized (digitalization $D = 1$) or standardized (routine $R = 1$). Tasks requiring physical presence (location $L = 1$) are automatically categorized as not offshorable.

$$offshorable = \begin{cases} 1 & \text{if } L = 0 \wedge (D = 1 \vee R = 1) \\ 0 & \text{if } L = 1 \end{cases} \quad (2)$$

### *4.4. Deep Learning Ensemble*

The deep learning ensemble aimed at further improving classification performance. The presented approach has been inspired by the human brain, which uses multiple interconnected neural networks that differ widely in anatomy and physiology [47] to increase accuracy and robustness.

Figure 2 offers an overview of the selected approach. In this deep learning ensemble, DistilBERT plays a central role in encoding the bipartite skill tuple into a contextual embedding. Subsequently, the ensemble leverages four different fully connected neural networks to assess different parts of DistilBERT outputs. Table 5 provides an overview of the hyperparameters used within the ensemble classifier.

Each classifier used in the ensemble draws upon different portions of the DistilBERT embeddings. Figure 3a,b illustrate how a tensor of three sequentially hidden layers (marked in blue) is combined with a common ground layer (marked in orange) to form the ensemble classifiers' input. The network-specific hidden layers and the common ground layer are concatenated and afterward transformed into a 1D vector for the fully connected neural network, which is then trained on these data. A majority voting mechanism is used to generate two crucial outputs: (i) an overarching assessment and (ii) a confidence score. The confidence score serves as a metric to gauge the consensus among the various neural networks regarding their evaluations.

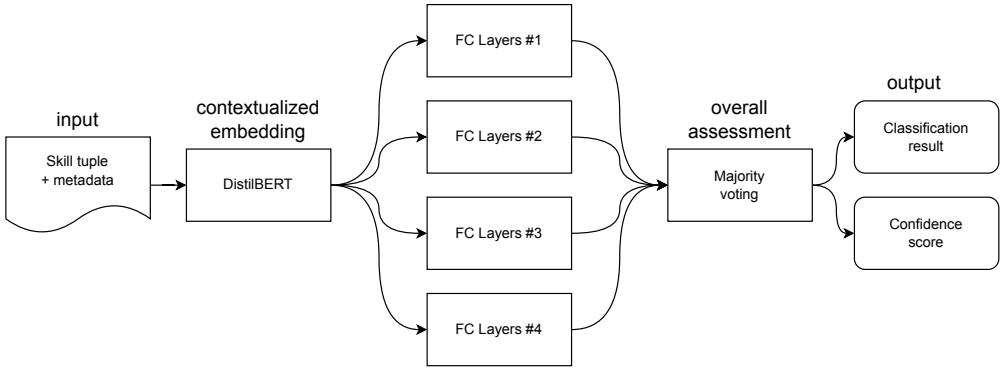
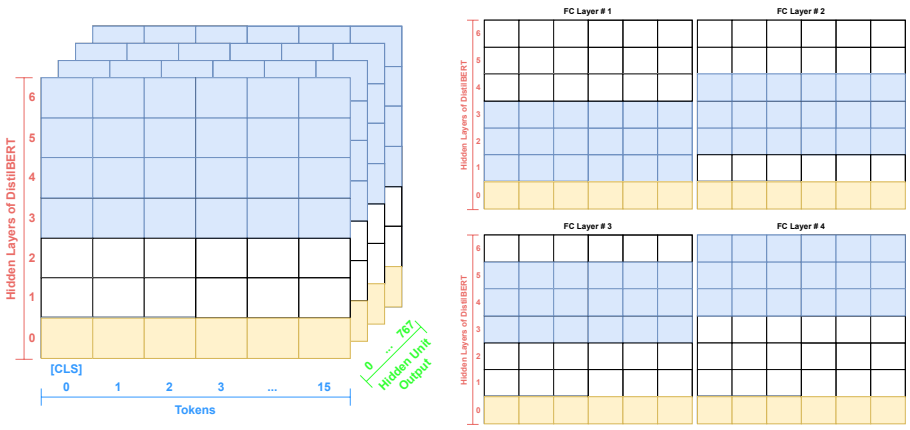



**Figure 2.** Deep learning ensemble classifier. The fully connected layers (FC Layers) independently assess the contextualized embeddings and provide their output to the majority voting component.

**Table 5.** Ensemble model configuration and parameters.

| **Ensemble Parameters** | |
|---|---|
| Pretrained Model | DistilBERT Base Multilingual Cased |
| Optimizer | AdamW |
| Learning rate | $1 \times 10^{-7}$ |
| Task | BertSequenceClassification |
| Input/Output Nodes | 49,152/1 |
| Hidden Layers | 12 |
| Hidden Activation Functions | SELU |
| Loss Function | Binary Cross Entropy |
| Output Activation Function | Sigmoid |




(**a**) (**b**)

**Figure 3.** Composition of the inputs for the ensemble model. (**a**) Tensor from DistilBERT Embeddings. (**b**) Diverse input for each network.

#### 4.4.1. Selection of Network Configurations for the Ensemble

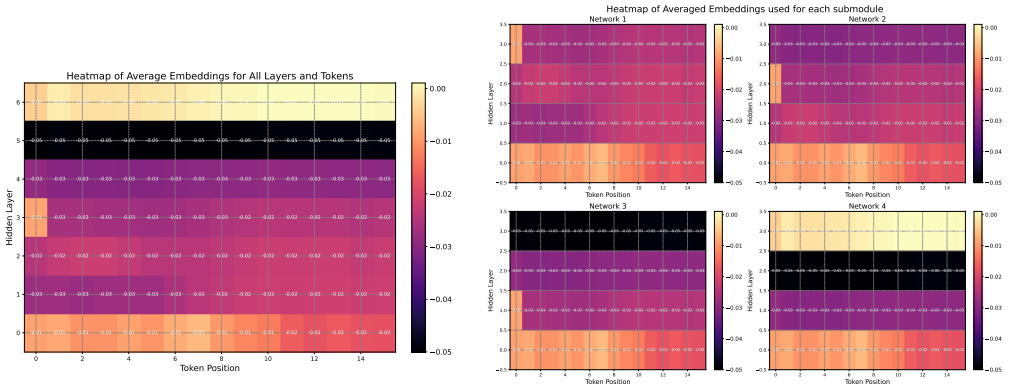
To prevent potential ties in majority voting, we conducted a preliminary study that evaluated the Area Under the Curve (AUC) scores for various ensemble configurations (Table 6).

From this analysis, we identified the top three performing networks for the final classifier. Plotting the averaged embeddings of the gold standard as a heatmap (Figure 4a) illustrates that the values in the last hidden layer lean towards zero. This observation may indicate why the fourth network is the least-performing one (Table 6), as it includes this particular layer. As explained earlier, the input for each network involves slicing DistilBERT embeddings, enabling different perspectives for the ensemble (Figure 4b). The input data

for networks one and two appear more balanced than those for networks three and four. This difference may explain why these networks outperform the others in the ensemble.

**Table 6.** AUC scores for each network/classification task (automatability, offshorability).

| Architecture | AUC (Offshorability) | AUC (Automatability) |
|---|---|---|
| FC Layers #1 | 0.915 | 0.942 |
| FC Layers #2 | 0.914 | 0.926 |
| FC Layers #3 | 0.910 | 0.923 |
| FC Layers #4 | 0.907 | 0.830 |




(**a**) (**b**)

**Figure 4.** Heatmaps of embeddings from DistilBERT. (**a**) Heatmap of the average gold standard embeddings from DistilBERT. (**b**) Heatmap of gold standard embeddings for each network from DistilBERT.

Figure 5 shows the Shapley values [48] for the longest bipartite skill label extracted from the gold standard. The Shapley values suggest that substantives tend to have negative values, whereas verbs and denominatives tend to have positive values, as illustrated in blue. This observation implies that although the classifier operates on bipartite skill labels, subtokens still provide additional context due to their impact on each other's meaning. We therefore consider all DistilBERT tokens in the ensemble networks.

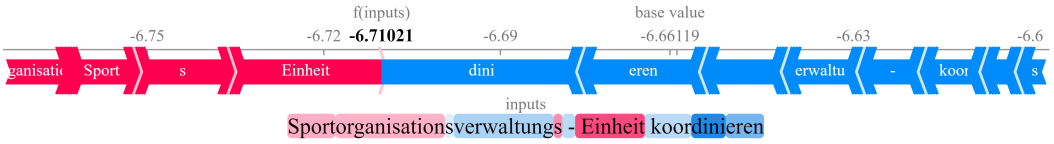



**Figure 5.** Shapley values of the longest bipartite skill label in the gold standard (English translation: coordinate Sports Event Management Unit).

#### 4.4.2. Determining the Optimal Voting Threshold

Subsequently, we used ROC curves to determine each network's optimal voting threshold (see Table 7). Given that our objective is to achieve a balance between true positives (TPs) and false positives (FPs), we tailored our selection to address the importance parity between "is_offshorability" and "is_not_offshorability" and the same for "is_automatable" and "is_not_automatable". This approach ensures a robust classification strategy that aligns with the nuanced requirements of the target domain.

**Table 7.** Assessment of the optimal threshold for the ROC Curves of the networks

| Architecture | Opt. Threshold (Offshorability) | Opt. Threshold (Automatability) |
|---|---|---|
| FC Layers #1 | 0.58 | 0.56 |
| FC Layers #2 | 0.60 | 0.56 |
| FC Layers #3 | 0.58 | 0.55 |

## 5. Evaluation

This section introduces the evaluation datasets (Section 5.1), provides assessments of the classifier performance on the random selection dataset (Section 5.2) and the skewed rule-based dataset (Section 5.3), as well as presents a discussion of the obtained results.

### *5.1. Datasets*

The evaluation builds upon two different gold standard datasets: (i) a dataset that contains a random selection of skills annotated by human experts (Section 5.1.1) and (ii) skills annotated based on expert-drafted heuristics (Section 5.1.2).

#### 5.1.1. Random Selection Gold Standard Dataset

The data were collected by jobchannel AG (Thalwil, Switzerland) (https://www.jobchannel.ch, accessed on 1 March 2024), a company focused on aggregating job data from online sources (e.g., boards, websites, etc.). The jobchannel ontology describes the various skills necessary for the Swiss job market. Each skill is split into a (predicate, topic) pair representing the action and the context in which the skill is performed.

The skill "Marketingkonzept planen" (plan marketing concepts), for example, comprises the topic "marketing concepts" and the predicate "plan".

Human annotators used the following guidelines based on Swiss practices for manually assessing skills with a binary value for offshorability and automatability:

1. Offshorability determines if a task can be performed regardless of the physical presence of the person performing it. Outsourcing issues may arise due to location dependencies, human interaction (e.g., especially if cultural preferences are considered), or even the need to move around large objects.
2. Automatability assesses whether technology (e.g., robots, drones, software, etc.) can currently fully perform the task. Activities are non-automatable if they are not clearly specified or need too much human interaction.

Table 8 summarizes the dataset statistics. The experts labeled 67.7% of the examined skills as offshorable and 58.5% as automatable. Some examples of expert assessments can be found in Table 9.

**Table 8.** Class distribution within the gold standard datasets. Please note that the rule-based dataset frequently only provides labels for only one of the classes (i.e., only offshorable or automatable).

| Corpus | Offshorable | | Automatable | | Total |
|---|---|---|---|---|---|
| | Yes | No | Yes | No | Annotations |
| Random selection | 1526 | 728 | 1319 | 935 | 4508 |
| Rule-based | 5280 | 602 | 2577 | 5478 | 13,937 |

**Table 9.** Example expert annotations for both offshorability and automatability.

| Predicate | Topic | Offshorability | Automatability |
|---|---|---|---|
| create | dossier | 1 | 1 |
| correct | jaw malposition | 0 | 0 |
| program | user interface | 1 | 0 |
| clean | object | 0 | 1 |

#### 5.1.2. Rule-Based Gold Standard Dataset

The creation of the rule-based gold standard dataset has been guided by a qualitative analysis of misclassifications obtained from the skill classifier. An analysis of these classification errors revealed that, in many cases, humans could apply simple rules to assign these skills to the correct class. Therefore, we asked domain experts to identify rules that help classify these skills. These rules have been used to create a separate dataset that aims to identify mistakes that are particularly obvious to humans. Tables 10 and 11 provide examples of these expert-created classification rules.

**Table 10.** Example rules for identifying offshorable and non-offshorable skills. The asterisk in the topic column indicates that the rule works across all skill topics.

| Predicate | Topic | Is Offshorable | Example |
|---|---|---|---|
| clean | * | no | clean house |
| drive | * | no | drive truck |
| program | * | yes | program python |
| write | * | yes | write article |

**Table 11.** Example rules for identifying automatable and non-automatable skills. The asterisk in the topic column indicates that the rule works across all skill topics.

| Predicate | Topic | Automatable | Example |
|---|---|---|---|
| moderate | * | no | moderate meeting |
| negotiate | * | no | negotiate contract |
| calculate | * | yes | calculate production time |
| monitor | * | yes | monitor process |

Based on these rules, we created a corpus that covers 13,937 skills. In contrast to the random selection gold standard, the rule-based gold standard only considers obvious classifications that are covered by these heuristics. It therefore does not contain any challenging cases in which expert assessments differed or required discussions and further clarifications to decide on the skill's correct class.

Table 8 provides dataset statistics for both gold standard datasets. Please note that the manually annotated random selection corpus assigns offshorability and automatability classifications to each skill. The rule-based corpus, in contrast, only covers the class outlined in the rule for that particular skill (i.e., either offshorability, automatability, or both). It is also important to note that class distribution differs significantly between both datasets. Offshorable skills are even more frequent in the rule-based corpus, while the distribution for automatability is reversed between the datasets.

### *5.2. Performance on the Random Selection Gold Standard*

This evaluation assesses the classifiers' performance on the random selection gold standard introduced in Section 5.1.1 using a k-fold cross-validation with four folds. The test data were divided into 80% training data and 20% evaluation data in each run. Once all runs were executed, the results were summarized.

This procedure was not applied to the GPT/heuristic classifier because it only uses few-shot training with examples that were not part of the evaluation dataset. As it can easily be seen in Tables 12 and 13, the DistilBERT classifier is the best performer. The Transformer model's effective use of self-attention and pre-training mechanisms was key to extracting meaningful semantics for classification, even from input data that are limited to bipartite skill labels.

SVM results were as expected as they did not perform well in the presence of limited training data. The low performance of the GPT-based heuristic is not entirely surprising as it lacked domain knowledge and was not optimized for classifying human resources tasks.

Although the ensemble model is considerably more complex than DistilBERT, it still performs similarly to this Transformer model. This result indicates that the additional generalization capabilities provided by this more sophisticated model did not translate into better performance since the model has been constrained by the scarcity of input data, which was limited to bipartite skill descriptions.

**Table 12.** The "offshorable" indicator's classification performance on the randomly selected dataset. The best results are indicated in bold.

| Model | f1 | Precision | Recall | Accuracy |
|---|---|---|---|---|
| SVM baseline | 0.60 | 0.64 | 0.68 | 0.68 |
| distilbert-base-multilingual-cased | **0.76** | **0.77** | **0.77** | **0.77** |
| ChatGPT combined with Heuristic | 0.69 | 0.69 | 0.68 | 0.68 |
| Ensemble model | 0.74 | **0.77** | 0.73 | 0.73 |

**Table 13.** The"automatable" indicator's classification performance on the randomly selected dataset. The best results are indicated in bold.

| Model | f1 | Precision | Recall | Accuracy |
|---|---|---|---|---|
| SVM baseline | 0.58 | 0.68 | 0.64 | 0.64 |
| distilbert-base-multilingual-cased | **0.75** | **0.75** | **0.74** | **0.74** |
| ChatGPT combined with Heuristic | 0.55 | 0.56 | 0.55 | 0.55 |
| Ensemble model | 0.73 | 0.74 | 0.73 | 0.73 |

### *5.3. Performance on the Rule-Based Dataset*

We draw upon the skewed rule-based dataset for assessing model robustness. This dataset only contains annotations that are (i) obvious to human experts and (ii) can be derived from well-established heuristics (Section 5.1.2).

These experiments, therefore, aim at assessing the classifiers' robustness against grave mistakes, i.e., cases where automatic classifications clearly violate human intuition. They have been designed to indicate the impact of mitigation strategies such as increasing model complexity (i.e., large language models) and enhancing model diversity (ensemble method) on the model's robustness against these types of mistakes.

Table 14 summarizes the performance of all four methods on the offshorable indicator. Both the Transformer and the ensemble model provide very good results for offshorability, while the ChatGPT heuristic scored even below the SVM baseline.

**Table 14.** Performance for the "offshorable" indicator on the skewed rule-based dataset. The best results are indicated in bold.

| Model | f1 | Precision | Recall | Accuracy |
|---|---|---|---|---|
| SVM baseline | 0.85 | 0.91 | 0.90 | 0.90 |
| distilbert-base-multilingual-cased | 0.93 | 0.94 | 0.93 | 0.93 |
| ChatGPT combined with Heuristic | 0.76 | 0.79 | 0.80 | 0.80 |
| Ensemble model | **0.94** | **0.95** | **0.94** | **0.94** |

Table 15 compares the systems' performance on the automatable indicator. The ensemble model provides the best results for the offshorable indicator and the DistilBERT classifier excels for the automatable category. Both models are very similar in terms of performance and clearly outperform the SVM baseline and the GPT heuristic.

These results provide the following insights:

- All models that have been fine-tuned on the random selection dataset have benefited tremendously from training on task-specific data. Consequently, even the fine-tuned SVM classifier outperforms the GPT heuristics.

- Both DistilBERT, the ensemble, and GPT heuristics leverage background knowledge from the base models, which improves model performance compared to the SVM baseline but is insufficient to compensate for fine-tuning on domain data (compare GPT's performance). GPT heuristics, therefore, struggle with providing accurate classifications based on the limited context available through prompt engineering.
- Additional ensemble complexity does not translate into better performance.

**Table 15.** Performance for the "automatable" indicator on the skewed rule-based dataset. The best results are indicated in bold.

| Model | f1 | Precision | Recall | Accuracy |
|---|---|---|---|---|
| SVM baseline | 0.52 | 0.81 | 0.54 | 0.54 |
| distilbert-base-multilingual-cased | **0.91** | **0.91** | **0.90** | **0.90** |
| ChatGPT combined with Heuristic | 0.46 | 0.58 | 0.46 | 0.46 |
| Ensemble model | 0.90 | 0.90 | **0.90** | **0.90** |

## 6. Discussion

The evaluation demonstrates that the fine-tuned models provide good results, even for the skewed rule-based dataset, which differs significantly from the random selection dataset in terms of class distribution (Table 8). This is even true for the automatable classification task, which is inherently more intricate and complex than assessing the offshorable indicator. Unlike the offshorable label, which is primarily impacted by whether a task needs to be performed on-site (e.g., particularly relevant for Gig economy platforms like Uber), the automatable label encompasses a multitude of underlying factors. These factors are considerably more nuanced and context-dependent and especially useful for high-skill, location-independent services such as Fiverr and Upwork.

The relevance of multidimensional aspects makes automatability more difficult to discern, which is also reflected in classification performance. The less complex models (e.g., SVM baseline and GPT) struggle with the automatable indicator. The difference in performance is less pronounced in the random selection gold standard dataset where both classifiers yield similar results. In contrast, SVM and GPT-based heuristics provide significantly worse classification performance for the automatable indicator on the skewed rule-based dataset. The SVM model clearly lacks the complexity required to successfully generalize its assessments, while the GPT classifier fails to correctly classify the label of unseen bipartite skills based on the provided few-show examples.

DistilBERT and the ensemble classifier, in contrast, provide f1 metrics and accuracy measures of 90% and above for both the automatable and offshorable labels, indicating that these models are sufficiently complex to approximate the classification model based on the available training data. The scarcity of the input data available in the job channel occupation ontology, which is limited to bipartite skill labels (i.e., a topic and verb describing the skill), poses a considerable challenge to the classification process. The evaluations in the previous section suggest that techniques that increase model complexity (e.g., ensembling) do not necessarily improve skill classification performance, indicating that the ensemble's loss function might be non-convex [49].

## 7. Conclusions and Outlook

This paper discussed machine learning models capable of classifying bipartite skill labels in terms of their offshorability and automatability. Both indicators help assess a skill's future readiness, i.e., how likely there will be a future demand for it. Bipartite skill labels describe a skill based on two terms—the *skill topic* (e.g., Python) and a *verb* that provides additional context regarding the task or activity (e.g., programming). This bipartite skill specification allows distinguishing the skill of *Programming Python* from skills such as *Teaching Python*, *Debugging Python*, or *Profiling Python* but do not provide any context beyond this information.

Our primary contributions lie in the customization and fine-tuning of deep learning methods tailored for the automatic classification of future readiness (in terms of automatability and offshorability) within the framework of bipartite skill labels. We conducted comprehensive evaluations of these methods to assess their performance and to investigate whether increasing model complexity translates into classification performance under the constraints imposed by the use of bipartite skill labels.

The paper compares the performance of four skill classifiers (SVM, Transformers, GPT-based, and DistilBERT ensemble) which were trained using a gold standard dataset of 2254 annotated bipartite skill specifications obtained from the scientific literature and domain experts. A fourfold cross-evaluation assessed classification performance on the gold standard dataset and was followed by a second set of experiments that evaluated the robustness of the trained classifiers by applying them to a considerably larger dataset of 13,937 annotations created based on expert-defined annotation rules.

The evaluation results indicate that domain-specific fine-tuning is essential to improve the accuracy of the classification algorithm. Once fine-tuned, even the SVM baseline outperformed the GPT classifier, which only benefited from few-shot learning through examples integrated via prompt engineering. The Transformer classifiers also leveraged the knowledge obtained during pre-training, easily outperforming models that did not benefit from pre-training. Given the scarcity of input data (the two terms obtained from the bipartite skill labels), increasing model complexity does not necessarily lead to improvements in classification performance. This result is noteworthy since the ensemble models have not outperformed the single DistilBERT model.

In conclusion, assessing a skill's automatability and offshorability offers insights into the future readiness of professional skills, therefore aiding in forecasting the likelihood of future demand for specific skill sets. Through customization and fine-tuning of deep learning methods, the research evaluates the performance of skill classifiers. It provides valuable findings for improving the accuracy of classification algorithms that are limited to bipartite skill labels in predicting job market demands.

Future work will integrate insights into a skill's future readiness with systems such as CareerCoach [41], which provides reskilling and upskilling suggestions. We also plan to leverage additional background knowledge in the classification process. O*NET and ESCO ontologies, for instance, encode knowledge on skills, competencies, and occupations that might be helpful in better contextualizing the classification input. Contextualization is also key to addressing the problem of scarce input data and, therefore, paving the way towards successfully using more complex classifier setups. For this purpose, the authors also plan to incorporate domain knowledge, e.g., a sustainability knowledge graph initially developed in earlier work for UN Environment [50] and further extended in the SDG-HUB project (https://www.weblyzard.com/sdg-hub, accessed on 1 March 2024) and *Climateurope2* (https://www.climateurope2.eu, accessed on 1 March 2024), a Support Action funded by the European Union with an interest in job creation in the climate services sector and in the impact of the green transition on the European labor market [51].

**Author Contributions:** Conceptualization, A.W.; methodology, A.W., N.S., R.W., A.G. and A.M.P.B.; software, N.S., R.W. and A.G.; resources, N.S. and R.W.; data curation, N.S., R.W. and A.G.; writing—original draft preparation, A.W., N.S., R.W., A.G. and A.M.P.B.; writing—revisions and extensions, A.W., N.S., R.W., A.G., A.M.P.B. and A.S.; visualization, R.W., A.G., A.M.P.B. and A.S.; supervision, A.W.; funding acquisition, A.S. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research has been conducted within the Future of Work project (https://semanticlab.net/future-of-work, accessed on 1 March 2024) funded by Innosuisse. Adrian M. P. Braşoveanu was partially funded through the SDG-HUB project funded by the Austrian Federal Ministry for Climate Action, Environment, Energy, Mobility and Technology (BMK) as part of the SDG-HUB Project (GA No. 892212). Arno Scharl was partially funded by the European Union through the Horizon Europe project Climateurope2 (G.A. 101056933).

**Data Availability Statement:** The datasets used for the analysis presented in this article are not available publicly as their copyright lies with *jobchannel ag*. Requests to access the datasets should be directed to info@jobchannel.ch.

**Acknowledgments:** We would like to express our gratitude to Philipp Kuntschik, who acquired the *Future of Work* project, and to Cornel Müller, Flavio Battaini and Felix Busch from *jobchannel ag* for creating the bipartite skill dataset.

**Conflicts of Interest:** The authors declare no conflicts of interest. The funder had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

## References

1. Bick, A.; Blandin, A.; Mertens, K. Work from Home After the COVID-19 Outbreak. *Am. Econ. J. Macroecon.* **2020**, *15*, 1–39. [CrossRef]
2. Suesstrunk, N.; Weichselbraun, A.; Waldwogel, R. Large Language Models versus Foundation Models for Assessing the Future-Readiness of Skills. In *Nachhaltige Information—Information für Nachhaltigkeit. Tagungsband des 17. Internationalen Symposiums für Informationswissenschaft (ISI 2023), Chur, Schweiz, 7–9 November 2023*; Semar, W., Ed.; Verlag Werner Huelsbusch: Gluckstadt, Germany, 2023; pp. 294–311. [CrossRef]
3. Shuai, X.; Chmura, C.; Stinchcomb, J. COVID-19, labor demand, and government responses: Evidence from job posting data. *Bus. Econ.* **2021**, *56*, 29–42. [CrossRef] [PubMed]
4. Zarifhonarvar, A. Economics of chatgpt: A labor market view on the occupational impact of artificial intelligence. *J. Electron. Bus. Digit. Econ.* **2023**. [CrossRef]
5. Poláková, M.; Suleimanová, J.H.; Madzík, P.; Copuš, L.; Molnárová, I.; Polednová, J. Soft skills and their importance in the labour market under the conditions of Industry 5.0. *Heliyon* **2023**, *9*, e18670. [CrossRef] [PubMed]
6. Bishnoi, M.M.; Ramakrishnan, S.; Suraj, S.; Dwivedi, A. Impact of AI and COVID-19 on manufacturing systems: An Asia Pacific Perspective on the two Competing exigencies. *Prod. Manuf. Res.* **2023**, *11*, 2236684. [CrossRef]
7. Mezhoudi, N.; Alghamdi, R.; Aljunaid, R.; Krichna, G.; Düstegör, D. Employability prediction: A survey of current approaches, research challenges and applications. *J. Ambient Intell. Humaniz. Comput.* **2023**, *14*, 1489–1505. [CrossRef] [PubMed]
8. Khaouja, I.; Kassou, I.; Ghogho, M. A Survey on Skill Identification From Online Job Ads. *IEEE Access* **2021**, *9*, 118134–118153. [CrossRef]
9. Hassan, M.U.; Alaliyat, S.; Sarwar, R.; Nawaz, R.; Hameed, I.A. Leveraging deep learning and big data to enhance computing curriculum for industry-relevant skills: A Norwegian case study. *Heliyon* **2023**, *9*, e15407. [CrossRef]
10. Lukauskas, M.; Šarkauskaitė, V.; Pilinkienė, V.; Stundžienė, A.; Grybauskas, A.; Bruneckienė, J. Enhancing skills demand understanding through job ad segmentation using NLP and clustering techniques. *Appl. Sci.* **2023**, *13*, 6119. [CrossRef]
11. Autor, D.H.; Dorn, D. The Growth of Low-Skill Service Jobs and the Polarization of the US Labor Market. *Am. Econ. Rev.* **2013**, *103*, 1553–1597. [CrossRef]
12. Josten, C.; Lordan, G. Automation and the changing nature of work. *PLoS ONE* **2022**, *17*, e0266326. [CrossRef] [PubMed]
13. Josten, C.; Lordan, G. Robots at Work: Automatable and Non-automatable Jobs. In *Handbook of Labor, Human Resources and Population Economics*; Zimmermann, K.F., Ed.; Springer International Publishing: Cham, Switzerland, 2020; pp. 1–24. [CrossRef]
14. Eloundou, T.; Manning, S.; Mishkin, P.; Rock, D. GPTs are GPTs: An Early Look at the Labor Market Impact Potential of Large Language Models. *arXiv* **2023**, arXiv:2303.10130. [CrossRef]
15. Scao, T.L.; Fan, A.; Akiki, C.; Pavlick, E.; Ilić, S.; Hesslow, D.; Castagné, R.; Luccioni, A.S.; Yvon, F.; Gallé, M.; et al. BLOOM: A 176B-Parameter Open-Access Multilingual Language Model. *arXiv* **2023**, arXiv:2211.05100. [CrossRef]
16. Brynjolfsson, E.; Mitchell, T.; Rock, D. What Can Machines Learn, and What Does It Mean for Occupations and the Economy? *AEA Pap. Proc.* **2018**, *108*, 43–47. [CrossRef]
17. Bai, G.; Liu, J.; Bu, X.; He, Y.; Liu, J.; Zhou, Z.; Lin, Z.; Su, W.; Ge, T.; Zheng, B.; et al. MT-Bench-101: A Fine-Grained Benchmark for Evaluating Large Language Models in Multi-Turn Dialogues. *arXiv* **2024**, arXiv:2402.14762. [CrossRef]
18. Zhang, Q.; Zhang, T.; Zhai, J.; Fang, C.; Yu, B.; Sun, W.; Chen, Z. A Critical Review of Large Language Model on Software Engineering: An Example from ChatGPT and Automated Program Repair. *arXiv* **2023**, arXiv:2310.08879. [CrossRef]
19. Dixon, J.; Hong, B.; Wu, L. The Robot Revolution: Managerial and Employment Consequences for Firms. *Manag. Sci.* **2021**, *67*, 5586–5605. [CrossRef]
20. Lipsey, R.G.; Carlaw, K.; Bekar, C. *Economic Transformations: General Purpose Technologies and Long-Term Economic Growth*; OCLC: ocm60931387; Oxford University Press: Oxford, UK; New York, NY, USA, 2005.
21. Wagner, G.; Prester, J.; Paré, G. Exploring the boundaries and processes of digital platforms for knowledge work: A review of information systems research. *J. Strateg. Inf. Syst.* **2021**, *30*, 101694. [CrossRef]
22. Nevo, D.; Kotlarsky, J. Crowdsourcing as a strategic IS sourcing phenomenon: Critical review and insights for future research. *J. Strateg. Inf. Syst.* **2020**, *29*, 101593. [CrossRef]
23. Shafiei Gol, E.; Stein, M.K.; Avital, M. Crowdwork platform governance toward organizational value creation. *J. Strateg. Inf. Syst.* **2019**, *28*, 175–195. [CrossRef]

24. Khan, N.; Sikes, J. *IT under Pressure*; Technical Report; McKinsey: Chicago, IL, USA, 2014.
25. Dunn, M.; Munoz, I.; Jarrahi, M.H. Dynamics of flexible work and digital platforms: Task and spatial flexibility in the platform economy. *Digit. Bus.* **2023**, *3*, 100052. [CrossRef]
26. Pinheiro, A.; Sochirca, E.; Afonso, O.; Neves, P.C. Automation and off (re) shoring: A meta-regression analysis. *Int. J. Prod. Econ.* **2023**, *264*, 108980. [CrossRef]
27. Lin, T.; Wang, Y.; Liu, X.; Qiu, X. A survey of transformers. *AI Open* **2022**, *3*, 111–132. [CrossRef]
28. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, MN, USA, 2–7 June 2019; pp. 4171–4186. [CrossRef]
29. Sanh, V.; Debut, L.; Chaumond, J.; Wolf, T. DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. In Proceedings of the 5th Workshop on Energy Efficient Machine Learning and Cognitive Computing, Vancouver, BC, Canada, 13 December 2019. [CrossRef]
30. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv* **2019**, arXiv:1907.11692. [CrossRef]
31. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is All you Need. In *Proceedings of the Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, Long Beach, CA, USA, 4–9 December 2017*; Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H.M., Fergus, R., Vishwanathan, S.V.N., Garnett, R., Eds.; NeurIPS: San Diego, CA, USA, 2017; pp. 5998–6008. [CrossRef]
32. Bommasani, R.; Hudson, D.A.; Adeli, E.; Altman, R.; Arora, S.; von Arx, S.; Bernstein, M.S.; Bohg, J.; Bosselut, A.; Brunskill, E.; et al. On the Opportunities and Risks of Foundation Models. *arXiv* **2022**, arXiv:2108.07258. [CrossRef]
33. Minaee, S.; Kalchbrenner, N.; Cambria, E.; Nikzad, N.; Chenaghlu, M.; Gao, J. Deep Learning-based Text Classification: A Comprehensive Review. *ACM Comput. Surv.* **2022**, *54*, 62:1–62:40. [CrossRef]
34. da Costa, L.S.; Oliveira, I.L.; Fileto, R. Text classification using embeddings: A survey. *Knowl. Inf. Syst.* **2023**, *65*, 2761–2803. [CrossRef]
35. Pham, P.; Nguyen, L.T.; Pedrycz, W.; Vo, B. Deep learning, graph-based text representation and classification: A survey, perspectives and challenges. *Artif. Intell. Rev.* **2023**, *56*, 4893–4927. [CrossRef]
36. Siddiqui, T.; Amer, A.Y.A. A Comprehensive Review on Text Classification and Text Mining Techniques Using Spam Dataset Detection. *Math. Comput. Sci.* **2023**, *2*, 1–17. [CrossRef]
37. Zhao, W.X.; Zhou, K.; Li, J.; Tang, T.; Wang, X.; Hou, Y.; Min, Y.; Zhang, B.; Zhang, J.; Dong, Z.; et al. A Survey of Large Language Models. *arXiv* **2023**, arXiv:2303.18223. [CrossRef]
38. Brown, T.B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language Models are Few-Shot Learners. In *Proceedings of the Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, Virtual, 6–12 December 2020*; Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., Lin, H., Eds.; NeurIPS: San Diego, CA, USA, 2020.
39. Liesenfeld, A.; Lopez, A.; Dingemanse, M. Opening up ChatGPT: Tracking openness, transparency, and accountability in instruction-tuned text generators. In Proceedings of the 5th International Conference On Conversational User Interfaces, Eindhoven, The Netherlands, 19–21 July 2023; pp. 1–6. [CrossRef]
40. Al-Hashimi, M.; Hamdan, A.; Razzaque, A.; Al-Sartawi, A.; Reyad, S. Skill gaps in management information systems alumni. In Proceedings of the International Conference on Artificial Intelligence and Computer Vision (AICV2020), Cairo, Egypt, 8–10 April 2020; Springer: Cham, Switzerland, 2020; pp. 773–782. [CrossRef]
41. Weichselbraun, A.; Waldvogel, R.; Fraefel, A.; van Schie, A.; Kuntschik, P. Building Knowledge Graphs and Recommender Systems for Suggesting Reskilling and Upskilling Options from the Web. *Information* **2022**, *13*, 510. [CrossRef]
42. Xu, T.; Chen, W.; Wang, P.; Wang, F.; Li, H.; Jin, R. CDTrans: Cross-domain Transformer for Unsupervised Domain Adaptation. In Proceedings of the Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, 25–29 April 2022. [CrossRef]
43. Weichselbraun, A. Inscriptis—A Python-based HTML to text conversion library optimized for knowledge extraction from the Web. *J. Open Source Softw.* **2021**, *6*, 3557. [CrossRef]
44. Lee, J.; Tang, R.; Lin, J. What Would Elsa Do? Freezing Layers During Transformer Fine-Tuning. *arXiv* **2019**, arXiv:1911.03090. [CrossRef]
45. Akiba, T.; Sano, S.; Yanase, T.; Ohta, T.; Koyama, M. Optuna: A Next-generation Hyperparameter Optimization Framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD 2019), Anchorage, AK, USA, 4–8 August 2019*; Teredesai, A., Kumar, V., Li, Y., Rosales, R., Terzi, E., Karypis, G., Eds.; ACM: New York, NY, USA, 2019; pp. 2623–2631. [CrossRef]
46. Ma, H.; Zhang, C.; Bian, Y.; Liu, L.; Zhang, Z.; Zhao, P.; Zhang, S.; Fu, H.; Hu, Q.; Wu, B. Fairness-guided Few-shot Prompting for Large Language Models. In *Proceedings of the Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023 (NeurIPS 2023), New Orleans, LA, USA, 10–16 December2023*; Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., Levine, S., Eds.; NeurIPS: San Diego, CA, USA, 2023. [CrossRef]
47. Johnson, J.A.; Bullock, D.H. Fragility in AIs Using Artificial Neural Networks. *Commun. ACM* **2023**, *66*, 28–31. [CrossRef]

48. Lundberg, S.M.; Lee, S. A Unified Approach to Interpreting Model Predictions. In *Proceedings of the Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, Long Beach, CA, USA, 4–9 December 2017*; Guyon, I., von Luxburg, U., Bengio, S., Wallach, H.M., Fergus, R., Vishwanathan, S.V.N., Garnett, R., Eds.; NeurIPS: San Diego CA, USA, 2017; pp. 4765–4774.
49. Mattei, P.; Garreau, D. Are ensembles getting better all the time? *arXiv* **2023**, arXiv:2311.17885. [CrossRef]
50. Scharl, A.; Herring, D.D.; Rafelsberger, W.; Hubmann-Haidvogel, A.; Kamolov, R.; Fischl, D.; Föls, M.; Weichselbraun, A. Semantic Systems and Visual Tools to Support Environmental Communication. *IEEE Syst. J.* **2017**, *11*, 762–771. [CrossRef]
51. Vandeplas, A.; Vanyolos, I.; Vigani, M.; Vogel, L. *The Possible Implications of the Green Transition for the EU Labour Market*; Technical Report; Directorate General Economic and Financial Affairs (DG ECFIN), European Economy Discussion Papers; European Union: Luxembourg, 2022.