DOI: 10.15439/2024F9098
ISSN 2300-5963 ACSIS, Vol. 40

No Train, No Pain? Assessing the Ability of LLMs for Text Classification with no Finetuning

Richard Fechner*† , Jens Dörpinghaus*‡
* Federal Institute for Vocational Education and Training (BIBB), Bonn, Germany,
Correspondence: richard.fechner@bibb.de,
jens.doerpinghaus@bibb.de, https://orcid.org/0000-0003-0245-7752
† University of Tübingen, Germany
‡ University Koblenz, Koblenz, Germany

***Abstract*—Modern SotA Text Classification algorithms depend heavily on well annotated and diverse data capturing the intricacies of the unknown data distribution. What options do we have when labeled data is sparse or annotation is expensive and time consuming? With the advent of strong LLM backbones, we have another option at our disposal: Text Classification by making use of the reasoning ability and the strong general prior of contemporary foundation models. In this work we assess the ability of cutting edge LLMs for Text Classification and find that for the right combination of backbone and prompt strategy we're able to near-rival trained baselines for the advanced task of mapping job-postings to a taxonomy of industrial sectors without any finetuning. All our code is made publicly available at our github repository[1].**

I. Introduction

TEXT classification is a widely used technology with a broad range of applications. However, it is rare for a one-size-fits-all solution to exist, and the situation becomes even more complex when the availability of training data and the complexity of clustering questions are taken into account. In previous research, we have worked on the domain of industrial sectors in labor market research data, see [1]. The classification of industrial sectors is of great importance, yet the range of available textual data is vast and only a limited amount of annotated training data exists. It was demonstrated that a categorization is possible, yet the quality of the categorization depends on both the training and evaluation data. Consequently, the specific clustering is dependent on the application and the research question.

For instance, all approaches failed for job advertisements, which are often used in labor market research. Conversely, a satisfactory recall was achieved on Wikipedia data. The proposed method was not yet ready for productive use, but it demonstrated that the initial research question was challenging due to the diversity of data and expected outcomes, as well as the interdisciplinary nature of the research. In this specific area, educational research and the social sciences had a different perspective on industrial sectors than, for example, economics. Therefore, understanding the correct classification depends not only on the research questions but also on the perspective of different scientific domains.

This paper will build upon the existing body of research on the classification of online job advertisements (OJAs) in industrial sectors. Our primary research questions are as follows:

1) *How may we harness the strong prior and reasoning ability of LLMs for knowledge intensive Text Classification directly when we have little to no labeled data?*
2) *How do different prompting strategies and models perform?*

When classifying a job-posting, we may classify the industrial section (IS) of the advertised job or the IS of the company posting the inquiry. However, the IS of the company and the jobad musn't match, i.e. a bakery might post a jobad for a roofer or IT-specialist. *In the following work we're concerned with classifying the IS of the company.* By nature, job-postings contain more information about the job and information about the company besides the name is sparse. The process of annotation is hence very time intensive and tedious as missing information about the company has to be searched on the web, analysed and finally combined with prior knowledge to obtain a good classification. Additionally, text data is often noisy, containing web-scraping boilerplate-text. The human universal prior allows for an easy differentiation of all these effects to the point, where we may perceive a problem at hand to be solvable "out of the box" for machines, when in fact some problems are impossible to solve when not equipped with prior knowledge and reasoning ability. A classical example is an agent for a self-driving car. Both the human driver and the algorithm perceive the same information (stereo RGB images), yet we have no agent driver which can reliably navigate in changing environments. In the same sense, text-classification is perceived as an easy task, where in reality it may be very hard. For specific data a certain amount of reasoning capability is needed in order to perform classification. Hypotheses have to be weighted against each other, even among annotators it is often not clear what class a jobposting may belong to. All the more important is the ability of a classification to be interpretable in the sense that a second annotator might judge the reasoning steps taken that led to the final classification.

[1] https://github.com/rfechner/fedcsis24-llm-textcat

Topical area: Advanced Artificial Intelligence in Applications

II. Related Work

We split this section into two subsections, the first giving a brief (and incomplete) summary of the recent history of text-classification, putting emphasis on neural methods, in particular the dominant Transformer architectures like BERT and the alignment of modern LLMs. The second subsection discusses the related work in the general domain of labor market analysis.

A. On the evolution of Neural Text Classification

Neural approaches to Text-Classification have gained attention since the introduction to architectures like the LSTM [2] and later the Transformer [3] from which the latter emerged as the popular architectural choice for working with text data. The BERT architecture [4] and its derivatives [5], [6], [7] are widely used and are still a common choice for sentence, text or document classification [8]. On a meta-level, we'd like to narrow down the choices for the next token by injecting more information either during inference or training time. Works like TransformerXL [9] or Longformer [10] try to trade off performance and context size which is imperative to capturing semantics in a text. In the domain of job-advertisements the authos of [11] have used continous pre-training on in-domain data (i.e. job-postings) to reduce the confusion of the language model on downstream tasks. For recent large models like ones of the LLaMA family [12] this approach becomes unattractive as the amount of data and compute needed to pre-train a model is likely very large. Instead we'd like to adapt the output distribution of the LLM by exterior methods and rely more heavily on the models reasoning capability. Aligning LLMs to conform to desired behaviour and alleviating the common mishaps of LLMs such as hallucinations or the lack of structure in the models answer is an area of active research. Techniques such as prompt engineering are among the most natural ways of alignment. On the other hand, prompts may mislead a LLM and throw it off in such a way that makes it ignore all previous safety instructions, leading to possible misuse [13]. A prompting strategy like Chain-of-Thought [14] or Tree-of-Thought [15] has been shown to substantially improve model performance. Addressing the issue of hallucination and lack of up-to-date in-domain knowledge is Retrieval Augmented Generation (RAG) which augments the token generation of an LLM with context provided by a so-called retriever [16]. Most recent work by the open source community was focused on creating so called chains of Language Models or "LangChains" [17] for short, structuring the token-generation process and unifying the previously mentioned exterior alignment methods. These methods in turn suffer from error accumulation over the multiple prediction steps.

B. On Related work in labor market analysis

Very little work has been done in this area. There are several applications for the given research question: For example, Pejic et al. state the need to analyse Industry 4.0 skills, but do not present a generic categorization approach, but rather pre-select job advertisements according to their needs [18]. Chaisricharoen et al. noted the importance of industrial sectors for legal categories. However, their work is limited to industry-standard keywords [19]. For the generic categorization of English texts, some work has been done by McCallum [20] and Kibriya et al. [21]. However, the data and industrial sectors are mainly for marketing purposes and cannot be used in economic and sociological research. Several other works rely on these data-sets, see for example [22], [23], which underlines the general need for publicly available training and evaluation data.

Text mining on labor market data is a widely considered topic. For an automated analysis of labor-market related texts, the situation in German-speaking countries like Germany, Austria and Switzerland is not much different to English-speaking countries: "Catalogs play a valuable role in providing a standardized language for the activities that people perform in the labor market" [24]. However, while these catalogs are widely used for creating and computing statical values, for managing labor market and educational needs or for recommending trainings and jobs, there is no single ground truth. According to Rodrigues et al., one reason for this could be the fact that labor market concepts are modeled by multiple disciplines, each with a different perspective on the labor market [25]. For German texts, in particular job advertisements, Gnehm et al.[26] introduced transfer learning and domain adaptation approaches with jobBERT-de and jobGBERT. This model was also used for the detection of skill requirements in German job advertisements [27], [28].

For regional data, especially in German-speaking countries, industrial sectors are widely used as a basis for economic and labour market research, see for example [29], [30], [31], they are particularly important for future skills and qualifications [32]. Although classification is a key issue for industrial sectors, see [33], little research has been carried out using computational methods. Examples are mainly limited to regional industries [34] or agriculture and green economy [35].

To our knowledge, no work has been done on German texts. Company data are usually collected and sold by commercial providers such as statista. There is also an online guide from the Federal Office of Economic Affairs and Export Control (BAFA) ("Merkblatt Kurzanleitung Wirtschaftszweigklassifikation"[2]), but this is only a short version of the data available from the Federal Statistical Office. Therefore, we will now discuss the available data.

III. Data

As discussed in the Appendix section on general Information about the German Industrial Sector Taxonomy WZ-2008 (See Appendix: A), several classifications of industrial sectors exist. We will continue by giving insight into the dataset construction and annotation process.

[2] https://www.bafa.de/SharedDocs/Downloads/DE/Wirtschaft/unb_kurzanleitung_wirtschaftszweigklassifikation.pdf.

A. Dataset construction and Annotation Process

Out of a large database of unlabeled job-postings, we drew a sample of about 2000 Jobpostings (1976. Assuming a general “distribution of jobads” $p(x)$, it should be noted that the drawn samples came from another (conditional) distribution $p(x|y)$ where y is the actual advertised job. More precisely we drew from a distribution which advertised open positions for roofers overproportionally. Hence, there exists a strong bias in the evaluation dataset, which is further discussed in the section on bias.

A small team of annotators took extensive time and co-ordinantion to annotate these jobpostings. The process of annotation includes reading through the sample, gathering information about the industrial section of the employer and finally coming up with hypotheses, which are then checked for validity. In the end, one has to verify that the hypothesized industrial section matches. Later, we will discuss how we modelled the prompting strategy after the annotation process. Invalid data, e.g. datapoints which were not actually jobpostings but advertisements or simply degenerate, were filtered out during the annotation process. We split the dataset into a 0.8-train, 0.1-test and 0.1-validation sets, to train and evaluate baseline models. The sample distribution was preserved inside the testset. To gain intuition on what the sample label frequencies (See Appendix: B) and examples of the data (See Appendix: C), we refer the reader to the Appendix.

B. Online Job Advertisements

In our research, we focus on several large corpora for OJAs. The first dataset was obtained from the German Federal Employment Agency (Bundesagentur für Arbeit – BA). This dataset contains approximately 5.5 million OJAs spanning the years 2013 to 2022. This portal is one of Germany’s largest job portals. The OJA records include several metadata, including a job classification (KldB) and industrial sectors according to WZ08. All data is manually curated. The second corpus comprises approximately 4 million OJAs from various data sources, including job portals such as Academics, Monster, and BA. This data contains several metadata, a classification of occupations according to ISCO, and industrial sectors on WZ08. However, it should be noted that these annotations are not manually curated but rather the result of unknown AI approaches which do not have a high level of quality.

IV. Method

First it should be noted that our experiments were strongly influenced by the fact that most LLMs were most likely trained on the WZ1993, WZ2003 and WZ2008 taxonomies. This opens up possibilties for prompts, as we may assume that the model has some form of understanding of the classes it is supposed to map onto. We conducted experiments on a small group of openly avaiable (open weights) contemporary Instruction finetuned LLMs using the python-ollama library [36] running on a local NVIDIA L40 GPU. At this point, we’d like to note that due to data privacy laws we are unable to test API-models as GPT-4o or Claude Sonnet. We tested several prompt strategies . When designing the prompts, we stuck to the general principle for success when engineering a prompt: Inducing a bias towards clear and small, step by step reasoning. The outputs of the model are verified heuristically at each intermediate step, making sure that the output is well behaved. At the end, an output parser makes a final response validation, making sure that the predicted class is valid. For invalid assistant responses, the chat is at most repeated a fixed number of times until the query is failed for the specific datapoint. Failure was most commonly due to an invalid output format or invalid classification (i.e. hallucination) of the model.

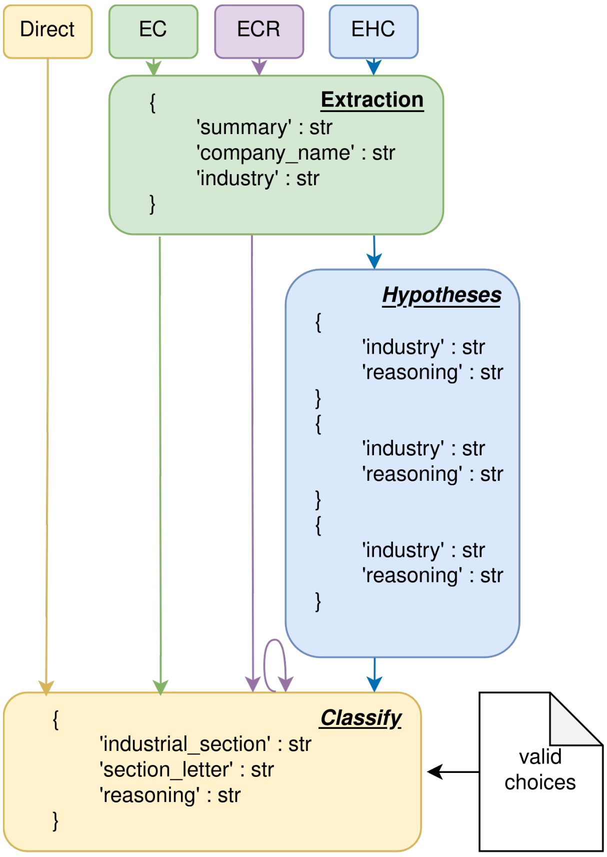

Fig. 1. Different strategies for prompting : “Direct”, “Extract-Classify” (EC), “Extract-Classify-Reflect” (ECR) and “Extract-(Generate) Hypotheses-Classify” (EHC). Input is a raw jobposting. All outputs are formatted in JSON. Output is verfied for a valid industrial section to ensure a non-degenerate answer.

For the baseline model, we finetuned a german distilBERT model [37] and a finetuned model [38] on the text classification task using the `transformers` library and Focal Loss [39] to put more emphasis on low-frequency classes. Additionally, we trained a few more standart classifiers from the `sklearn` library [40].

V. Results

Generally, we can see in the results that bigger models outperform smaller ones for Direct prompting. The 70B-parameter version of LLaMA3 (See Table I, ✠) reaches

TABLE I
METRICS FOR DIFFERENT LLMS, PROMPTING STRATEGIES AND PROMPT TYPES. LARGE MODELS (LLAMA3:70B AND COMMAND-R:35B) OUTPERFORM SMALLER ONES.

			Metrics (Macro/Weighted)			
LLM	Strategy	Prompt Type	Precision	Recall	F1 Score	Failed Samples
llama3:8b	Direct	zero_shot	0.01 / 0.00	0.12 / 0.01	0.02 / 0.00	0
		one_shot	0.04 / 0.13	0.03 / 0.01	0.01 / 0.02	0
		few_shot	0.11 / 0.61	0.05 / 0.16	0.04 / 0.19	2
	EC	zero_shot	0.21 / 0.73	0.20 / 0.42	0.14 / 0.39	16
		one_shot	0.04 / 0.28	0.07 / 0.08	0.02 / 0.10	0
		few_shot	0.00 / 0.00	0.06 / 0.01	0.00 / 0.00	2
	ECR	zero_shot	0.15 / 0.62	0.22 / 0.54	0.15 / **0.53**	15
		one_shot	0.03 / 0.23	0.08 / 0.16	0.03 / 0.18	0
		few_shot	0.00 / 0.00	0.06 / 0.01	0.00 / 0.00	6
	EHC	zero_shot★	**0.27 / 0.69**	0.21 / 0.52	**0.19** / 0.49	12
		one_shot⧫	0.04 / 0.20	0.11 / 0.10	0.103 / 0.11	31
		few_shot⧫	0.09 / 0.47	0.14 / 0.04	0.02 / 0.02	84
aya:8b	Direct	zero_shot	0.00 / 0.00	0.03 / 0.01	0.00 / 0.00	0
		one_shot	0.06 / 0.47	0.06 / 0.02	0.00 / 0.00	0
		few_shot	0.03 / 0.25	0.04 / 0.31	0.04 / 0.27	5
	EC	zero_shot	0.12 / 0.54	0.14 / 0.48	0.10 / 0.40	2
		one_shot	0.00 / 0.00	0.06 / 0.01	0.00 / 0.00	0
		few_shot	0.03 / 0.23	0.01 / 0.01	0.00 / 0.02	0
	ECR	zero_shot⧫	0.11 / 0.48	0.23 / 0.50	0.13 / 0.44	28
		one_shot⧫	0.00 / 0.00	0.07 / 0.02	0.00 / 0.00	33
		few_shot	0.00 / 0.00	0.07 / 0.01	0.00 / 0.00	2
	EHC	zero_shot⧫	0.17 / 0.67	0.15 / 0.54	0.11 / 0.41	97
		one_shot	0.07 / 0.47	0.08 / 0.03	0.01 / 0.03	5
		few_shot	0.10 / 0.57	0.10 / 0.02	0.03 / 0.03	3
gemma:7b	Direct	zero_shot	0.07 / 0.39	0.04 / 0.02	0.01 / 0.02	0
		one_shot⧫	0.00 / 0.00	0.02 / 0.01	0.00 / 0.00	25
		few_shot⧫	0.04 / 0.27	0.06 / 0.46	0.05 / 0.34	34
	EC	zero_shot	0.16 / 0.32	0.14 / 0.44	0.15 / 0.37	3
		one_shot⧫	0.00 / 0.00	0.09 / 0.03	0.01 / 0.00	131
		few_shot⧫	0.03 / 0.15	0.07 / 0.38	0.04 / 0.21	76
	ECR	zero_shot⧫	0.14 / 0.36	0.22 / 0.46	0.14 / 0.40	47
		one_shot⧫	0.00 / 0.00	0.00 / 0.00	0.00 / 0.00	208
		few_shot⧫	0.04 / 0.14	0.09 / 0.38	0.05 / 0.21	143
	EHC	zero_shot	0.14 / 0.66	0.15 / 0.47	0.10 / 0.37	18
		one_shot⧫	0.00 / 0.00	0.08 / 0.04	0.01 / 0.00	134
		few_shot⧫	0.02 / 0.13	0.07 / 0.03	0.00 / 0.01	196
phi3:3.8b	Direct	zero_shot	0.04 / 0.29	0.04 / 0.06	0.01 / 0.09	0
		one_shot	0.02 / 0.12	0.08 / 0.02	0.01 / 0.02	4
		few_shot	0.04 / 0.25	0.06 / 0.16	0.03 / 0.18	0
	EC	zero_shot⧫	0.12 / 0.23	0.18 / 0.43	0.12 / 0.29	20
		one_shot⧫	0.03 / 0.17	0.08 / 0.13	0.02 / 0.14	22
		few_shot	0.04 / 0.17	0.13 / 0.03	0.02 / 0.03	18
	ECR	zero_shot⧫	0.20 / 0.59	0.21 / 0.48	0.18 / 0.37	24
		one_shot	0.09 / 0.56	0.04 / 0.15	0.02 / 0.16	17
		few_shot⧫	0.06 / 0.41	0.02 / 0.10	0.02 / 0.15	56
	EHC	zero_shot⧫	0.25 / 0.56	0.19 / 0.47	0.19 / 0.37	76
		one_shot	0.04 / 0.21	0.11 / 0.07	0.02 / 0.08	20
		few_shot⧫	0.04 / 0.19	0.09 / 0.04	0.02 / 0.04	23
command-r:35b	Direct	zero_shot⧫	0.16 / 0.54	0.18 / 0.11	0.11 / 0.13	31
		one_shot	0.16 / 0.28	0.14 / 0.17	0.10 / 0.20	0
		few_shot	0.05 / 0.27	0.17 / 0.31	0.06 / 0.28	3
	EC	zero_shot	0.31 / 0.68	0.23 / 0.64	0.24 / 0.60	0
	EHC	zero_shot⧫	0.34 / 0.76	0.31 / 0.75	0.30 / 0.73	21
llama3:70b	Direct	zero_shot✠	0.34 / **0.80**	**0.29 / 0.77**	0.28 / **0.76**	1
		one_shot	0.32 / 0.72	0.32 / 0.65	0.26 / 0.62	1
		few_shot	0.28 / 0.69	0.29 / 0.70	0.27 / 0.67	1
	EC	zero_shot⧫	0.35 / 0.76	0.29 / 0.69	0.29 / 0.68	128
	EHC	zero_shot⧫	0.35 / 0.71	0.30 / 0.66	0.28 / 0.64	137

competetive scores on the benchmark **without any finetuning**. Smaller models (See Table I, ★) are able to archive solid performance with the right prompting strategy. Most configurations however (See Table I, ⧫) are disqualified for their high failure rate ($\geq 10\%$) during evaluation. This is attributed to the fact that output produced by the model wasn't conforming to the specifications of the respective prompting strategy and a patience level was reached leading to termination. Smaller models as phi3:3.8B performed evaluation faster (about 1 second/sample for Direct, 5 for EC/ECR and about 15 seconds/sample for EHC), but lack strong performance. The large models, as LLaMA3:70B took longer (about 5 seconds/sample for Direct and 50 seconds/sample for EHC). We've trained a few baseline models II for comparison and evaluated on the same test-set.

TABLE II
METRICS FOR BASELINE MODELS

Model	Metrics (Macro/Weighted)		
	Precision	Recall	F1 Score
distilbert-base-german-cased	0.31/0.84	0.31/0.86	0.31/0.85
agne/jobBERT-de	0.35/0.86	**0.36**/0.87	0.34/0.86
MNNaiveBayes	0.10 / 0.70	0.12/0.81	0.11/0.75
RandomForest◊	**0.49** / 0.86	0.34/**0.89**	**0.39/0.87**
LinearSVC	0.46 / **0.87**	0.34/**0.89**	0.38/**0.87**

We see that SotA models still outperform text generation based methods. All baselines apart from the Naive Bayes Classifier are within the same performance class, with the RandomForest Classifier excelling even compared to Transformer-based language models.

VI. DISCUSSION

We showed that some LLMs are capable of delivering zero-shot competetive performance on the task of Text Classification when compared to contemporary neural and conservative methods. Again, we'd like to note that all tested LLMs were most likely trained on the taxonomies WZ-1993, WZ-2003 and WZ-2008, which raises the question, whether we may be able to see the same performance on newly established taxonomies - most likely not. Other ways of injecting knowledge into the classification process would have to be explored.

Throughout our experiments, we've seen that introducing additional information (as in one-shot and few-shot prompting) may be harmful to performance *in this particular problem setup*. Most of the times a direct prompting strategy seems to be preferred. However, this may also be caused by the choice of prompt. We forced the models to output json in each output step, which may have hindered "flow of thought" of the model and lead to increased failure rates, as in our setup we failed a sample if the model output didn't conform to the specification after a fixed number of tries (the patience hyperparameter was chosen to be 3 in most cases).

Surprisingly, the more complex strategies like EC, ECR and EHC didn't increase performance for LLaMA:70B and command-r:35B, instead it seemed that it worsened performance unlike with smaller models where performance generally increased with this prompting strategy. The added complexity of the prompt seems to conflict with the complexity of the model. For large models a less strict prompt-output verification scheme should probably be explored, as less samples may fail. Another important point is the fact that we were able to "persuade" the model to output an argument (captured in the output value for 'reasoning') for the algorithms decision for the given industrial section. We acknowledge that it isn't a real "decription" of the internal model reasoning, as the model isn't keeping a hidden state over time but it still may prove very useful as it may help to reduce annotation time in real world applications.

A. Bias

Bias was introduced by selecting the sample distribution $p(x|y)$ which over-represents positions for roofers (Dachdecker) and the human annotation process. Inherently most companies allow for a multi-class classification, which introduces annotation conflicts and hinders learning. The training and test data distributions put about 80% of the probability mass on samples with sections F (Baugewerbe - Construction) and N78 (Vermittlung und Überlassung von Arbeitskräften - Placement and leasing of workers). Furthermore the pre-trained models `distilbert-base-german-cased` aswell as `agne/jobBERT-de` contain an unknown form of bias, as the pretraining datasets are not available.

In the conducted experiments, the choice of the wording of the prompt, the prompting strategy and the examples and solutions to the examples are chosen randomly from the training data distribution. It may be that choice of examples strongly influences the models performance on the test set.

VII. SUMMARY AND OUTLOOK

A. Summary

In this work, we've explored the ability of LLMs to perform Text Classification without any further finetuning. We've empirically shown that for some models the right prompting strategy yields comparable performance to methods which require extensive data (See Table II, ◊). Prompting LLaMA3:8b with the proposed Extract-Hypothesize-Classify Strategy (See Table I, ★) or LLaMA3:70b with a Direct Classification Strategy (See Table I, ✠) shows promise. Challenges remain, as for some LLMs failure rates are high (See Table I, ⧫) and reasoning capability is limited. Now, we'd like to answer the two questions posed in the introduction:

1) *How may we harness the strong prior, search and reasoning ability of LLMs for knowledge intensive Text Classification directly when we have little to no labeled data?*
   We may choose a good combination of LLM (one that is fit for our use-case) and prompting strategy. Additionally, we may have to inject domain knowledge to ensure valid output.
2) *How do different prompting strategies and models perform?*

> We have empirically shown that models are very sensitive to the a surplus of information. Generally a simple strategy, even a zero-shot strategy is the best for our use-case. This may of course be different for other taxonomies or text classification problems.

B. Outlook

Other ways of injecting knowledge into the classification process would have to be explored. To this end we hypothesize a sort of "Generation Augmented Retrieval" might be an interesting direrction of research, where unlike in RAG we augment the retriever with processed information about the data to be able to map more reliably into a latent space - and make a good prediction. An LLM may extract important information about the employer before mapping the extracted information into a latent space. This would indeed require labeled data to train the retriever - the question remains whether this approach yields a performance improvement.

A central take-away of our work is that for annotation-sparse problem setups LLM-based Text Classification will play an important role in the future - especially considering the growth in reasoning ability and capacity of modern models. Further work should be done in this direction. When context size grows and models are isntruction fine-tuned to make use of tools such as web-search and document-search we conjecture that LLM generation based classification will be able to outperform SoTA.

LLM Disclaimer

For the generation of LaTeX code (only Table structure) and for the purpose of prototyping experiment code we used Large Language Models.

Acknowledgments

We would like to thank our colleagues Christoph Holzmüller and Katerina Kostadinovska for helping us out with data annotation.

References

[1] R. Fechner, J. Dörpinghaus, and A. Firll, "Classifying industrial sectors from german textual data with a domain adapted transformer," in *2023 18th Conference on Computer Science and Intelligence Systems (FedCSIS)*. IEEE, 2023, pp. 463–470.

[2] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[3] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[4] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[5] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[6] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," *arXiv preprint arXiv:1908.10084*, 2019.

[7] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter," *arXiv preprint arXiv:1910.01108*, 2019.

[8] M. M. Mirończuk and J. Protasiewicz, "A recent overview of the state-of-the-art elements of text classification," *Expert Systems with Applications*, vol. 106, pp. 36–54, 2018.

[9] Z. Dai, Z. Yang, Y. Yang, J. Carbonell, Q. V. Le, and R. Salakhutdinov, "Transformer-xl: Attentive language models beyond a fixed-length context," *arXiv preprint arXiv:1901.02860*, 2019.

[10] I. Beltagy, M. E. Peters, and A. Cohan, "Longformer: The long-document transformer," *arXiv preprint arXiv:2004.05150*, 2020.

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

[12] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar *et al.*, "Llama: Open and efficient foundation language models," *arXiv preprint arXiv:2302.13971*, 2023.

[13] Y. Liu, G. Deng, Y. Li, K. Wang, T. Zhang, Y. Liu, H. Wang, Y. Zheng, and Y. Liu, "Prompt injection attack against llm-integrated applications," *arXiv preprint arXiv:2306.05499*, 2023.

[14] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou *et al.*, "Chain-of-thought prompting elicits reasoning in large language models," *Advances in Neural Information Processing Systems*, vol. 35, pp. 24 824–24 837, 2022.

[15] S. Yao, D. Yu, J. Zhao, I. Shafran, T. L. Griffiths, Y. Cao, and K. Narasimhan, "Tree of thoughts: Deliberate problem solving with large language models," *arXiv preprint arXiv:2305.10601*, 2023.

[16] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel *et al.*, "Retrieval-augmented generation for knowledge-intensive nlp tasks," *Advances in Neural Information Processing Systems*, vol. 33, pp. 9459–9474, 2020.

[17] H. Chase, "LangChain," Oct. 2022. [Online]. Available: https://github.com/langchain-ai/langchain

[18] M. Pejic-Bach, T. Bertoncel, M. Meško, and Ž. Krstić, "Text mining of industry 4.0 job advertisements," *International journal of information management*, vol. 50, pp. 416–431, 2020.

[19] R. Chaisricharoen, W. Srimaharaj, S. Chaising, and K. Pamanee, "Classification approach for industry standards categorization," in *2022 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (ECTI DAMT & NCON)*. IEEE, 2022, pp. 308–313.

[20] A. McCallum, K. Nigam *et al.*, "A comparison of event models for naive bayes text classification," in *AAAI-98 workshop on learning for text categorization*, vol. 752, no. 1. Madison, WI, 1998, pp. 41–48.

[21] A. M. Kibriya, E. Frank, B. Pfahringer, and G. Holmes, "Multinomial naive bayes for text categorization revisited," in *AI 2004: Advances in Artificial Intelligence: 17th Australian Joint Conference on Artificial Intelligence, Cairns, Australia, December 4-6, 2004. Proceedings 17*. Springer, 2005, pp. 488–499.

[22] H. Hayashi and Q. Zhao, "Quick induction of nntrees for text categorization based on discriminative multiple centroid approach," in *2010 IEEE International Conference on Systems, Man and Cybernetics*. IEEE, 2010, pp. 705–712.

[23] K. Kowsari, K. Jafari Meimandi, M. Heidarysafa, S. Mendu, L. Barnes, and D. Brown, "Text classification algorithms: A survey," *Information*, vol. 10, no. 4, p. 150, 2019.

[24] C. Ospino, "Occupations: Labor market classifications, taxonomies, and ontologies in the 21st century," *Inter-American Development Bank*, 2018.

[25] M. Rodrigues, Fernández-Macías, and Enrique, Sostero, Matteo, "A unified conceptual framework of tasks, skills and competences," Seville, 2021. [Online]. Available: https://joint-research-centre.ec.europa.eu/publications/unified-conceptual-framework-tasks-skills-and-competences_en

[26] A.-S. Gnehm, E. Bühlmann, and S. Clematide, "Evaluation of transfer learning and domain adaptation for analyzing german-speaking job advertisements," in *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, 2022, pp. 3892–3901.

[27] A.-S. Gnehm, E. Bühlmann, H. Buchs, and S. Clematide, "Fine-grained extraction and classification of skill requirements in german-speaking job ads." Association for Computational Linguistics, 2022.

[28] J. Büchel, J. Engler, and A. Mertens, "The demand for data skills in german companies: Evidence from online job advertisements," *How to Reconstruct Ukraine? Challenges, Plans and the Role of the EU*, p. 56, 2023.

[29] B. Gehrke, H. Legler, M. Leidmann, and K. Hippe, "Forschungs- und wissensintensive wirtschaftszweige: Produktion, wertschöpfung und beschäftigung in deutschland sowie qualifikationserfordernisse im europäischen vergleich," Studien zum deutschen Innovationssystem, Tech. Rep., 2009.

[30] N. Gillmann and V. Hassler, "Coronabetroffenheit der wirtschaftszweige in gesamt-und ostdeutschland," *ifo Dresden berichtet*, vol. 27, no. 04, pp. 03–05, 2020.

[31] U. Kies, D. Klein, and A. Schulte, "Cluster wald und holz deutschland: Makroökonomische bedeutung, regionale zentren und strukturwandel der beschäftigung in holzbasierten wirtschaftszweigen," *Cluster in Mitteldeutschland–Strukturen, Potenziale, Förderung*, p. 103, 2012.

[32] V.-P. Niitamo, "Berufs-und qualifikationsanforderungen im ikt-bereich in europa erkennen und messen," *Schmidt, SL; Strietska-Ilina, O.; Dworschak, B*, pp. 194–201, 2005.

[33] J. Hartmann and G. Schütz, "Die klassifizierung der berufe und der wirtschaftszweige im sozio-oekonomischen panel-neuvercodung der daten 1984-2001," SOEP Survey Papers, Tech. Rep., 2017.

[34] M. Titze, M. Brachert, and A. Kubis, "The identification of regional industrial clusters using qualitative input–output analysis (qioa)," *Regional Studies*, vol. 45, no. 1, pp. 89–102, 2011.

[35] U. Kies, T. Mrosek, and A. Schulte, "Spatial analysis of regional industrial clusters in the german forest sector," *International Forestry Review*, vol. 11, no. 1, pp. 38–51, 2009.

[36] Ollama, "Ollama software repository," 2024. [Online]. Available: https://github.com/ollama/ollama

[37] distilbert, "distilbert-base-german-cased software repository," 2024. [Online]. Available: https://huggingface.co/distilbert/distilbert-base-german-cased

[38] A.-S. "Gnehm, E. Bühlmann, and S. Clematide, ""evaluation of transfer learning and domain adaptation for analyzing german-speaking job advertisements"," in *"Proceedings of the 13th Language Resources and Evaluation Conference"*. "Marseille, France": "European Language Resources Association", june "2022".

[39] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

[40] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[41] Statistisches Bundesamt, "Klassifikation der Wirtschaftszweige," Wiesbaden, 2008. [Online]. Available: https://www.destatis.de/static/DE/dokumente/klassifikation-wz-2008-3100100089004.pdf

Appendix

(I) Info about WZ08

A. Classification of industrial sectors

Classification of industrial sectors include classifications for international comparative research (e.g., NACE) and administrative subdivisions (such as the Eurostat definition for knowledge- and technology-intensive sectors based on NACE), see our discussion in [1]. These classifications are usually interrelated. For example, the Classification of Economic Activities (WZ) is developed by the Federal Statistical Office of Germany and has been refined since 1950, with WZ 2008 being the latest edition which we will discuss later. Its objective is to ensure uniformity in the classification of economic activities across all official statistics in Germany. The classification is hierarchically structured.

The "Klassifikation der Wirtschaftszweige" (Classification of Branches of Industry, abbreviated as WZ) is utilized in Germany, particularly by the Statistische Bundesamt (Federal Statistical Office), for the classification of employers' economic activities in official statistics. The latest version is WZ 2008[3], which renders WZ 2003 and 1993 obsolete. This classification is compatible with the European "Nomenclature statistique des activités économiques dans la Communauté européenne" (NACE), but it includes more detailed data. NACE is the European classification system developed by Eurostat in the 1970s and updated regularly, with NACE Rev. 2 being the latest version from 2008. It provides a framework for collecting and presenting statistical data by economic activity and is hierarchically structured. For further information, please see [41]. Similar to NACE, WZ 2008 provides several hierarchical levels. A first level describes 21 sections (letters A-U), a second divisions, a third groups, a fourth classes. In contrast to NACE, WZ 2008 adds subgroups as fifth level, which is, however, only added to particular classes. WZ08 includes Sections (21), A-U, Divisions (88), 01-99, Groups (272), 01.1-99.0, Classes (615), 01.11-99.00, and Sub-classes (839), 01.11.0-99.00.0, see Figure 2.

While sectors are broad and specific, for example A (Agriculture, Forestry and Fishing) and B (Mining and Quarrying), others lack clear definition at this level, for example S (Other Service Activities). Conversely, classes and groups often exhibit indistinguishable characteristics, and the designation of divisions and groups typically provides minimal additional insight (e.g., 77 "Rental and leasing activities" versus 77.1 "Renting and leasing of motor vehicles". Furthermore, a company may belong to multiple industrial sectors, such as several manufacturing divisions. Nevertheless, the official guidelines recommend labeling the most dominant sector. Consequently, while the taxonomy of industrial sectors is well-defined by WZ08, we rely on external data to train and evaluate our approaches.

The International Standard Industrial Classification of All Economic Activities (ISIC) is a United Nations system for classifying economic activities, updated periodically since 1948, with ISIC Rev. 4 being the latest version. ISIC underpins both NACE and the German WZ. Its objective is to provide categories for the collection and reporting of statistics, and it is hierarchically structured. Other approaches are based on some of these taxonomies. For instance, the Eurostat classification of NACE sectors is based on technology intensity (manufacturing) and knowledge intensity (services). Manufacturing industries are classified as high, medium-high, medium-low, or low technology, while services are divided into knowledge-intensive and less-knowledge-intensive categories. This classification is available for both NACE Rev. 1.1 and NACE Rev. 2 (Eurostat 2009).

(II) Data Distribution and Examples

B. Frequencies

C. Some samples of the dataset:

Given is a jobposting where a job for a roofer (Section F) is advertised, but the company posting the ad belongs to

[3] All data is accessible in both English and German at https://www.destatis.de/DE/Methoden/Klassifikationen/Gueter-Wirtschaftsklassifikationen/Downloads/klassifikation-wz-2008-englisch.html. In this text, we generally use the official English translation for examples.

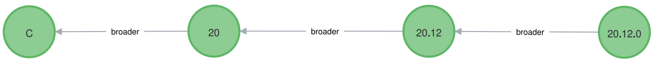

Fig. 2. An example subset of WZ08: Sector C (Manufacturing), division 20 (Manufacture of chemicals and chemical products), group 20.12 and class 20.12.0 (Manufacture of dyes and pigments).

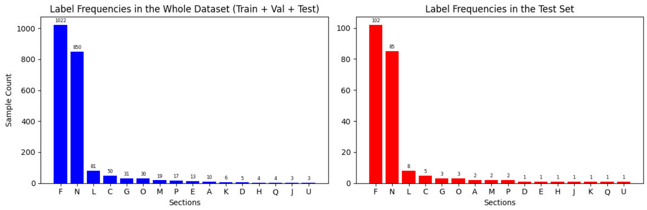

Fig. 3. Data frequencies of labels for whole dataset (left) and test set (right)

section N. Please note that we removed names of companies and any other sensetive information and replaced it with `[REMOVED]` in the following texts.

```
{'section_letter' : 'N',
 'text' : "Stellenangebot - Dachdecker/ Bauklempner zur Festeinstellung
gesucht (m/w) (Dachdecker/in und Bauklempner/in)

Überblick über das Stellenangebot

Referenznummer

[REMOVED]
Titel des Stellenangebots

Dachdecker/ Bauklempner zur Festeinstellung gesucht (m/w) (Dachdecker/in und Bauklempner/in)
Alternativberufe
Helfer/in - Hochbau
Konstruktionsmechaniker/in - Feinblechbautechnik
Stellenangebotsart

Arbeitsplatz (sozialversicherungspflichtig)
Arbeitgeber
[REMOVED]

Branche: Vermittlung von Arbeitskraeften, Betriebsgroeße: zwischen 51 und 500
Stellenbeschreibung

Im Auftrag unserer Partnerunternehmen suchen wir zur sofortigen Festanstellung mehrere
gelernte Dachdecker, Spengler, Dachdeckerhelfer (m/w) mit Berufserfahrung in Vollzeit.

Der Einsatz erfolgt Überwiegend im Süddeutschen Raum.
Ein Führerschein der Klasse3 (C1) ist von Vorteil, jedoch keine Bedingung.

Bei Interesse an diesem Angebot senden Sie uns bitte Ihre Bewerbungsunterlagen zu
(vorzugsweise per Email).
Ihre Aufgabe ist die Mithilfe bei:
- Bedachungen
- Dachum- und Ausbauarbeiten
- Modernisierungen
- Einbau von Dachfenstern und Gauben
- Wartung- und Reparaturarbeiten

Unsere Anforderungen an Sie:
- körperliche Fitness, Teamfähigkeit und selbständiges Arbeiten

Haben wir Ihr Interesse geweckt? Dann freuen wir uns auf Ihre Kontaktaufnahme."
}
```

Another example: A jobposting for a Product Manager posted by a company working in the industrial section G.

```
{'section_letter' : 'G',
'text' : "Produktmanager (m/w/d) Dachflächenfenster & Stahlelemente
DE, Germany
[REMOVED]
[REMOVED]

nach Vereinbarung / Qualifikation

Die [REMOVED] für Baustoffe, [REMOVED], ist mit einem Gruppenumsatz von
6,9 Mrd. Euro (2021) und über 1.500 Standorten in [REMOVED] eine der marktführenden
 Kooperationen im Baustoff-, Holz- und Fliesenhandel.
Auch in der Do-it-yourself-Branche nimmt das Unternehmen mit den [REMOVED] eine
führende Position ein. Zur Dienstleistungszentrale gehören als Tochterunternehmen
[REMOVED], die [REMOVED],
der [REMOVED] Versicherungsdienst, die [REMOVED] Logistik sowie die [REMOVED]
Beratungs- und
Beteiligungsgesellschaft und hagedoo mit insgesamt ca. 1.400 Mitarbeitern. Sie unterstützen
die Gesellschafter der [REMOVED]-Kooperation flächendeckend in sämtlichen
Bereichen deren unternehmerischen Handelns.

Für die Abteilung Einkaufssteuerung Logistikfachhandel unserer Zentrale in
[REMOVED] suchen wir ab
sofort in Vollzeit einen
Produktmanager (m/w/d) Dachflächenfenster & Stahlelemente
* Kontinuierliche Analyse und Weiterentwicklung aller Sortimente von [REMOVED] Logistik
* Sicherstellung der Einhaltung aller Prozesse des Warenflusses, insbesondere sämtlicher
 Listungstätigkeiten wie Stammdaten, Ein- und Verkaufspreise sowie Logistikparameter
* Abstimmung mit Lieferanten, dem Einkauf und der Logistik zur Aufrechterhaltung
der Lieferfähigkeit
* Erstellung von Vorgaben für die Abteilungen Customer Service und Beschaffung
* Projektarbeit zur Optimierung von Sortimenten und Prozessen
* Monitoring warenwirtschaftlicher Kennziffern

* Abgeschlossene kaufmännische Ausbildung oder Studienabschluss in einem für diese Position
relevanten Fachgebiet
* Berufserfahrung im Baustoffhandel oder in der Industrie im Ein- bzw. Verkauf ist von Vorteil
* Gute MS Office-Kenntnisse werden vorausgesetzt
* Grundlegende Sortiments- und Lieferantenkenntnisse im Bereich Dachflächenfenster oder
Stahlelemente sind wünschenswert
* Analytische Denk- und strukturierte Vorgehensweise sowie Innovationskraft und Vorwärtsdrang
* Selbstständigkeit, ausgeprägte Kommunikationsstärke, Überzeugungskraft und Belastbarkeit

* 30 Tage Urlaub
* Weihnachts- und Urlaubsgeld
* Mobiles Arbeiten und flexible Arbeitszeitkonten inklusive Gleitzeittagen
* Spannendes Aufgabengebiet und eigenverantwortliches Arbeiten
* Weiterentwicklungsmöglichkeiten durch Projektarbeit sowie weitergehende
Qualifizierungsmöglichkeiten
* Angenehme und offene Arbeitsatmosphäre in einem hochmotivierten und sympathischen
Team mit Patenmodell
* [REMOVED] mit diversen Vergünstigungen
* Rabatte bei regionalen Partnern (u. a. Heide Park, Fitnessstudio)
* Eigener Versicherungsdienst und vermögenswirksame Leistungen
* Betriebliche Benefits wie u.a. eine Kantine, Firmenfeiern, Gesundheitswochen und
freies WLAN für Mitarbeiter

Auf einen Blick:
* Bereich: Produktmanagement
* Einsatzort: [REMOVED]
* Arbeitszeit: 38,5 Stunden/Woche
* Eintrittstermin: ab sofort
* Arbeitsverhältnis: unbefristet"
}
```