Building a Custom Taxonomy of AI Skills and Tasks from the Ground Up with Job Postings

Stephen Meisenbacher
Technical University of Munich
Garching, Germany
stephen.meisenbacher@tum.de

Peter Norlander
Loyola University Chicago
Chicago, IL, USA
pnorlander@luc.edu

Abstract

Utilizing LLMs for automated taxonomy construction presents a clear opportunity for the comprehensive, yet efficient mapping of potentially complex domains. When contending with high volumes of rapidly growing corpora, however, it becomes unclear how to best leverage such data for optimal taxonomy construction. Taking the case of systematizing *AI skills in the workplace*, we use two large-scale job postings corpora to investigate key design decisions for the inclusion (or exclusion) of data points for taxonomy construction. We propose TAXONOMYBUILDER as a blueprint for our systematic study, with which we evaluate various configurations of custom, data-informed, and hierarchical taxonomies. We demonstrate that *less* data can provide more clarity: filtering inputs to TAXONOMYBUILDER provides better domain-specific coverage than offering unfiltered inputs to clustering and LLM-enhanced hierarchical taxonomy labeling tools.

1 Introduction

What AI Skills are in demand in recent years? To answer this and other timely questions about rapid change in labor markets, novel, open taxonomies of tasks and skills are needed to enable tracking of granular changes in work (Frank et al., 2019; Margaryan, 2023). When built from high-volume, real-time sources of labor market text such as job postings, such taxonomies are valuable sources of data to inform individual career development, educational program designs, workforce development interventions, and firm strategy (Zweig, 2026).

However, proprietary taxonomies, data access restrictions, and lack of standardization limit the development and use of automated tools to harness general information from job postings at scale (National Academies of Sciences, 2025). Open data, including novel taxonomies and data generated by them, are potential public goods to support broad skills development, training needs for workforce development, growth, democratization, and entrepreneurship (Lerner and Tirole, 2002; Nagaraj et al., 2020; Nagaraj, 2022). From a research standpoint, to assess whether AI is complementing or replacing human skills (Acemoglu et al., 2026), social scientists need detailed and up-to-date taxonomic information on the human tasks and skills of workers building and using AI.

When faced with such problems, researchers have often turned to *taxonomy construction*. The creation of taxonomies has spanned the decades, and established methodologies are both accessible and widely used (Michalski and Stepp, 1983; Nickerson et al., 2013). The shortcomings of taxonomy creation mirror those of many traditionally manual tasks, such as the expense of large-scale manual labor, researcher fatigue and bias, and the difficulty in empirically validating created artifacts with real-world data (Vu et al., 2025). The creation of representative taxonomies in custom, complex, and domain-specific fields remains challenging.

In the field of Natural Language Processing and especially in the era of LLMs, researchers have begun to explore the capabilities of automating the process of taxonomy construction. The promise of such approaches is the ability to sift through and categorize massive amounts of data, a step that would be highly expensive in the traditional manner of hand-crafted taxonomies (Vu et al., 2025). Improving initial approaches leveraging clustering algorithms (Gordon, 1996; Cimiano et al., 2004; Ienco and Meo, 2008), recent state-of-the-art methods have leveraged LLMs and their generative capabilities (Moskvoretskii et al., 2026).

As a motivation for our work, when confronted with the need to develop a taxonomy of AI skills from job postings, the literature offers prompt engineering guidance (Chen et al., 2023; Vu et al., 2025) and handles situations where *seed* information, or *seed taxonomies* can guide the LLM-assisted tax-

onomy building process (Zhang et al., 2025; Motamedi et al., 2026). However, in the situation at present, questions posed to us include: *which data* should be used (among all data from a massive text corpus), *how much* of this data should be used, and *in what way* can the built taxonomies be evaluated in turn on real-world datasets? Thus, while it has been established that leveraging LLMs can be helpful for taxonomy building, there is a need for systematic experimentation with key design parameters for best practices in custom domains.

To guide our experiments, we leverage two large corpora of job postings in the United States, spanning the past decade. To address the needs identified above, we build taxonomies that efficiently and comprehensively organize AI in the workplace, particularly skills, tasks, experience, and qualifications. To accomplish this, we design a simple, yet powerful pipeline that transforms mined candidate contexts into hierarchical taxonomies. We evaluate 12 configurations of our pipeline on automatic and LLM-based metrics, leading to clear recommendations for automatic taxonomy building.

Our findings reveal that when handling massive data corpora for automated taxonomy construction, *less is more*. Although the tendency to include as much data as possible, including via data augmentation, might plausibly increase domain coverage, our results demonstrate the opposite. As such, we learn that careful augmentation, strict clustering, and selective filtering can contribute to comprehensive, yet concise taxonomies for large, data-rich domains such as AI skills in the workplace. Our work makes the following additional contributions:

1. We deepen the capabilities of LLM-based taxonomy building by asking and answering key questions surrounding the setup and execution of automatic taxonomy creation pipelines.
2. We showcase the ability of our proposed pipeline, TAXONOMYBUILDER, to efficiently and comprehensively map complex domains such as *AI Skills in the Workplace*, replicated on two large-scale job posting corpora.
3. We make the resulting AI skills taxonomies publicly available and open for further use.
4. We open-source TAXONOMYBUILDER as a generalizable tool to any domain of interest, available at `https://github.com/sjmeis/TaxonomyBuilder` and as a Python package (TAXONOMYBUILDER).

2 Related Work

Automated taxonomy construction. Automated taxonomy construction, often discussed alongside *ontology construction*, *ontology engineering*, or *taxonomy induction*, has been approached in diverse ways even before the era of LLMs. Early methods focus on lexico-syntactic patterns to define relationships in a hierarchy (Hearst, 1992), which are efficient yet suffer from low recall. Subsequent methods relied on the distributional hypothesis (Padó and Lapata, 2007), using the backbone of clustering algorithms and early text representations (e.g., bag of words). Hierarchical approaches (Gordon, 1996; Fountain and Lapata, 2012), allow for the vertical scaling of taxonomies, extending beyond similarity-based clustering of related terms. Other approaches supplement these methods with rich existing knowledge bases such as WordNet (Pasca, 2004; Kozareva and Hovy, 2010).

Automated taxonomy construction has been aided by probabilistic and model-based approaches (Snow et al., 2006; Poon and Domingos, 2010) or graph-based methods (Velardi et al., 2013). Building on the success of embeddings as text representations, ensuing works leveraged these embeddings to form hybrid taxonomy construction methods (Fu et al., 2014; Espinosa-Anke et al., 2016; Zhang et al., 2018). Other works utilized modern reinforcement learning (Mao et al., 2018) or transfer learning (Navarro-Almanza et al., 2020).

LLM-assisted taxonomy construction. The emergence of LLMs has sparked research interest in improving automated taxonomy construction (Vu et al., 2025). Many approaches have been proposed in recent years, including methods to augment existing taxonomies (Kargupta et al., 2025) as well as those that specialize in specific domains, such as scholarly research (Zhu et al., 2025; Lahiri et al., 2025). Typical LLM-based workflows include an extraction and clustering phase (Huang and He, 2025), followed by generative labeling via prompting or fine-tuning (Chen et al., 2023), and finally taxonomy construction (Li et al., 2025; Motamedi et al., 2026). While this workflow is bottom-up (starting from leaf nodes), others follow a top-down approach (Zeng et al., 2024; Marchenko and Dvoichenkov, 2024; Wan et al., 2024).

Our work adopts the bottom-up approach, as we investigate the case of modeling AI skills in job postings, for which no defined taxonomies exist. We build upon the work of approaches

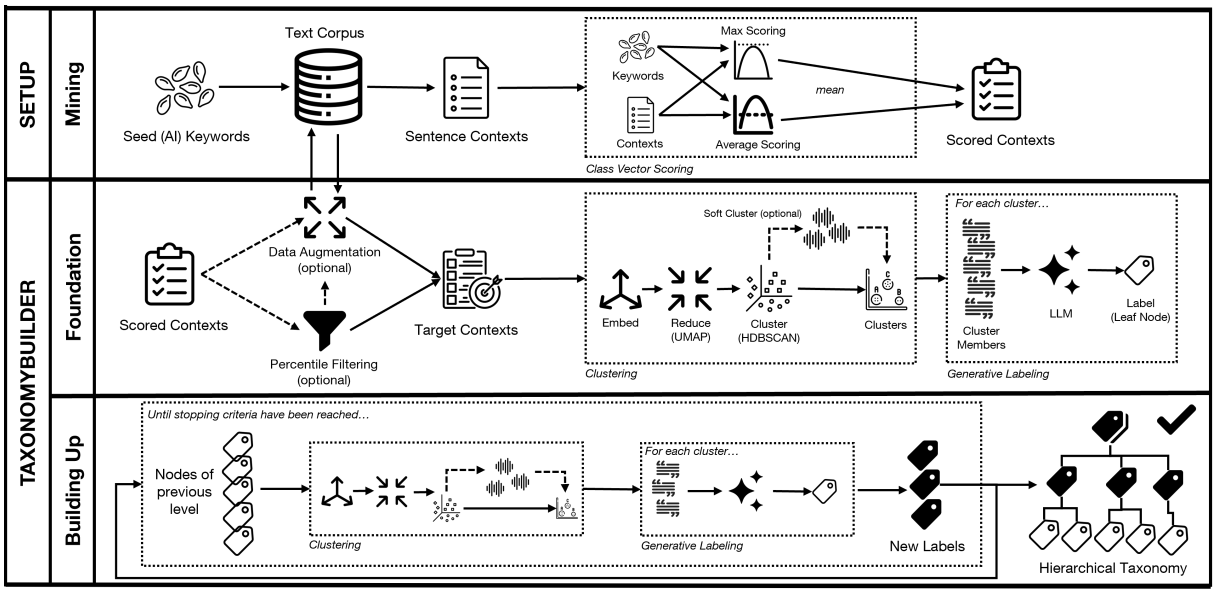

Figure 1: The TAXONOMYBUILDER method. In the top lane, we detail the setup method we follow as a precursor to taxonomy construction, which consists of keyword-based context mining and class-based scoring. The TAXONOMYBUILDER method, in turn, consists of two primary stages (depicted in the center and bottom lanes): (1) the construction of the foundation (leaf) level, followed by iterative vertical construction of further levels of the taxonomy hierarchy. We refer the readers to the main body for specific details on all stages of the process.

such as LLMTAXO (Zhang et al., 2025), who lay the groundwork for cluster labeling methods with LLMs (as well as LLM-based evaluation), and TAXOADAPT (Kargupta et al., 2025), which introduce important inter-mechanism redundancy checks. In terms of domain, the closest work to date is that of Li et al. (2025), who also operate in the domain of job posting data. However, we note that Li et al. (2025) explore LLM-assisted taxonomy construction on very limited data samples (with a maximum of just over 10k job postings), whereas we study a case with millions to tens of millions of unique data points in question. In this, we introduce and systematically evaluate important questions surrounding optimal data usage, which has been previously unexplored, and evaluation, which becomes highly relevant in big data scenarios to ensure proper domain coverage.

3 Datasets

We use two large corpora of job postings from the United States. The first is a collection of over 30 million job postings from the National Labor Exchange (NLx) between 2024 and 2025. To maintain reproducibility, we also evaluate our proposed method on an open corpus of 1.5 million federal job postings collected by Resh et al. (2025). Both datasets are described briefly below.

NLx. The National Labor Exchange (NLx) "provides workforce development professionals, academic researchers, employers, and other organizations that rely on labor market information (LMI) with high-quality, transparent, real-time and historical data that represents the diversity of jobs available in the labor market"[1] As of March 2026, the exchange hosts nearly 3.1 million open positions from 300,000 employers.

The entirety of the NLx corpus currently comprises over 173 million unique job postings from 2007 onwards. To maintain a manageable subset and to focus in on years which feature modern AI skill requirements, we narrow down our dataset to data from 2024 and 2025. Together, this results in a dataset of 32.12 million job postings within the two-year timespan. The average word count of the job postings is 875 (using NLTK on a sample of all December 2025 postings). We make no modifications to the original NLx job posting texts.

Use of NLx data precludes sharing any data in its original form. As such, we replicate our experiments on a second, publicly available dataset.

USAJOBS. We also utilize a corpus of 1.54 million job postings from the USAJOBS platform, the US government's official employment website, made available by Resh et al. (2025). This corpus

[1] https://nlxresearchhub.org/about-us/

contains job postings for positions in the federal government from 2017-2023. The USAJOBS postings are generally long, with an average word count of 5054 (as per NLTK on November 2023 postings).

Test set. It should be noted that we reserve the last month's data in each corpus for evaluation purposes, i.e., December 2025 of NLx and November 2023 of USAJOBS. Thus, all procedures described in the next section, in which data is mined or augmented, leave out this reserved test set.

Data release. We make the USAJOBS dataset publicly available[2]. Regarding outputs and analysis of NLx results, we cannot share the original data, a note we expand upon in our Ethics Statement.

4 The TAXONOMYBUILDER Method

In the following, we detail the design and implementation of TAXONOMYBUILDER, a simple, lightweight, and efficient library for scaling from large corpora of unstructured texts to comprehensive and hierarchical taxonomies of concepts. The entire method is graphically visualized in Figure 1.

4.1 The Setup: Mining contexts of interest

The first stage of taxonomy construction, which we formally placed outside of the confines of TAXONOMYBUILDER, is the curation of candidate *contexts* (i.e., sentences) which will serve as the basis for an AI skills taxonomy. Using our large corpus of job postings, we take a keyword-based approach for an initial efficient mining of candidates, followed by a scoring procedure via *class vectors*.

Candidate mining. Due to very large-scale nature of our text corpora, we opt for an efficient, coarse-grained keyword-based search for the initial extraction of candidate contexts for an AI skills taxonomy. To build a comprehensive list of AI-related keywords, we consult various previous works that curate dictionaries of AI keywords sources (Baruffaldi et al., 2020; Alekseeva et al., 2021; Lou and Wu, 2021; Goldfarb et al., 2023; Lightcast, 2023; Maslej et al., 2025; Tambe, 2026).

We compiled the keywords from the above sources, performed light editing (e.g., splitting entries with '/' into two and removing parentheses), and then de-duplicated the list. This yielded a final set of 590 AI keywords, found in our repository.

[2] https://huggingface.co/datasets/loyoladatamining/usajobs Note: the corpus has been updated to include more postings than were used in this work.

We then performed an efficient keyword search using the PYAHOCORASICK library, keeping all sentences matching one or more of the keywords, with either a white space or sentence boundary (to avoid inner-string matches). We also enumerated the number of sentence matches per job posting. This keyword-based search yielded 5.58 million candidate sentences for NLx and 213k for USAJOBS. As a further filtering step, we kept only the candidate sentences from job postings with three or more matches (so as to remove noise and non-AI jobs). This resulted in a final set of 251k candidates for NLx and 19k candidates for USAJOBS.

Candidate scoring. Recognizing that a keyword-search introduces the potential for false positives (a prime example being *torch* as a welding instrument and not the machine learning library), we leveraged an efficient yet powerful method to score and rank all extracted candidates, allowing for further filtering. Motivated by the idea of *class label vector representations* (Schopf et al., 2020) and mean-max class scoring (Meisenbacher et al., 2024), we devised a hybrid scoring method that compares all candidate sentences to all keywords (here, our AI keywords) via cosine similarity of their sentence embeddings (Reimers and Gurevych, 2019).

For each given candidate sentence, its embedding is compared to all keyword embeddings, and then mean and max similarity scores are averaged to produce a final class-relatedness score. For numerical stability, we calculate this score on the basis of three embedding models: ALL-MINILM-L12-V2 (Reimers and Gurevych, 2019), GTE-LARGE (Li et al., 2023), and EMBEDDINGGEMMA-300M (Vera et al., 2025). The scores of these three models were averaged for the final score.

We note that while the three model approach was used for this work, the released TAXONOMYBUILDER code currently only supports class-based filtering with one embedding model of choice.

4.2 Forming the foundation

With the (large) set of candidate sentences mined, the next step becomes to distill these statements into a foundational layer of a taxonomy, one that is both representative of their diversity expressed among the candidates, but that also significantly compresses this information to form an accessible taxonomy. This process represents the first stage of TAXONOMYBUILDER: *forming the foundation*.

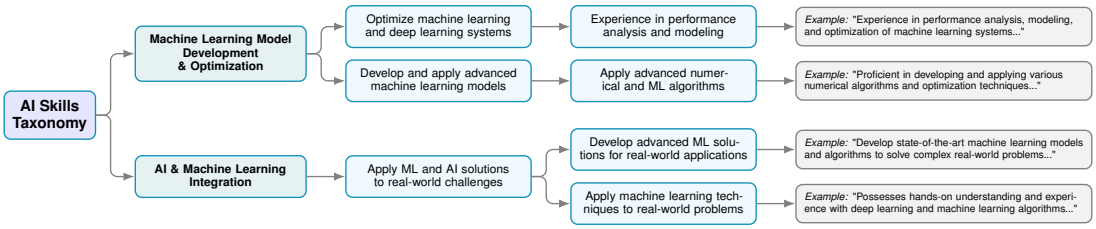

Figure 2: Abridged example of the taxonomy structure produced by TAXONOMYBUILDER for AI skills. The right-most nodes exemplify the initial foundation built from raw context candidates. Our methods then proceeds to build upwards ("left") upon this foundation. Note that *AI Skills Taxonomy* is inserted for completion.

Preliminaries: more and/or less data? How to effectively and intelligently leverage massive text corpora for an AI skills taxonomy using TAXONOMYBUILDER requires investigation and evaluation. Given the core set of candidate statements described above, we explore two divergent paths, asking the question of whether *more* (via augmentation) and/or *less* (via filtering) data is most sensible.

The first option, *data augmentation*, supplements the high-precision, low-recall, keyword-based candidate search with efficient embedding similarity augmentation. Given the set of (embedded) candidates, we perform a semantic search against all the documents in the respective corpus, *augmenting* the candidate set with any new contexts found that exceed a cosine similarity threshold (we choose 0.9). For this procedure, we solely use EMBEDDINGGEMMA-300M. Using this threshold augmented the NLx candidates with an additional 746k candidates and 31k candidates for USAJOBS.

We also explore the option of *less* data, in order to filter out noisy or irrelevant statements that have received a low class score. In this work, we experiment with keeping candidates (by score) above the top 75th, 50th, and 25th percentile. Note that such filtering also cascades to augmented candidates: if an original candidate is filtered out, all of its corresponding augmented sentences are as well.

Clustering. Given the target candidate contexts (following optional augmentation or filtering), the next step in TAXONOMYBUILDER is to cluster these contexts. Following the recommendations of Zhang et al. (2025), we choose HDBSCAN as the clustering algorithm (Campello et al., 2013), using the GPU-accelerated library provided by rapids.ai[3]. We perform clustering on embeddings that have been dimensionality reduced via UMAP to 10 dimensions (also using the CUML acceleration). We set the minimum cluster size to 5.

[3] https://rapids.ai/cuml-accel/

Since HDBSCAN by design assigns data points to a "noise cluster" if no fitting cluster is found, we explore the option of re-introducing such points to the clustering results via *soft clustering*. This process involves calculating the distance of a noise point to all cluster centroids, and assigning it the cluster of closest proximity. We leave this option as a third point of investigation, along with data augmentation and percentile filtering.

The results of this process is a collection of clustered candidates, each of which represent a single entry in the bottom (leaf) level of the taxonomy.

Generative cluster labeling. To transform clusters of related statements into concise entries for a taxonomy foundation, we leverage the generative capabilities of LLMs to produce accurate representations ("labels") of each cluster. This technique has been used in several previous works as an effective distillation method (Chen et al., 2023; Zeng et al., 2024; Zhang et al., 2025). For our main experiments, we use GPT-4O-MINI (OpenAI et al., 2024) with the prompt in Table 3 of the Appendix.

Label verification and pruning. To avoid error or redundancy propagation, we introduce a similarity-based check to the generated labels before proceeding to hierarchical taxonomy building. Specifically, the pairwise cosine similarity between each label embedding and all other labels is calculated, and all sets of labels exceeding a similarity threshold (here, 0.95) are aggregated into a consolidated label (using the zero-shot prompt of Table 4). The resulting collection of labels represent the leaf nodes of the taxonomy to be constructed.

4.3 Building up

The final stage of TAXONOMYBUILDER involves the iterative (vertical) creation of subsequent levels of the taxonomy. Given the foundational level labels, these are then used as the input for the next level, following the same embedding-clustering-

labeling pipeline as before. This process continues until one of two stopping criteria are met: (1) the number of labels generated at a given label are less than a user-defined value (default: 10), or (2) the number of levels generated has reached a user-defined threshold (default: 5). This mitigates the possibility of either scarce top levels, or taxonomies that are too tall and therefore hard to interpret.

An example AI skills taxonomy, excerpted from an output of our experiments, is depicted in Figure 2. The full taxonomies resulting from all experimental configurations are located in our repository.

5 Experimental Setup

The primary objective of our experiments is to evaluate the downstream effects of key design choices in the inclusion of data for taxonomy construction using TAXONOMYBUILDER. In particular, we study the impact of data augmentation, filtering measures, and soft clustering on the internal and external validity of the constructed AI skills taxonomies. Thus, we employ an intrinsic clustering-specific metric for clustering performance, as well as LLM-as-a-Judge criteria to evaluate the produced taxonomies as a whole. Finally, we empirically measure external validity by testing how well the constructed taxonomies generalize to real-world data, i.e., our held-out job posting test sets.

5.1 Taxonomy configurations

Using our two datasets, we construct an array of taxonomies based on different configurations, enabling a comparative study of key design decisions. In particular, we vary the following configurations:

- **Data Augmentation**: whether the original candidate sets are augmented with new examples from the corpus. We utilize EMBEDDINGGEMMA-300M with a similarity threshold of 0.9.
- **Percentile Filtering**: leveraging class vector scoring, at which score threshold are candidates filtered out. We choose three configurations: top 25th, 50th, and 75th percentile of scores, i.e., only keeping candidates within these percentiles.
- **Soft Clustering**: whether to include noise points back in clusters or not (via centroid proximity).

With the above configurations, we create and evaluate **12** taxonomies for each dataset (24 in total), leveraging the *TaxonomyBuilder* method.

5.2 Evaluation Metrics

We select three classes of evaluation metrics, spanning *clustering evaluation*, *LLM-based taxonomy assessment*, and *taxonomy domain coverage*.

Clustering evaluation. As the variation of our configurations, described above, can reasonably have a direct influence on the outcome of the HDBSCAN clustering, we employ a well-known cluster evaluation metric, namely the silhouette score. A higher scores in this metric implies that clusters are more well-defined. We use the SKLEARN implementation, and we average the per-layer scores across all levels of a constructed taxonomy.

LLM-as-a-Judge. To evaluate a constructed taxonomy holistically, we adopt the LLM-as-a-Judge approach of Zhang et al. (2025), who design an evaluation based on taxonomy metrics from Kaplan et al. (2022), for a set of four evaluation categories:

- **Clarity**: consists of four metrics, namely *precision*, *ambiguity*, *consistency*, and *accessibility*.
- **Hierarchical Coherence**: consists of *gradational specificity* and *parent-child coherence*.
- **Orthogonality**: *distinctiveness* and *non-overlap*.
- **Completeness**: *coverage*, *depth*, and *balance*.

These metrics are defined in the LLM-as-a-Judge prompt we use, which is slightly adapted from Zhang et al. (2025). This is found in Table 5 of the Appendix. We use GPT-5.4-NANO (2026-03-17) (Singh et al., 2025) with a temperature of 0.

Domain coverage. To evaluate the generalizability and representativeness of a constructed taxonomy, we perform domain coverage tests, adapted from the evaluation of Li et al. (2025).

This metric measures how well the base level statements ("labels") of a taxonomy can be used to map and classify real-world data. We first prepare a test set from the held-out data from each dataset (see Section 3), namely a random sample 10k job postings from NLx and 1k postings from USAJOBS. We sentence tokenize all of these postings (using NLTK), and prompt GPT-5.4-NANO and Google's GEMINI-3.1-FLASH-LITE (with temperature of 0) to label each sentence as an AI-related job skill or not. The full prompt used for this is found in Table 6 of the Appendix.

With the labeled test sets, we evaluate coverage by running embedding-based semantic search procedures at various *confidence levels*, or thresholds.

Dataset	Configuration		Silh.	LLM-as-a-Judge (Category Avg.)				Lenient Coverage @ τ				Strict Coverage @ τ				Best Util.
	Aug. / Soft	Pct.		Clar.	H.Coh.	Orth.	Comp.	0.9	0.8	0.7	0.6	0.9	0.8	0.7	0.6	
NLx	Y / Y	25	0.047	2.62	3.08	1.62	3.02	0.472	**0.595**	0.585	0.346	0.498	0.545	0.447	0.227	0.580
	Y / Y	50	0.056	2.86	3.25	1.81	2.96	0.470	0.592	0.603	0.376	0.497	0.547	0.464	0.253	0.604
	Y / Y	75	0.092	3.07	3.57	2.21	**3.29**	0.462	0.579	0.630	0.427	0.494	0.557	0.494	0.297	0.610
	Y / N	25	-0.000	2.89	3.07	1.79	2.90	**0.475**	0.592	0.576	0.316	**0.500**	0.540	0.436	0.201	0.580
	Y / N	50	0.032	2.83	3.15	1.85	2.93	0.473	0.590	0.597	0.362	0.499	0.543	0.458	0.241	0.608
	Y / N	75	0.074	2.82	3.36	2.14	2.90	0.465	0.576	0.626	0.402	0.495	0.553	0.488	0.276	0.604
	N / Y	25	0.073	2.75	3.20	1.90	3.07	0.460	0.554	0.702	0.546	0.493	**0.600**	0.599	0.398	0.468
	N / Y	50	**0.111**	**3.25**	**3.75**	**2.25**	3.22	0.457	0.538	**0.707**	0.602	0.489	0.598	0.636	0.445	0.814
	N / Y	75	0.065	2.88	3.25	2.00	3.17	0.456	0.504	0.682	**0.663**	0.489	0.570	**0.672**	**0.505**	0.844
	N / N	25	0.026	2.92	3.33	1.83	2.89	0.461	0.558	0.700	0.533	0.497	0.596	0.589	0.385	0.447
	N / N	50	0.079	2.83	3.50	1.83	3.22	0.459	0.541	0.705	0.599	0.493	0.596	0.623	0.441	0.837
	N / N	75	0.096	2.92	3.50	2.00	3.11	0.456	0.510	0.685	0.648	0.489	0.576	0.662	0.488	**0.851**
USAJOBS	Y / Y	25	0.155	3.34	2.70	2.14	2.73	0.485	**0.514**	0.543	0.319	**0.510**	0.518	0.469	0.244	0.294
	Y / Y	50	0.159	3.40	2.81	2.13	2.84	0.483	0.496	0.573	0.436	0.506	0.526	0.518	0.354	0.300
	Y / Y	75	0.430	2.75	1.75	1.50	2.33	0.481	0.490	0.586	0.584	0.501	0.525	0.559	0.466	0.823
	Y / N	25	0.561	3.37	2.76	2.28	2.78	**0.486**	**0.514**	0.553	0.330	0.507	0.518	0.476	0.254	0.296
	Y / N	50	0.516	3.47	2.79	2.26	2.95	0.483	0.498	0.576	0.425	0.506	0.525	0.517	0.345	0.317
	Y / N	75	0.577	3.08	2.50	1.50	2.56	0.481	0.491	0.583	0.556	0.500	0.531	0.552	0.446	0.839
	N / Y	25	0.345	3.33	1.61	1.56	2.04	0.483	0.492	**0.587**	0.587	0.505	**0.534**	0.572	0.469	**0.872**
	N / Y	50	0.362	3.50	1.93	2.07	2.05	0.481	0.489	0.577	0.607	0.503	0.527	0.588	0.486	0.862
	N / Y	75	0.112	**3.62**	**3.30**	**2.46**	3.06	0.480	0.484	0.542	**0.644**	0.499	0.516	**0.637**	**0.526**	0.829
	N / N	25	0.673	3.33	1.71	1.67	1.97	0.482	0.491	0.586	0.589	0.506	0.533	0.571	0.470	0.835
	N / N	50	**0.703**	2.50	1.50	1.50	2.33	0.482	0.489	0.571	0.601	0.503	0.529	0.582	0.483	0.852
	N / N	75	0.481	3.50	3.05	2.38	**3.08**	0.481	0.484	0.541	0.636	0.501	0.515	0.625	0.515	0.821

Table 1: Consolidated evaluation results of taxonomies constructed with TAXONOMYBUILDER across the NLx and USAJOBS Datasets. The different configurations are represented by whether they include data **aug**mentation (Y) or not (N), and similarly for **soft** clustering or not. LLM-as-a-Judge categories represent the average of their component metrics. The *Best Util* column reports the Label Utilization corresponding to the threshold (τ) that achieved the highest *strict* coverage. **Bold** values indicate the best score in each column per dataset.

We choose $\tau \in \{0.9, 0.8, 0.7, 0.6\}$ indicating that a sentence in question must match a taxonomy label with cosine similarity $> \tau$ to be classified as "AI". To calculate the similarity score for thresholding, we use an ensemble of three embedding models, akin to ensemble used for class scoring in Section 4. At each τ, we report the macro-F1 score, which represents the *coverage rate*. This includes a *lenient* score, where the ground truth of "AI" (label = 1) occurs when at least one LLM label is 1 (see above), and *strict* where both LLMs agree. At the best (strict) coverage rate, we also report the *label utilization rate*, i.e., the percentage of taxonomy statements successfully matched to the test set in the optimal (highest coverage) scenario. Together, these metrics illustrate both *how well* a taxonomy represents a domain and *how efficiently* this is done with respect to the breadth of the taxonomy.

6 Results and Statistical Analysis

The consolidated results are presented in Table 1, where LLM-as-a-Judge scores are reported as averages across the four categories. The label utilization rate is only reported for the corresponding best strict coverage rate. For the complete results, we refer the readers to Table 7 of the Appendix.

An analysis of the results shows that the best results are achieved generally when *either* data augmentation *or* soft clustering is used, and a majority (22/28) of the best scores per metric are achieved at or above the 50th percentile filtering mark. To verify these observations, as well as to investigate the main effects of the different taxonomy construction configurations, we perform a macro-level statistical analysis using a factorial ANOVA test. The ANOVA test was conducted in the aggregated metrics (i.e., as presented in Table 7) to evaluate the effects of data augmentation, percentile filtering, and soft clustering on our evaluation metrics ($2 \times 3 \times 2$ factors). The statistically significant results of this analysis are presented in Table 2, and the full results can be found in Table 8 of the Appendix.

7 Discussion

In the following, we reflect on the main findings of our work, and we discuss paths for future research.

Less is more. Our work centers on the important question of systemically understanding complex, shifting, and evolving domains such as *AI skills*, specifically through the construction of representative taxonomies from massive text corpora. While

Dataset	Metric	Significant Factor	F-value	p-value	η^2	Direction of Effect / Explanation
NLx	Silhouette Score	Percentile	4.36	$0.059^\dagger$	0.387	Stricter percentile filtering improves cluster fit
		Augmentation	3.68	$0.097^\dagger$	0.163	Augmentation marginally decreases cluster fit
	LLM: Orthogonality	Percentile	3.45	$0.091^\dagger$	0.467	Stricter percentile filtering improves orthogonality
	LLM: Completeness	Soft Clustering	3.92	$0.088^\dagger$	0.223	Soft clustering marginally improves completeness
	Coverage (Strict, $\tau = 0.8$)	Augmentation	40.81	**<0.001***	0.836	Augmentation severely reduces strict coverage rate
	Label Utilization	Percentile	4.60	$0.053^\dagger$	0.475	Stricter filtering drastically improves label utilization
USAJOBS	Silhouette Score	Soft Clustering	18.76	**0.003***	0.713	Soft clustering severely degrades cluster fit
	Label Utilization	Augmentation	13.48	**0.008***	0.518	Augmentation severely reduces label utilization

* Significant at $p < 0.05$; † Marginally significant at $p < 0.10$.

Table 2: Factorial ANOVA results for effects of taxonomy construction configurations on the reported metrics. Only significant ($p < 0.05$) and marginally significant ($p < 0.10$) effects are reported. Effect sizes are denoted by η^2.

the presence of such data may potentially serve as a fertile foundation for the challenging task of automatic taxonomy construction, we demonstrate that there are important design decisions revolving around *how to leverage this data*. Such questions have not been explored by previous works, who operate on much more compact datasets.

Several of our key findings point to a *less is more* philosophy when building taxonomies from large-scale data. We find that data augmentation does not help the quality of the resulting taxonomies, demonstrated by the fact that only one LLM-as-a-Judge category was won with data augmentation (out of the eight settings). In this light, using soft clustering after *excluding* data augmentation proved to be a winning recipe, achieving the best score in 20/28 cases. Thus, soft clustering likely only makes sense when not including augmented samples.

Studying the effect of percentile-based filtering measure reveals a similar trend. Taxonomies built with a weaker 25th percentile filter won zero LLM-as-a-Judge categories, and interestingly, only outperform all other taxonomies at very strict (high τ) coverage tests. As such, being too lenient with data inclusion (i.e., including more noisy points), can lead to sub-optimal domain-specific taxonomies.

Taxonomy building as trade-off balancing. We highlight another important finding, specifically in evaluating the *coverage* of taxonomies in a custom domain. One can observe an inverse relationship between data inclusion and coverage performance. Interestingly, at the strictest coverage threshold ($\tau = 0.9$), the coverage rates (both strict and lenient) are won exclusively by taxonomies built with a 25th percentile filter (more data). This is exactly the opposite for the lowest threshold ($\tau = 0.6$), at which exclusively 75th percentile taxonomies outperform all others. Intuitively, this might make sense, as larger, potentially noisy taxonomies may provide wide coverage and thereby excel at higher similarity thresholds, whereas more concise taxonomies can still perform better at lower thresholds due to being more well-defined and less cluttered.

Such findings point to the need to study *trade-offs* in automatic taxonomy construction, particularly in the balance between domain coverage and intra-taxonomy variability (i.e., noisiness). Data augmentation and more permissive filtering could serve certain use cases well, in which breadth is prioritized over intepretability. On the other hand, taxonomies that are repetitive or not focused enough on the domain in question may only increase ambiguity and lack usability or complicate comprehension. Therefore, although our results support the recommendation of less (data) being more powerful, we caution that this is case-specific, and ultimately, trade-offs should be carefully weighed.

8 Conclusion

We propose TAXONOMYBUILDER as a generalizable tool to taxonomize custom domains from large-scale text corpora. Using TAXONOMYBUILDER, we conduct a systematic investigation into the practical considerations of leveraging large amounts of data, focusing on questions of optimal data use for grounding taxonomy construction. Our results highlight that less is more, and data augmentation should be exercised with caution. In addition, performing selective filtering of candidate inputs to the construction process is often advantageous. As future work, we recommend (1) the continued exploration of design decisions in automatic taxonomy construction, (2) the carrying out of ablation studies regarding LLM choice, clustering parametrization, and taxonomy pruning, and (3) the formalization of trade-off-specific taxonomy evaluation metrics, focusing on real-world generalizability.

Limitations

We acknowledge the main limitations of our work, primarily in that we did not study the impact our our investigated factors (data augmentation, soft clustering, and percentile filtering) on other proposed LLM-based taxonomy construction methods. Rather, we focused on our proposed TAXONOMY-BUILDER method, as this framework was custom built for this systematic study, thereby not necessitating the modification of previous methods.

We also note the lack of an ablation study, as mentioned in the Conclusion, which we propose as direct follow-up work to ours. Important points of investigation could include but are not limited to: choice of LLM for label generation, the design of more intelligent methods to reduce noise in the context provided to the LLM (while maintaining representativeness, and conducting top-down checks to counteract bottom-up error propagation (e.g., induced via irrelevant candidate sentences).

Ethics Statement

We respect the confidentiality of data shared under the agreement with NLx and publicly release only aggregate information (the constructed taxonomies), and mitigate restrictions on sharing with the inclusion of a public corpus in the same domain.

We also note that the constructed taxonomies provided in our repository have been redacted to remove company names remaining in the leaf nodes.

References

Daron Acemoglu, David Autor, and Simon Johnson. 2026. Building Pro-Worker Artificial Intelligence.

Liudmila Alekseeva, José Azar, Mireia Giné, Sampsa Samila, and Bledi Taska. 2021. The demand for AI skills in the labor market. *Labour Economics*, 71:102002.

Stefano Baruffaldi, Brigitte van Beuzekom, Hélène Dernis, Dietmar Harhoff, Nandan Rao, David Rosenfeld, and Mariagrazia Squicciarini. 2020. Identifying and measuring developments in artificial intelligence: Making the impossible possible. *OECD Science, Technology and Industry Working Papers*, 2020(05).

Ricardo JGB Campello, Davoud Moulavi, and Jörg Sander. 2013. Density-based clustering based on hierarchical density estimates. In *Pacific-Asia conference on knowledge discovery and data mining*, pages 160–172. Springer.

Boqi Chen, Fandi Yi, and Dániel Varró. 2023. Prompting or fine-tuning? a comparative study of large language models for taxonomy construction. In *2023 ACM/IEEE International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C)*, pages 588–596.

Philipp Cimiano, Andreas Hotho, and Steffen Staab. 2004. Comparing conceptual, divisive and agglomerative clustering for learning taxonomies from text. In *Proceedings of the 16th European Conference on Artificial Intelligence*, ECAI'04, page 435–439, NLD. IOS Press.

Luis Espinosa-Anke, Jose Camacho-Collados, Claudio Delli Bovi, and Horacio Saggion. 2016. Supervised distributional hypernym discovery via domain adaptation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 424–435, Austin, Texas. Association for Computational Linguistics.

Trevor Fountain and Mirella Lapata. 2012. Taxonomy induction using hierarchical random graphs. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 466–476, Montréal, Canada. Association for Computational Linguistics.

Morgan R. Frank, David Autor, James E. Bessen, Erik Brynjolfsson, Manuel Cebrian, David J. Deming, Maryann Feldman, Matthew Groh, José Lobo, Esteban Moro, Dashun Wang, Hyejin Youn, and Iyad Rahwan. 2019. Toward understanding the impact of artificial intelligence on labor. *Proceedings of the National Academy of Sciences*, 116(14):6531–6539.

Ruiji Fu, Jiang Guo, Bing Qin, Wanxiang Che, Haifeng Wang, and Ting Liu. 2014. Learning semantic hierarchies via word embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1199–1209, Baltimore, Maryland. Association for Computational Linguistics.

Avi Goldfarb, Bledi Taska, and Florenta Teodoridis. 2023. Could machine learning be a general purpose technology? A comparison of emerging technologies using data from online job postings. *Research Policy*, 52(1):104653.

Allan D Gordon. 1996. Hierarchical classification. *Clustering and classification*, page 65.

Marti A. Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *COLING 1992 Volume 2: The 14th International Conference on Computational Linguistics*.

Chen Huang and Guoxiu He. 2025. Text clustering as classification with llms. In *Proceedings of the 2025 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region*, SIGIR-AP 2025, page 374–384, New York, NY, USA. Association for Computing Machinery.

Dino Ienco and Rosa Meo. 2008. Towards the automatic construction of conceptual taxonomies. In *International Conference on Data Warehousing and Knowledge Discovery*, pages 327–336. Springer.

Angelika Kaplan, Thomas Kühn, Sebastian Hahner, Niko Benkler, Jan Keim, Dominik Fuchß, Sophie Corallo, and Robert Heinrich. 2022. Introducing an evaluation method for taxonomies. In *Proceedings of the 26th International Conference on Evaluation and Assessment in Software Engineering*, EASE '22, page 311–316, New York, NY, USA. Association for Computing Machinery.

Priyanka Kargupta, Nan Zhang, Yunyi Zhang, Rui Zhang, Prasenjit Mitra, and Jiawei Han. 2025. TaxoAdapt: Aligning LLM-based multidimensional taxonomy construction to evolving research corpora. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 29834–29850, Vienna, Austria. Association for Computational Linguistics.

Zornitsa Kozareva and Eduard Hovy. 2010. A semi-supervised method to learn and construct taxonomies using the web. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1110–1118, Cambridge, MA. Association for Computational Linguistics.

Avishek Lahiri, Yufang Hou, and Debarshi Kumar Sanyal. 2025. TaxoAlign: Scholarly taxonomy generation using language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 30203–30223, Suzhou, China. Association for Computational Linguistics.

Josh Lerner and Jean Tirole. 2002. Some Simple Economics of Open Source. *The Journal of Industrial Economics*, 50(2):197–234.

Nan Li, Bo Kang, and Tijl De Bie. 2025. Building data-driven occupation taxonomies: A bottom-up multi-stage approach via semantic clustering and multi-agent collaboration. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 1596–1614, Suzhou (China). Association for Computational Linguistics.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023. Towards general text embeddings with multi-stage contrastive learning. *Preprint*, arXiv:2308.03281.

Lightcast. 2023. The Lightcast open skills taxonomy. Technical report, Lightcast. White paper. Formerly Burning Glass Technologies/Emsi Burning Glass.

Bowen Lou and Lynn Wu. 2021. AI on Drugs: Can Artificial Intelligence Accelerate Drug Development? Evidence from a Large-Scale Examination of Bio-Pharma Firms. *Management Information Systems Quarterly*, 45(3):1451–1482.

Yuning Mao, Xiang Ren, Jiaming Shen, Xiaotao Gu, and Jiawei Han. 2018. End-to-end reinforcement learning for automatic taxonomy induction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2462–2472, Melbourne, Australia. Association for Computational Linguistics.

Oleksandr Marchenko and Danylo Dvoichenkov. 2024. Taxorankconstruct: A novel rank-based iterative approach to taxonomy construction with large language models. In *ISS@ IT&I*, pages 11–27.

Anoush Margaryan. 2023. Artificial intelligence and skills in the workplace: An integrative research agenda. *Big Data & Society*, 10(2):20539517231206804.

Nestor Maslej, Loredana Fattorini, Raymond Perrault, Yolanda Gil, Vanessa Parli, Njenga Kariuki, Emily Capstick, Anka Reuel, Erik Brynjolfsson, John Etchemendy, Katrina Ligett, Terah Lyons, James Manyika, Juan Carlos Niebles, Yoav Shoham, Russell Wald, Toby Walsh, Armin Hamrah, Lapo Santarlasci, and 4 others. 2025. Artificial Intelligence Index Report 2025. *arXiv preprint*. ArXiv:2504.07139 [cs].

Stephen Meisenbacher, Tim Schopf, Weixin Yan, Patrick Holl, and Florian Matthes. 2024. An improved method for class-specific keyword extraction: A case study in the German business registry. In *Proceedings of the 20th Conference on Natural Language Processing (KONVENS 2024)*, pages 159–165, Vienna, Austria. Association for Computational Linguistics.

Ryszard S Michalski and Robert E Stepp. 1983. Learning from observation: Conceptual clustering. In *Machine learning*, pages 331–363. Elsevier.

Viktor Moskvoretskii, Irina Nikishina, Ekaterina Neminova, Alina Lobanova, Alexander Panchenko, and Chris Biemann. 2026. Large language models for creation, enrichment and evaluation of taxonomic graphs. *Semantic Web*, 17(1):22104968251404186.

Elham Motamedi, Inna Novalija, and Luis Rei. 2026. Semi-automatic hierarchical taxonomy creation from existing taxonomies with large language models. *Business & Information Systems Engineering*, pages 1–23.

Abhishek Nagaraj. 2022. The Private Impact of Public Data: Landsat Satellite Maps Increased Gold Discoveries and Encouraged Entry. *Management Science*, 68(1):564–582.

Abhishek Nagaraj, Esther Shears, and Mathijs de Vaan. 2020. Improving data access democratizes and diversifies science. *Proceedings of the National Academy of Sciences*, 117(38):23490–23498.

Engineering National Academies of Sciences, and Medicine. 2025. *Artificial Intelligence and the Future of Work*. The National Academies Press, Washington, DC.

Raúl Navarro-Almanza, Reyes Juárez-Ramírez, Guillermo Licea, and Juan R Castro. 2020. Automated ontology extraction from unstructured texts using deep learning. In *Intuitionistic and Type-2 fuzzy logic enhancements in neural and optimization algorithms: Theory and applications*, pages 727–755. Springer.

Robert C Nickerson, Upkar Varshney, and Jan Muntermann. 2013. A method for taxonomy development and its application in information systems. *European journal of information systems*, 22(3):336–359.

OpenAI, :, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Mądry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, and 401 others. 2024. Gpt-4o system card. *Preprint*, arXiv:2410.21276.

Sebastian Padó and Mirella Lapata. 2007. Dependency-based construction of semantic space models. *Computational Linguistics*, 33(2):161–199.

Marius Pasca. 2004. Acquisition of categorized named entities for web search. In *Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management*, CIKM '04, page 137–145, New York, NY, USA. Association for Computing Machinery.

Hoifung Poon and Pedro Domingos. 2010. Unsupervised ontology induction from text. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 296–305, Uppsala, Sweden. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. 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)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

William Resh, Keunyoung Lee, and Yi Ming. 2025. U.S. Federal Civil Position Job Postings (2018-2023).

Tim Schopf, Daniel Braun, and Florian Matthes. 2020. Semantic label representations with lbl2vec: A similarity-based approach for unsupervised text classification. In *International conference on web information systems and technologies*, pages 59–73. Springer.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, Akshay Nathan, Alan Luo, Alec Helyar, Aleksander Madry, Aleksandr Efremov, Aleksandra Spyra, Alex Baker-Whitcomb, Alex Beutel, Alex Karpenko, and 465 others. 2025. Openai gpt-5 system card. *Preprint*, arXiv:2601.03267.

Rion Snow, Daniel Jurafsky, and Andrew Y. Ng. 2006. Semantic taxonomy induction from heterogenous evidence. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 801–808, Sydney, Australia. Association for Computational Linguistics.

Prasanna B. Tambe. 2026. Reskilling the Workforce for AI: Domain Expertise and Algorithmic Literacy. *Management Science*, 72(1):515–537.

Paola Velardi, Stefano Faralli, and Roberto Navigli. 2013. OntoLearn reloaded: A graph-based algorithm for taxonomy induction. *Computational Linguistics*, 39(3):665–707.

Henrique Schechter Vera, Sahil Dua, Biao Zhang, Daniel Salz, Ryan Mullins, Sindhu Raghuram Panyam, Sara Smoot, Iftekhar Naim, Joe Zou, Feiyang Chen, Daniel Cer, Alice Lisak, Min Choi, Lucas Gonzalez, Omar Sanseviero, Glenn Cameron, Ian Ballantyne, Kat Black, Kaifeng Chen, and 70 others. 2025. Embeddinggemma: Powerful and lightweight text representations. *Preprint*, arXiv:2509.20354.

Binh Vu, Rashmi Govindraju Naik, Bao Khanh Nguyen, Sina Mehraeen, and Matthias Hemmje. 2025. Automated taxonomy construction using large language models: A comparative study of fine-tuning and prompt engineering. *Eng*, 6(11):283.

Mengting Wan, Tara Safavi, Sujay Kumar Jauhar, Yujin Kim, Scott Counts, Jennifer Neville, Siddharth Suri, Chirag Shah, Ryen W. White, Longqi Yang, Reid Andersen, Georg Buscher, Dhruv Joshi, and Nagu Rangan. 2024. Tnt-llm: Text mining at scale with large language models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '24, page 5836–5847, New York, NY, USA. Association for Computing Machinery.

Qingkai Zeng, Yuyang Bai, Zhaoxuan Tan, Shangbin Feng, Zhenwen Liang, Zhihan Zhang, and Meng Jiang. 2024. Chain-of-layer: Iteratively prompting large language models for taxonomy induction from limited examples. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, CIKM '24, page 3093–3102, New York, NY, USA. Association for Computing Machinery.

Chao Zhang, Fangbo Tao, Xiusi Chen, Jiaming Shen, Meng Jiang, Brian Sadler, Michelle Vanni, and Jiawei Han. 2018. Taxogen: Unsupervised topic taxonomy construction by adaptive term embedding and clustering. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '18, page 2701–2709, New York, NY, USA. Association for Computing Machinery.

Haiqi Zhang, Zhengyuan Zhu, Zeyu Zhang, and Chengkai Li. 2025. LLMTaxo: Leveraging large language models for constructing taxonomy of factual

claims from social media. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 19627–19641, Vienna, Austria. Association for Computational Linguistics.

Kun Zhu, Lizi Liao, Yuxuan Gu, Lei Huang, Xiaocheng Feng, and Bing Qin. 2025. Context-aware hierarchical taxonomy generation for scientific papers via LLM-guided multi-aspect clustering. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 15616–15634, Suzhou, China. Association for Computational Linguistics.

Ben Zweig. 2026. *Job Architecture: Building a Language for Workforce Intelligence*. Wiley, Hoboken.

A Supplemental

Prompts. Table 3 provides the LLM prompt used to create labels for collections of candidate sentences, in order to form a level of the taxonomy being constructed. Table 4 provides the prompt for the aggregation process (redundancy reduction).

On the evaluation side, Table 5 outlines the full LLM-as-a-Judge prompt, based on Zhang et al. (2025). Table 6 contains the prompt used to label the test sets for NLx and USAJOBS.

Results. Table 7 details the complete experimental results, providing a more full account than Table 1. Table 8 presents the full ANOVA test results, in supplement to the abridged version of Table 2.

You will be given a list of statements. Your job is to produce a single sentence that summarizes these statements into a coherent task / skill description. A task is a specfic activity or function that a person would be required to do on the job. A skill is the ability to perform a specific task and apply knowledge, particularly in the work context. Avoid using generalizations like "various" and "across domains". Answer simply with the generated description, nothing else is required. Provide your feedback as follows: Output::: Description: (GENERATED DESCRIPTION) Here are some examples: statements: ['develop or implement data analysis algorithms.', 'design and apply bioinformatics algorithms including unsupervised and supervised machine learning, dynamic programming, or graphic algorithms.', 'analyze or manipulate bioinformatics data using software packages, statistical applications, or data mining techniques.', 'develop or apply data mining and machine learning algorithms.', 'develop machine learning operations.', '+ develop and implement new computational and statistical methods.', 'he/she may also evaluate and develop novel algorithms and approaches for data analysis.', '* you will develop custom data models and algorithms to apply to data sets.'] Output::: Description: Develop or apply data mining and machine learning algorithms. statements: ['• basic computer skills.', 'basic computer software skills.', 'basic computer software skills, i.e.', 'be able to perform basic computer skills.', 'basic skills in use of computers and software programs.', 'skills: basic computer skills.', '* basic programming experience.', 'computer skills and basic knowledge of software applications.', '•basic computer skills.', '- basic computer skills.', 'skills: basic computer knowledge.', 'ability to perform basic computer skills.', 'basic problem solving skills associated with software applications used is expected.', '- computer programming skills.'] Output::: Description: Has basic computer skills, including programming and use of computers and software programs. Now here are the actual statements. statements: {CANDIDATES} Output::: Description:

Table 3: Prompt for creation of leaf-level taxonomy entries.

You will be given a list of statements. They all express a similar meaning, with slight variations and differences. Your job is to produce a single sentence that summarizes these statements into one coherent description, that captures the essence and intricacies of all statements. Answer simply with the generated description, nothing else is required. Provide your feedback as follows: Output::: Description: (GENERATED DESCRIPTION) Now here are the actual statements. statements: {LABELS} Output::: Description:

Table 4: Prompt for aggregation of similar labels (for label pruning).

You will be given a taxonomy related to AI skills in the workplace. Please evaluate the taxonomy in the json format using the evaluation metrics provided. Give each evaluation criteria a score from 1-5. No additional explanation is necessary. Output only the json results.
****
Below is the taxonomy:
{TAXONOMY JSON STRING}
****
Here are the metrics:
Taxonomy Evaluation Metrics
Give the score from [1-5] 5: best/strongly agree, 1: worst/strongly disagree

Clarity: Assess whether the topic labels are clear, precise, and unambiguous.
Purpose: Ensure that each topic label communicates its content effectively to avoid confusion.
Evaluation Criteria:
• Precision: Each topic label uses specific and well-defined terms.
o Score:
• Unambiguity: Topic labels should have only one interpretation, preventing misunderstanding.
o Score:
• Consistency: Use of terminology is consistent across all levels of the taxonomy.
o Score:
• Accessibility: Language is straightforward, avoiding jargon where possible unless it is standard within the covered domain.
o Score:

Hierarchical Coherence: Assess whether the taxonomy follows a clear and meaningful hierarchical structure.
Purpose: Ensure that the taxonomy's structure facilitates easy navigation and understanding by clearly organizing information from the most general to the most specific.
Evaluation Criteria:
• Gradational Specificity: There is a logical progression from broader to more specific categories.
o Score:
• Parent-Child Coherence: Parent-child relationships are well-formed, ensuring that child nodes logically belong to their parent nodes.
o Score:
• Consistency: The hierarchy maintains consistent levels of detail throughout the taxonomy, ensuring that no topics are too broad or too narrow relative to others at the same level.
o Score:

Orthogonality: Assess whether the topics are well-differentiated without duplication.
Purpose: Maintain distinct boundaries between topics to ensure that each topic captures unique aspects of the domain.
Evaluation Criteria:
• Distinctiveness: Topics at each level progressively add meaningful distinctions rather than just rephrasing broader topics.
o Score:
• Non-overlap: For each topic, there is minimal to no overlap in the scope or content with other topics.
o Score:

Completeness: Assess whether the taxonomy captures a broad and representative set of topics across different aspects of the domain.
Purpose: Cover as many areas of the topic to ensure the taxonomy is comprehensive.
Evaluation Criteria:
• Domain Coverage: The taxonomy covers a variety of significant aspects of the domain it represents.
o Score:
• Depth: The taxonomy provides sufficient depth in each branch to capture nuanced distinctions within topics.
o Score:
• Balance: The topics are evenly distributed across the taxonomy. This involves assessing whether some branches are disproportionately detailed while others are underdeveloped, which could lead to an imbalance that might skew the taxonomy's effectiveness and navigability.
o Score:

Important: ensure that the output is in json, and following the above-given criteria names exactly.

Now please provide the scores.

Table 5: LLM-as-a-Judge prompt for comprehensive taxonomy evaluation.

You will be given a dictionary of statements from a job posting, each mapped to a unique ID (integer).
Your job is to indicate which of the statements (marked by their IDs), contain information related to any Artificial Intelligence (AI) task, skill, activity, expertise, or requirement.
"AI-related" can also include soft skills required for AI jobs in addition to technical abilities or tasks.
Make sure, though, that only AI-related statements are chosen!
Answer simply with a comma-separated Python list of ONLY the statement IDs marked as AI-related. Make sure to stick to this output format!

Provide your feedback as follows:

Output:::
Classification: (LIST OF IDENTIFIED AI-RELATED STATEMENT IDs)

Now here are the actual statements.

texts: {DICTIONARY OF MAPPED JOB POSTING SENTENCES}

Output:::
Classification:

Table 6: Prompt for LLM labeling of test set for domain coverage tests.

Data	Config		Silh.	Clarity				H. Coh.		Orth.		Comp.			Cov (0.9)			Cov (0.8)			Cov (0.7)			Cov (0.6)		
	Aug/Soft	Pct		Pr	Un	Co	Ac	Gr	PC	Di	NO	Co	De	Ba	L	S	U	L	S	U	L	S	U	L	S	U
NLx	Y / Y	25	0.047	3.14	2.33	2.14	2.86	2.86	3.29	2.10	1.14	3.62	3.43	2.00	0.472	0.498	0.101	0.595	0.545	0.580	0.585	0.447	0.813	0.346	0.227	0.847
	Y / Y	50	0.056	3.50	2.69	2.13	3.13	3.06	3.44	2.19	1.44	3.75	3.13	2.00	0.470	0.497	0.091	0.592	0.547	0.604	0.603	0.464	0.838	0.376	0.253	0.871
	Y / Y	75	0.092	3.71	2.86	2.43	3.29	3.43	3.71	2.57	1.86	4.00	3.71	2.14	0.462	0.494	0.058	0.579	0.557	0.610	0.630	0.494	0.867	0.427	0.297	0.899
	Y / N	25	-0.000	3.43	2.43	2.14	3.57	2.86	3.29	2.29	1.29	3.71	3.00	2.00	0.475	0.500	0.113	0.592	0.540	0.580	0.576	0.436	0.815	0.316	0.201	0.852
	Y / N	50	0.032	3.50	2.50	2.20	3.10	2.90	3.40	2.30	1.40	3.60	3.20	2.00	0.473	0.499	0.104	0.590	0.543	0.608	0.597	0.458	0.850	0.362	0.241	0.881
	Y / N	75	0.074	3.57	2.71	2.00	3.00	3.00	3.71	2.57	1.71	3.57	3.14	2.00	0.465	0.495	0.067	0.576	0.553	0.604	0.626	0.488	0.870	0.402	0.276	0.907
	N / Y	25	0.073	3.40	2.40	2.00	3.20	2.80	3.60	2.40	1.40	3.80	3.40	2.00	0.460	0.493	0.042	0.554	0.600	0.468	0.702	0.599	0.824	0.546	0.398	0.897
	N / Y	50	0.111	4.00	3.00	2.83	3.17	3.50	4.00	2.67	1.83	4.00	3.67	2.00	0.457	0.489	0.031	0.538	0.598	0.460	0.707	0.636	0.814	0.602	0.445	0.908
	N / Y	75	0.065	3.50	2.50	2.50	3.00	3.00	3.50	2.50	1.50	4.00	3.50	2.00	0.456	0.489	0.025	0.504	0.570	0.447	0.682	0.672	0.844	0.663	0.505	0.936
	N / N	25	0.026	3.67	2.67	2.00	3.33	3.00	3.67	2.33	1.33	3.67	3.00	2.00	0.461	0.497	0.056	0.558	0.596	0.447	0.700	0.589	0.829	0.533	0.385	0.916
	N / N	50	0.079	3.67	2.67	2.00	3.00	3.33	3.67	2.33	1.33	4.00	3.67	2.00	0.459	0.493	0.043	0.541	0.596	0.447	0.705	0.623	0.837	0.599	0.441	0.924
	N / N	75	0.096	3.67	2.67	2.33	3.00	3.33	3.67	2.33	1.67	4.00	3.33	2.00	0.456	0.489	0.031	0.510	0.576	0.451	0.685	0.662	0.851	0.648	0.488	0.943
USAJOBS	Y / Y	25	0.155	3.92	3.45	2.61	3.37	2.16	3.24	2.45	1.83	3.02	2.55	2.63	0.485	0.510	0.060	0.514	0.518	0.294	0.543	0.469	0.846	0.319	0.244	0.931
	Y / Y	50	0.159	3.97	3.54	2.75	3.33	2.18	3.43	2.38	1.89	3.11	2.78	2.64	0.483	0.506	0.070	0.496	0.526	0.300	0.573	0.518	0.848	0.436	0.354	0.933
	Y / Y	75	0.430	3.50	2.50	2.00	3.00	2.00	1.50	2.00	1.00	3.00	2.00	2.00	0.481	0.501	0.035	0.490	0.525	0.253	0.586	0.559	0.823	0.584	0.466	0.924
	Y / N	25	0.561	3.90	3.51	2.73	3.34	2.19	3.32	2.57	2.00	3.01	2.59	2.73	0.486	0.507	0.066	0.514	0.518	0.296	0.553	0.476	0.835	0.330	0.254	0.922
	Y / N	50	0.516	3.96	3.62	2.87	3.44	2.16	3.41	2.50	2.03	3.09	2.84	2.91	0.483	0.506	0.070	0.498	0.525	0.317	0.576	0.517	0.855	0.425	0.345	0.948
	Y / N	75	0.577	3.33	2.67	2.67	3.67	2.33	2.67	2.00	1.00	3.00	2.00	2.67	0.481	0.500	0.028	0.491	0.531	0.271	0.583	0.552	0.839	0.556	0.446	0.947
	N / Y	25	0.345	3.67	3.44	1.89	4.33	1.56	1.67	1.44	1.67	1.67	1.78	2.67	0.483	0.505	0.053	0.492	0.534	0.275	0.587	0.572	0.872	0.587	0.469	0.947
	N / Y	50	0.362	3.57	3.71	2.29	4.43	1.57	2.29	1.71	2.43	1.86	1.71	2.57	0.481	0.503	0.028	0.489	0.527	0.245	0.577	0.588	0.862	0.607	0.486	0.946
	N / Y	75	0.112	4.26	3.85	3.07	3.30	2.67	3.93	2.85	2.07	3.33	3.00	2.85	0.480	0.499	0.014	0.484	0.516	0.157	0.542	0.637	0.829	0.644	0.526	0.968
	N / N	25	0.673	3.42	3.42	2.33	4.17	1.58	1.83	1.50	1.83	1.67	1.67	2.58	0.482	0.506	0.050	0.491	0.533	0.255	0.586	0.571	0.835	0.589	0.470	0.950
	N / N	50	0.703	2.50	2.00	2.00	3.50	1.50	1.50	2.00	1.00	3.00	2.50	1.50	0.482	0.503	0.056	0.489	0.529	0.237	0.571	0.582	0.852	0.601	0.483	0.959
	N / N	75	0.481	4.20	3.70	2.90	3.20	2.40	3.70	2.75	2.00	3.50	3.05	2.70	0.481	0.501	0.037	0.484	0.515	0.174	0.541	0.625	0.821	0.636	0.515	0.972

Table 7: Comprehensive evaluation results of taxonomies constructed with TAXONOMYBUILDER across the NLx and USAJOBS Datasets. The different configurations are represented by whether they include data augmentation (Y) or not (N), and similarly for soft clustering or not. *Pct* denotes the percentile filtering threshold. All LLM-as-a-Judge metrics are reported in the order they were introduced in the paper, and are abbreviated with two representative letters. Domain Coverage (*Cov)* metrics include Lenient (L), Strict (S), and Utilization (U) across four thresholds.

Dataset	Evaluation Metric	Factor	F-value	p-value	η^2
NLx	Silhouette Score	Augmentation	3.68	0.097†	0.163
		Soft Clustering	3.11	0.121	0.138
		Percentile	4.36	0.059†	0.387
	LLM: Clarity	Augmentation	0.63	0.454	0.065
		Soft Clustering	0.14	0.716	0.015
		Percentile	0.91	0.444	0.190
	LLM: Hierarchical Coherence	Augmentation	2.82	0.137	0.190
		Soft Clustering	0.09	0.770	0.006
		Percentile	2.48	0.153	0.334
	LLM: Orthogonality	Augmentation	0.48	0.512	0.032
		Soft Clustering	0.38	0.555	0.026
		Percentile	3.45	0.091†	0.467
	LLM: Completeness	Augmentation	2.98	0.128	0.170
		Soft Clustering	3.92	0.088†	0.223
		Percentile	1.84	0.228	0.209
	Strict Coverage ($\tau = 0.8$)	Augmentation	40.81	**<0.001***	0.836
		Soft Clustering	0.11	0.750	0.002
		Percentile	0.46	0.650	0.019
	Best Label Utilization	Augmentation	3.18	0.118	0.164
		Soft Clustering	<0.01	0.986	<0.001
		Percentile	4.60	0.053†	0.475
USAJOBS	Silhouette Score	Augmentation	0.38	0.556	0.015
		Soft Clustering	18.76	**0.003***	0.713
		Percentile	0.09	0.912	0.007
	LLM: Clarity	Augmentation	0.07	0.797	0.009
		Soft Clustering	0.25	0.634	0.033
		Percentile	0.11	0.895	0.030
	LLM: Hierarchical Coherence	Augmentation	0.83	0.392	0.094
		Soft Clustering	0.01	0.933	0.001
		Percentile	0.50	0.628	0.113
	LLM: Orthogonality	Augmentation	0.01	0.920	0.002
		Soft Clustering	0.03	0.873	0.004
		Percentile	0.03	0.973	0.008
	LLM: Completeness	Augmentation	1.27	0.297	0.127
		Soft Clustering	0.18	0.686	0.018
		Percentile	0.79	0.489	0.158
	Strict Coverage ($\tau = 0.8$)	Augmentation	0.17	0.691	0.021
		Soft Clustering	0.04	0.856	0.004
		Percentile	0.48	0.640	0.117
	Best Label Utilization	Augmentation	13.48	**0.008***	0.518
		Soft Clustering	<0.01	0.974	<0.001
		Percentile	2.77	0.130	0.213

* Significant at $p < 0.05$; † Marginally significant at $p < 0.10$.

Table 8: Complete factorial ANOVA results. Significant ($p < 0.05$) and marginally significant ($p < 0.10$) effects are denoted. Effect size is represented by η^2.