Exploring Large Language Models for Feature Selection: A Data-centric Perspective

Dawei Li*
Arizona State University
Tempe, AZ, USA
daweili5@asu.edu

Zhen Tan*
Arizona State University
Tempe, AZ, USA
ztan36@asu.edu

Huan Liu
Arizona State University
Tempe, AZ, USA
huanliu@asu.edu

ABSTRACT

The rapid advancement of Large Language Models (LLMs) has significantly influenced various domains, leveraging their exceptional few-shot and zero-shot learning capabilities. In this work, we aim to explore and understand the LLMs-based feature selection methods from a data-centric perspective. We begin by categorizing existing feature selection methods with LLMs into two groups: data-driven feature selection which requires numerical values of samples to do statistical inference and text-based feature selection which utilizes prior knowledge of LLMs to do semantical associations using descriptive context. We conduct experiments in both classification and regression tasks with LLMs in various sizes (e.g., GPT-4, ChatGPT and LLaMA-2). Our findings emphasize the effectiveness and robustness of text-based feature selection methods and showcase their potentials using a real-world medical application. We also discuss the challenges and future opportunities in employing LLMs for feature selection, offering insights for further research and development in this emerging field.

1. INTRODUCTION

Recent years have witnessed the remarkable development of Large Language Models (LLMs) [1; 4; 53; 58] across various domains and areas [37; 6; 36; 3]. By leveraging extensive training corpora and well-designed prompting strategies, LLMs demonstrate impressive few-shot and zero-shot capabilities in diverse tasks such as question answering [65; 64; 57], information extraction [60] and knowledge discovery [46; 63; 62]. The tuning-free nature also makes in-context learning (ICL) in LLMs achieve a great balance between efficiency and effectiveness [54].

Feature selection [10; 35] is a critical data serving step that ensures relevant and high-quality data for downstream machine learning and data mining applications. While existing data-driven selection methods have achieved great success in scenarios with abundant data and metadata, there is an increasing demand for efficient feature selection methods with few or even zero samples for various reasons [72]. This need is particularly pronounced in sensitive applications such as predicting survival times for cancer patients [56; 66], where privacy concerns may prevent hospitals and patients from sharing their data, posing difficulties in the feature selection and engineering process. To address this challenge, recent studies [26; 23] have explored leveraging the few-shot capability in LLMs to perform feature selection in low-resource settings and got promising results.

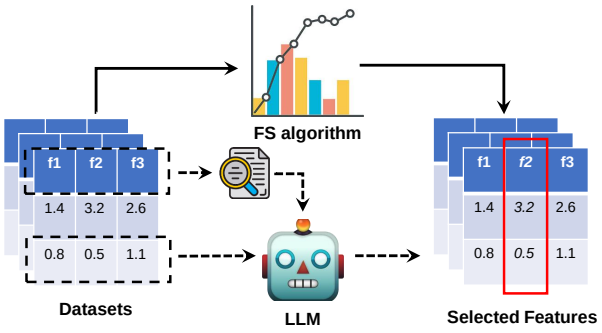

Figure 1: Comparison of traditional feature selection (FS) algorithms and LLM-based methods. Instead of requiring the whole dataset to make statistic inference, recent works prompt LLMs to select features in an efficient way. This is often achieved in a (*i*) data-driven, or (*ii*) text-based way.

In this work, our objective is to thoroughly explore and understand LLMs-based feature selection methods from a data-centric perspective. The conclusions and insights drawn from this exploration can provide insightful guidance for real-world applications where different types of resources and data are available. To begin with, we categorize the prompting strategies in previous studies [8; 26; 40; 23] for LLMs-based feature selection into two groups: (*i*) data-driven methods, which provide specific samples to LLMs [40; 23], and (*ii*) text-based methods, which incorporate feature and task descriptions into the instruction [8; 26]. These two prompting strategies require different data types: data-driven methods rely on sample points from datasets to do statistical inference while text-based methods need descriptive context for better semantic association between features and target variables. Figure 1 presents an overall comparison between the abovementioned methods and traditional feature selection algorithms. These differences make us curious about how LLMs perform with each of them under different data availability settings.

We conduct extensive experiments to explore the two methods in both classification and regression tasks with different LLMs in various sizes (E.g. GPT-4, ChatGPT and LLaMA-2). ***A key finding*** based on the results is that, text-based

*Equal Constributions

feature selection using LLMs is more effective and stable across various low-resource settings. Additionally, it shows a more pronounced scaling law with respect to the size of LLMs compared to data-driven approaches. Furthermore, we carried out a comparative evaluation between text-based feature selection using LLMs and traditional feature selection methods. ***A general observation*** is that, the text-based approach is relatively more robust and competitive across different resource availability settings.

Based on the abovementioned findings, we further explore the *applicability* of text-based feature selection with LLMs in a medical application. Specifically, we focus on the prediction of survival time for cancer patients [56; 66], which is a crucial task to evaluate both patient health and treatment effectiveness. To enhance the LLMs' understanding of medical-specific gene names, we developed a **R**etrieval-**A**ugmented **F**eature **S**election **(RAFS)** method that leverages descriptions from the National Institutes of Health (NIH) as auxiliary context. Experiment results demonstrate our RAFS's effectiveness in performing effective feature selection while safeguarding patient's privacy. Finally, we outline the existing challenges and potential opportunities in employing LLMs for feature selection.

To summarize, our contributions in this work are as follows:

- We propose a general taxonomy for the existing LLMs-based feature selection methods, splitting them into data-driven and text-based methods.
- Through an analysis under varying data availability conditions, we identify the strengths and weaknesses of these two methods, finding that text-based approaches are more effective and robust.
- We showcase the utilization of the text-based feature selection method with LLMs in a real-world medical application and introduce RAFS, a method designed to handle domain-specific feature selection with LLMs.
- We systematically analyze the existing challenges and potential future directions for using LLMs in feature selection, providing further insights and guidelines for future studies.

2. RELATED WORK

2.1 Feature Selection

Feature selection is the process of identifying and selecting the most relevant and important features or variables from a dataset to improve the performance and efficiency of a machine learning model [10; 20; 5; 35]. These feature selection methods can be generally categorized into three groups: filter, wrapper, and embedded approaches. Filter methods [31] first rank features by performing correlation analysis and then selecting the most important ones for the following learning step. Typical filter methods include mutual information [32; 11], Fisher score [24; 19] and maximum mean discrepancy [52]. By contrast, wrapper methods [30] use heuristic search strategies to identify a feature subset that optimally enhances the performance of certain prediction models (e.g., sequential selection [42] and recursive feature elimination [21]). For embedded approaches, it works together with specific machine learning models in the training phase by adding various regularization items in the loss function to encourage feature sparsity [55; 71].

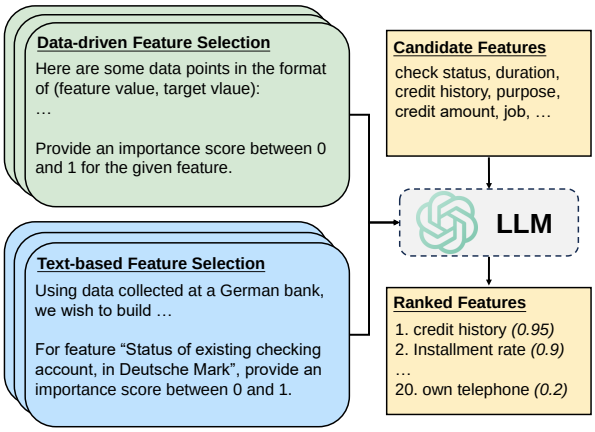

Figure 2: Prompting strategies for data-driven and text-based feature selection methods with LLMs.

2.2 Feature Selection with LLMs

There are already some works exploring the adaptation of LLMs in feature selection. [8] try to extract the relevant knowledge from LLMs as the task prior to performing feature selection, reinforcement learning and casual discovery. For feature selection, they design a prompt to instruct GPT-3 [4] to generate whether given features are important by answering "Yes" or "No". Following them, [26] expand the LLMs-based feature selection and propose three different pipelines that directly utilize the generated text output. They also conduct extensive experiments in evaluation across various model scales and prompting strategies. Besides, some studies devise more complex pipelines with LLMs in feature selection and feature engineering. [40] introduce an In-Context Evolutionary Search (ICE-SEARCH) in Medical Predictive Analytics (MPA) applications. It involves recurrently optimizing the selected features by prompting LLMs to perform feature filtering based on test scores. [23] employ LLMs as feature engineers to produce meta-features beyond the original features and combine them with simple machine learning models to improve predictions in downstream tasks. In this work, we aim to explore and understand LLMs in performing feature selection from a data perspective, offering further insights and hints for the adaptation of LLM-based feature selectors in real-world applications.

3. A DATA-CENTRIC TAXONOMY

Given a pre-trained LLM M, we follow the scoring-based method proposed by [26], which prompt M to generate an importance score s_i for the given feature/ concept f_i in the dataset d:

$$s_i = \mathrm{M}(P_{f_i}), \quad i \in \{1, ..., l\}, \tag{1}$$

where l is the total number of the features in dataset d. P_{f_i} refers to the specific prompt we use to generate the importance score. We will discuss two methods for constructing prompts in Sections 3.1 and 3.2, each focusing on different capabilities of LLMs. Figure 2 demonstrates the detailed prompting strategy for each of them.

3.1 Data-driven Feature Selection

Recently, LLMs have been employed to directly handle nu-

meric data, demonstrating their capabilities in numerical prediction and analytics [18; 27]. Therefore, we build a data-driven feature selection method with LLMs by providing both features' value n_{f_i} and the value of the target variable n_y. Intuitively, LLMs are supposed to infer the correlation and perform statistical analysis to determine the importance of the given feature in the dataset.

To be more specific, assume there are m samples available in the dataset d, we first build the sample pairs SP_i using values of the i_{th} feature and target variable:

$$SP_i = \{(n^j_{f_i}, n^j_y)\}, \quad i \in \{1, ..., l\}, j \in \{1, ..., m\}. \quad (2)$$

Then, we curate the prompt P_{f_i} using SP_i as few-shot examples and other instruction context C:

$$P^{Data}_{f_i} = \text{prompt}(C, SP_i), \quad (3)$$

here *prompt* is a function to concatenate the information and build a fluent instruction for LLMs.

3.2 Text-based Feature Selection

Another line of work [8; 26] tries to employ the extensive semantics knowledge in LLMs [33] to perform feature selection. Specifically, they incorporate detailed dataset descriptions in the prompt, instructing LLMs to semantically distinguish the importance of a given feature using their inherent knowledge and experience.

In our studies, we consider two concrete descriptive contexts: dataset description (des_d) and feature description (des_{f_i}). The dataset description includes the task's objective, details about the dataset's collection, and an explanation of the target variable. The feature description focuses on detailing and clarifying the feature to be scored.

Formally, we build prompts by integrating the abovementioned information:

$$P^{Text}_{f_i} = \text{prompt}(C, des_d, des_{f_i}). \quad (4)$$

We give specific instruction examples for the two feature selection methods in Appendix A.

4. ANALYSES

4.1 Experiment Settings

In this section, we evaluate the performance of the LLM-based feature selection methods using various datasets and models.

Models. Below are the LLMs used in our experiment.

- LLaMA-2 [58]: 7B parameters.
- LLaMA-2 [58]: 13B parameters.
- ChatGPT [45]: ∼175B parameters[1].
- GPT-4 [1]: ∼1.7T parameters[1].

We use the "gpt-4-turbo-2024-04-09" and "gpt-3.5-turbo-0125 models via API calling. For LLaMA-2, we do local inference with the checkpoints available from Huggingface, namely "llama-2-70b-chat-hf" and "llama-2-13b-chat-hf".

Compared Methods As the main methods to be analyzed in this section, we use "w/ data" and "w/ text" to represent the data-driven and text-based feature selection methods. We also compare the LLM-based feature selection methods with the following traditional feature selection baselines:

- Filtering by Mutual Information (MI) [32].
- Recursive Feature Elimination (RFE) [21].
- Minimum Redundancy Maximum Relevance selection (MRMR) [11].
- Random feature selection.

Dataset	# of samples	# of features
Adult	48842	14
Bank	45211	16
Communities	1994	102
Credit-g	1000	20
Heart	918	11
Myocardial	686	92
Diabetes	442	20
NBA	538	28
Rideshare	5000	18
Wine	6497	11

Table 1: Statistics of the datasets used.

Datasets. In our evaluation, we consider both classification and regression tasks. For the classification task, we use six datasets: Adult [2], Bank [44], Communities [50], Credit-g [28], Heart[2] and Myocardial [16]. For the regression task, we use four datasets: Diabetes [13], NBA[3], Rideshare[4] and Wine [2]. Detailed statistics of datasets are given in Table 1.

Implementation Details. For each dataset, we fix the feature selection ratio to 30%. We vary the data availability for evaluations with 16-shot, 32-shot, 64-shot, and 128-shot configurations. The test performance is measured using a downstream L2-penalized logistic/ linear regression model, selected via grid search with 5-fold cross-validation. We use the area under the ROC curve (AUROC) to evaluate classification tasks and mean absolute error (MAE) for regression.

4.2 Result Analysis

We present our main experimental results in Figure 3 and Figure 4 for analyzing, and highlighting the following findings for answering the RESEARCH QUESTION:

Finding 1: Text-based feature selection is more effective than data-driven ones with LLMs in low-resource settings. As results demonstrated in Figure 3 (a), almost in every LLM and task (except LLaMA-2-7B in classification), the performance of small machine learning models with the text-based feature selection method surpasses that of the data-driven feature selection method. This finding is consistent when we delve into feature selection methods' performance in each data availability, as depicted in Figure 4. Additionally, in Figure 3 (a), we notice for the same LLM, the text-based feature selection method

[1]∼ denotes the estimated size [26] of closed-source LLMs

[2]https://kaggle.com/datasets/fedesoriano/heart-failure-prediction

[3]https://www.kaggle.com/datasets/bryanchungweather/nba-player-stats-dataset-for-the-2023-2024

[4]https://www.kaggle.com/datasets/aaronweymouth/nyc-rideshare-raw-data

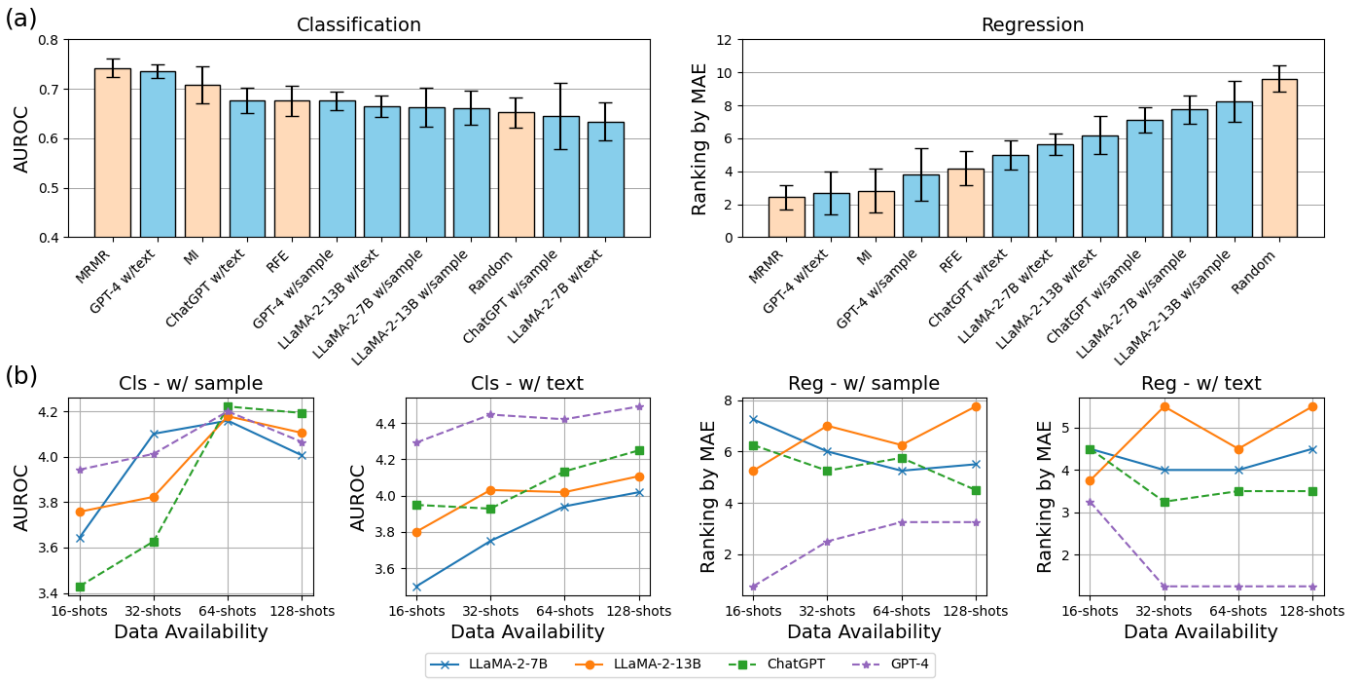

Figure 3: (a) Average AUROC (left; higher is better) and ranking by MAE (right; lower is better) across all datasets. (b) Each LLM's feature selection results, separated by task types (CLS and REG) and selection methods (w/sample and w/text).

usually leads to a smaller standard variant among various data availability settings. This further underscores the robustness and independence of the text-based feature selection method with respect to sample size.

	AUROC	Ranking by MAE
MI	0.779	**1.75**
RFE	0.758	3.50
MRMR	**0.798**	2.25
GPT-4 w/text	0.783	2.50

Table 2: Feature selection results in the full dataset with traditional data-driven methods and "GPT-4 w/text".

Finding 2: Text-based feature selection with the most advanced LLMs achieves comparable performance with traditional feature selection methods in every data availability setting. In Figure 3 (a), we observe that while GPT-4 with the text-based feature selection method performs slightly below the best traditional method (MRMR), it still demonstrates comparable performance, making it a competitive feature selection method in few-shot scenarios. However, when the LLM backbone is switched to less capable models, such as LLaMA, the text-based selection method shows a significant performance drop. Additionally, we experiment on the full dataset using 'GPT-4 w/text' alongside three traditional feature selection methods, and found that GPT-4 with the text-based method remains competitive even in the full-shot scenario.

Finding 3: Data-driven feature selection using LLMs struggles when number of samples increases. An interesting phenomenon we observed is a significant performance drop in the classification task when the sample size increases from 64 to 128 using the data-driven feature selection method (Figure 3 (b)). This drop is consistently observed across all four LLMs, indicating that each model generates poorer feature subsets as the sample size grows. We attribute this issue to LLMs struggling with processing long sequences, a challenge highlighted in many previous studies [12; 39]. This limitation constrains the effectiveness of data-driven feature selection, which is why we did not include it in the full-shot experiment.

Finding 4: Text-based feature selection exhibits a stronger scaling law with model size compared to data-driven feature selection with LLMs. We investigated how scaling laws in model size affect feature selection capabilities. In Figure 3 (b), we observe a clear correlation between the size of LLMs and their text-based feature selection capabilities. In contrast, while GPT-4 shows significantly superior performance in data-driven feature selection, the other three LLMs do not clearly follow the scaling law. This suggests that text-based feature selection is a reliable approach that can be enhanced by using powerful LLMs.

5. SURVIVAL PREDICTION - A CASE STUDY

We use a biomedical task to showcase the utilization of LLMs-based feature selection in real-world applications. Survival time prediction [56; 66] aims to predict cancer patients' survival time based on their physical and physiological indicators, playing a critical role in patient risk management and boosting treatment selection. One of the significant challenges in survival prediction datasets is the huge volume of features (e.g., there are around 20,000 gene expression features in the TCGA [56] dataset). While previous studies performed data-driven feature selection methods such as principal component analysis (PCA) to address this issue [67], as we mentioned in Section 1, It would cause serious

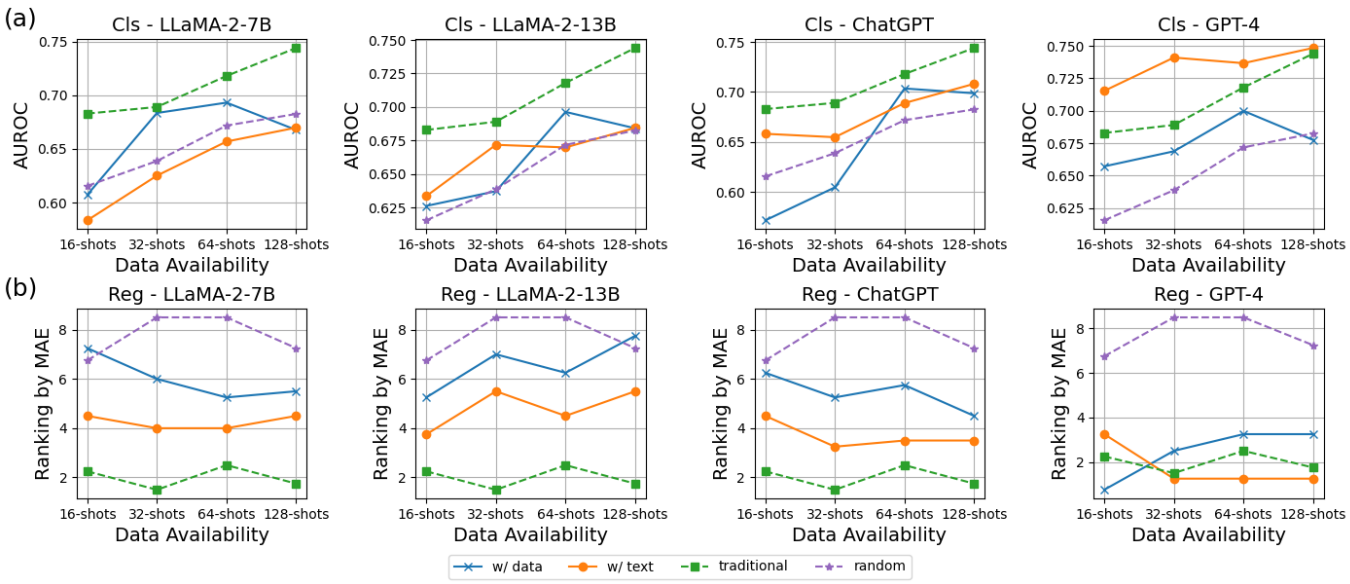

Figure 4: (a) Each feature selection method's results in the classification task, categorized by different LLMs; for each method, we add an error bar to represent its standard variant among various data availabilities. (b) Each feature selection method's results in the regression task, categorized by different LLMs. In each sub-figure, we include the average performance of traditional data-driven methods and the random selection method for comparison.

privacy concerns for both patients and hospitals.

Impressed by the competitive performance and sample-free nature of text-based feature selection with LLMs, here we adopt it in the survival prediction application. In our preliminary experiments, we found LLMs have difficulties in directly understanding the domain-specific feature name (e.g., gene ID). Therefore, we borrow insights from retrieval-augmented generation (RAG) with LLMs [15; 7; 34] and propose **R**etrieval-**A**ugmented **F**eature **S**election **(RAFS)** to efficiently handle these biomedical-specific feature names. Specifically, we retrieve meta information (e.g., official full name, summary and annotation information) about each feature name from the online National Center for Biotechnology Information (NCBI)[5] and provide this information to LLMs as the support document for better feature selection.

5.1 Experiment Settings

We conduct experiments using the Lung Adenocarcinoma (LUAD) dataset in The Cancer Genome Atlas (TCGA) benchmark [56]. Akin to [67], we use clinical indicators and gene expression as the full feature set and fix the feature selection ratio to be 30%. We use PriorityLasso [29] as our machine learning backbone and report three metrics: Antolini's Concordance (Antolini's C) [56], Integrated Brier score (IBS) [17] and D-Calibration (D-CAL) [22], all of which are commonly-used metrics for survival prediction.

5.2 Result Analysis

As the results show in Table 3, we find that even training the model on a randomly selected subset yields slightly better performance than training on the full feature set. This implies the huge volume of features in TCGA-LUAD negatively impacts model performance, highlighting the importance of feature selection. Moreover, we notice feature selection with our RAFS leads to significant performance improvements and consistently outperforms the random selection baseline. These findings suggest that RAFS is an effective approach for handling privacy-sensitive and large-scale biomedical datasets.

[5] https://www.ncbi.nlm.nih.gov/

	Antolini's C↑	IBS↓	D-CAL↓
PriorityLasso	0.6306	0.1863	1.8518
w/ random	0.6516	0.1833	2.0255
w/ RAFS	**0.6566**	**0.1830**	**1.7666**

Table 3: Experiment results in TCGA-LUAD. We add random selection as the baseline to compare our RAFS with.

6. OUTLOOK

In this section, we discuss potential opportunities for LLMs in feature selection, aiming to provide guidelines and hints for future works.

Synergy of LLMs-based and traditional feature selection. As we discuss in Sections 1 and 4.2, text-based feature selection with LLMs is competitive and resource-efficient compared with traditional feature selection methods. However, each approach relies on different sources of information—specific samples or context descriptions to perform feature selection. This diversity in information utilization makes them complementary. It would be valuable to explore how to combine text-based and traditional feature selection methods to create more effective and robust feature selection systems across various data availability scenarios. Also, it would be interesting to explore the synergy of text-based and data-driven methods to further enhance LLMs-based feature selection under resource constrains.

Data-driven analysis with Agentic LLMs. In Section 4.2, we conclude that poor statistical inference capabil-

ities in long-sequence input hinder LLMs in data-driven feature selection. While this finding implies the sole adaptation of LLMs may not be enough for performing data-driven feature selection, the introduction of agent-based LLMs should be considered as an alternative [68; 61]. These methods equip LLM with various tools [47; 70; 51] and APIs [48; 41], enabling them to execute actions and plans to solve complex and multi-step problems. However, there are only a few works that focus on the development of agentic LLMs as data engineers and analytics [25; 14; 59], for actively performing various features or data processing with the assistance of statistical tools or software. Research in this direction will be valuable for enhancing and evaluating LLMs from analytical and statistical perspectives.

Foundation models for feature/data engineering. Many recent works have developed various foundation models in many data mining and machine learning fields, such as graph learning [38; 43; 69] and time series prediction [49; 27]. A large foundation model for feature/ data engineering should be able to understand different types of information from the datasets and perform efficient manipulation and processing [9] to prepare appropriate data for downstream models/ applications. Developing such a foundation model would greatly benefit the data mining and machine learning communities by providing a unified, easy-to-use interface for complex data processing tasks.

7. CONCLUSION

In this study, we explore feature selection methods based on LLMs from a data-centric perspective. We categorize existing LLM-based feature selection approaches into two main types: data-driven, which relies on statistical inference from specific samples, and text-based, which utilizes the extensive knowledge of LLMs for semantic association. Our experiments and analyses reveal that text-based feature selection with LLMs outperforms data-driven methods in terms of effectiveness, stability, and robustness. Based on these findings, we introduce a Retrieval-Augmented Feature Selection (RAFS) method designed to manage large volumes of domain-specific feature candidates in the context of cancer survival time prediction. Additionally, we provide a comprehensive analysis of the current challenges and potential opportunities at the intersection of LLMs and feature selection/engineering in Section 6, aiming to offer insights and guidance for future research in this area.

Acknowledgments

The material in this presentation is based upon work supported by the U.S. Department of Homeland Security under Grant Award Number, 17STQAC00001-08-00, the U.S. Office of Naval Research (ONR) under grant N00014-21-1-4002, and the National Science Foundation (NSF) under grants IIS-2229461. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the U.S. Department of Homeland Security and the National Science Foundation.

8. REFERENCES

[1] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[2] A. Asuncion, D. Newman, et al. Uci machine learning repository, 2007.

[3] A. Beigi, B. Jiang, D. Li, T. Kumarage, Z. Tan, P. Shaeri, and H. Liu. Lrq-fact: Llm-generated relevant questions for multimodal fact-checking. *arXiv preprint arXiv:2410.04616*, 2024.

[4] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[5] G. Chandrashekar and F. Sahin. A survey on feature selection methods. *Computers & electrical engineering*, 40(1):16–28, 2014.

[6] Y. Chang, X. Wang, J. Wang, Y. Wu, L. Yang, K. Zhu, H. Chen, X. Yi, C. Wang, Y. Wang, et al. A survey on evaluation of large language models. *ACM Transactions on Intelligent Systems and Technology*, 2023.

[7] J. Chen, H. Lin, X. Han, and L. Sun. Benchmarking large language models in retrieval-augmented generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 17754–17762, 2024.

[8] K. Choi, C. Cundy, S. Srivastava, and S. Ermon. Lmpriors: Pre-trained language models as task-specific priors. In *NeurIPS 2022 Foundation Models for Decision Making Workshop*, 2022.

[9] L. Cui, H. Li, K. Chen, L. Shou, and G. Chen. Tabular data augmentation for machine learning: Progress and prospects of embracing generative ai. *arXiv preprint arXiv:2407.21523*, 2024.

[10] M. Dash and H. Liu. Feature selection for classification. *Intelligent data analysis*, 1(1-4):131–156, 1997.

[11] C. Ding and H. Peng. Minimum redundancy feature selection from microarray gene expression data. *Journal of bioinformatics and computational biology*, 3(02):185–205, 2005.

[12] Z. Dong, T. Tang, J. Li, W. X. Zhao, and J.-R. Wen. Bamboo: A comprehensive benchmark for evaluating long text modeling capacities of large language models. *arXiv preprint arXiv:2309.13345*, 2023.

[13] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least angle regression. *The Annals of statistics*, 32(2):407–451, 2004.

[14] X. Fang, W. Xu, F. A. Tan, J. Zhang, Z. Hu, Y. J. Qi, S. Nickleach, D. Socolinsky, S. Sengamedu, C. Faloutsos, et al. Large language models (llms) on tabular data: Prediction, generation, and understanding-a survey. *Transactions on Machine Learning Research*, 2024.

[15] Y. Gao, Y. Xiong, X. Gao, K. Jia, J. Pan, Y. Bi, Y. Dai, J. Sun, and H. Wang. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*, 2023.

[16] S. Golovenkin, V. Shulman, D. Rossiev, P. Shesternya, S. Nikulina, Y. Orlova, and V. Voino-Yasenetsky. Myocardial infarction complications. UCI Machine Learning Repository, 2020. DOI: https://doi.org/10.24432/C53P5M.

[17] E. Graf, C. Schmoor, W. Sauerbrei, and M. Schumacher. Assessment and comparison of prognostic classification schemes for survival data. *Statistics in medicine*, 18(17-18):2529–2545, 1999.

[18] N. Gruver, M. Finzi, S. Qiu, and A. G. Wilson. Large language models are zero-shot time series forecasters. *Advances in Neural Information Processing Systems*, 36, 2024.

[19] Q. Gu, Z. Li, and J. Han. Generalized fisher score for feature selection. In *Proceedings of the Twenty-Seventh Conference on Uncertainty in Artificial Intelligence*, pages 266–273, 2011.

[20] I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Journal of machine learning research*, 3(Mar):1157–1182, 2003.

[21] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik. Gene selection for cancer classification using support vector machines. *Machine learning*, 46:389–422, 2002.

[22] H. Haider, B. Hoehn, S. Davis, and R. Greiner. Effective ways to build and evaluate individual survival distributions. *Journal of Machine Learning Research*, 21(85):1–63, 2020.

[23] S. Han, J. Yoon, S. O. Arik, and T. Pfister. Large language models can automatically engineer features for few-shot tabular learning. In *Forty-first International Conference on Machine Learning*, 2024.

[24] P. E. Hart, D. G. Stork, R. O. Duda, et al. *Pattern classification*. Wiley Hoboken, 2000.

[25] S. Hong, Y. Lin, B. Liu, B. Wu, D. Li, J. Chen, J. Zhang, J. Wang, L. Zhang, M. Zhuge, et al. Data interpreter: An llm agent for data science. *arXiv preprint arXiv:2402.18679*, 2024.

[26] D. P. Jeong, Z. C. Lipton, and P. Ravikumar. Llm-select: Feature selection with large language models. *arXiv preprint arXiv:2407.02694*, 2024.

[27] M. Jin, S. Wang, L. Ma, Z. Chu, J. Y. Zhang, X. Shi, P.-Y. Chen, Y. Liang, Y.-F. Li, S. Pan, et al. Time-llm: Time series forecasting by reprogramming large language models. *arXiv preprint arXiv:2310.01728*, 2023.

[28] A. Kadra, M. Lindauer, F. Hutter, and J. Grabocka. Well-tuned simple nets excel on tabular datasets. *Advances in neural information processing systems*, 34:23928–23941, 2021.

[29] S. Klau, V. Jurinovic, R. Hornung, T. Herold, and A.-L. Boulesteix. Priority-lasso: a simple hierarchical approach to the prediction of clinical outcome using multi-omics data. *BMC bioinformatics*, 19:1–14, 2018.

[30] R. Kohavi and G. H. John. Wrappers for feature subset selection. *Artificial intelligence*, 97(1-2):273–324, 1997.

[31] C. Lazar, J. Taminau, S. Meganck, D. Steenhoff, A. Coletta, C. Molter, V. de Schaetzen, R. Duque, H. Bersini, and A. Nowe. A survey on filter techniques for feature selection in gene expression microarray analysis. *IEEE/ACM transactions on computational biology and bioinformatics*, 9(4):1106–1119, 2012.

[32] D. D. Lewis. Feature selection and feature extraction for text categorization. In *Speech and Natural Language: Proceedings of a Workshop Held at Harriman, New York, February 23-26, 1992*, 1992.

[33] D. Li, Z. Tan, T. Chen, and H. Liu. Contextualization distillation from large language model for knowledge graph completion. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 458–477, 2024.

[34] D. Li, S. Yang, Z. Tan, J. Y. Baik, S. Yun, J. Lee, A. Chacko, B. Hou, D. Duong-Tran, Y. Ding, et al. Dalk: Dynamic co-augmentation of llms and kg to answer alzheimer's disease questions with scientific literature. *arXiv preprint arXiv:2405.04819*, 2024.

[35] J. Li, K. Cheng, S. Wang, F. Morstatter, R. P. Trevino, J. Tang, and H. Liu. Feature selection: A data perspective. *ACM computing surveys (CSUR)*, 50(6):1–45, 2017.

[36] Y. Li, A. Dao, W. Bao, Z. Tan, T. Chen, H. Liu, and Y. Kong. Facial affective behavior analysis with instruction tuning. *arXiv preprint arXiv:2404.05052*, 2024.

[37] P. Liang, R. Bommasani, T. Lee, D. Tsipras, D. Soylu, M. Yasunaga, Y. Zhang, D. Narayanan, Y. Wu, A. Kumar, et al. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*, 2022.

[38] J. Liu, C. Yang, Z. Lu, J. Chen, Y. Li, M. Zhang, T. Bai, Y. Fang, L. Sun, P. S. Yu, et al. Towards graph foundation models: A survey and beyond. *arXiv preprint arXiv:2310.11829*, 2023.

[39] N. F. Liu, K. Lin, J. Hewitt, A. Paranjape, M. Bevilacqua, F. Petroni, and P. Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024.

[40] S. Liu, F. Lvu, X. Liu, et al. Ice-search: A language model-driven feature selection approach. *arXiv preprint arXiv:2402.18609*, 2024.

[41] X. Liu, Z. Li, P. Li, S. Xia, X. Cui, L. Huang, H. Huang, W. Deng, and Z. He. Mmfakebench: A mixed-source multimodal misinformation detection benchmark for lvlms. *arXiv preprint arXiv:2406.08772*, 2024.

[42] S. Luo and Z. Chen. Sequential lasso cum ebic for feature selection with ultra-high dimensional feature space. *Journal of the American Statistical Association*, 109(507):1229–1240, 2014.

[43] H. Mao, Z. Chen, W. Tang, J. Zhao, Y. Ma, T. Zhao, N. Shah, M. Galkin, and J. Tang. Graph foundation models. *arXiv preprint arXiv:2402.02216*, 2024.

[44] S. Moro, P. Cortez, and P. Rita. A data-driven approach to predict the success of bank telemarketing. *Decision Support Systems*, 62:22–31, 2014.

[45] OpenAI. Introducing chatgpt. *OpenAI*, 2022.

[46] S. Pan, L. Luo, Y. Wang, C. Chen, J. Wang, and X. Wu. Unifying large language models and knowledge graphs: A roadmap. *IEEE Transactions on Knowledge and Data Engineering*, 2024.

[47] B. Paranjape, S. Lundberg, S. Singh, H. Hajishirzi, L. Zettlemoyer, and M. T. Ribeiro. Art: Automatic multi-step reasoning and tool-use for large language models. *arXiv preprint arXiv:2303.09014*, 2023.

[48] S. G. Patil, T. Zhang, X. Wang, and J. E. Gonzalez. Gorilla: Large language model connected with massive apis. *arXiv preprint arXiv:2305.15334*, 2023.

[49] K. Rasul, A. Ashok, A. R. Williams, A. Khorasani, G. Adamopoulos, R. Bhagwatkar, M. Biloš, H. Ghonia, N. V. Hassen, A. Schneider, et al. Lag-llama: Towards foundation models for time series forecasting. *arXiv preprint arXiv:2310.08278*, 2023.

[50] M. Redmond. Communities and Crime. UCI Machine Learning Repository, 2009. DOI: https://doi.org/10.24432/C53W3X.

[51] T. Schick, J. Dwivedi-Yu, R. Dessì, R. Raileanu, M. Lomeli, E. Hambro, L. Zettlemoyer, N. Cancedda, and T. Scialom. Toolformer: Language models can teach themselves to use tools. *Advances in Neural Information Processing Systems*, 36, 2024.

[52] L. Song, A. Smola, A. Gretton, J. Bedo, and K. Borgwardt. Feature selection via dependence maximization. *Journal of Machine Learning Research*, 13(5), 2012.

[53] Z. Tan, A. Beigi, S. Wang, R. Guo, A. Bhattacharjee, B. Jiang, M. Karami, J. Li, L. Cheng, and H. Liu. Large language models for data annotation: A survey. *arXiv preprint arXiv:2402.13446*, 2024.

[54] Z. Tan, J. Peng, T. Chen, and H. Liu. Tuning-free accountable intervention for llm deployment–a metacognitive approach. *arXiv preprint arXiv:2403.05636*, 2024.

[55] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 58(1):267–288, 1996.

[56] K. Tomczak, P. Czerwińska, and M. Wiznerowicz. Review the cancer genome atlas (tcga): an immeasurable source of knowledge. *Contemporary Oncology/Współczesna Onkologia*, 2015(1):68–77, 2015.

[57] Y. Tong, D. Li, S. Wang, Y. Wang, F. Teng, and J. Shang. Can llms learn from previous mistakes? investigating llms' errors to boost for reasoning. *arXiv preprint arXiv:2403.20046*, 2024.

[58] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

[59] P. Trirat, W. Jeong, and S. J. Hwang. Automl-agent: A multi-agent llm framework for full-pipeline automl. *arXiv preprint arXiv:2410.02958*, 2024.

[60] S. Wadhwa, S. Amir, and B. C. Wallace. Revisiting relation extraction in the era of large language models. In *Proceedings of the conference. Association for Computational Linguistics. Meeting*, volume 2023, page 15566. NIH Public Access, 2023.

[61] L. Wang, C. Ma, X. Feng, Z. Zhang, H. Yang, J. Zhang, Z. Chen, J. Tang, X. Chen, Y. Lin, et al. A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 18(6):186345, 2024.

[62] S. Wang, Z. Tan, R. Guo, and J. Li. Noise-robust fine-tuning of pretrained language models via external guidance. *arXiv preprint arXiv:2311.01108*, 2023.

[63] X. Wang, Z. Chen, H. Wang, Z. Li, W. Guo, et al. Large language model enhanced knowledge representation learning: A survey. *arXiv preprint arXiv:2407.00936*, 2024.

[64] X. Wang, J. Wei, D. Schuurmans, Q. V. Le, E. H. Chi, S. Narang, A. Chowdhery, and D. Zhou. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*, 2023.

[65] 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*, 35:24824–24837, 2022.

[66] D. Wissel, N. Janakarajan, A. Grover, E. Toniato, M. R. Martínez, and V. Boeva. Survboard: standardised benchmarking for multi-omics cancer survival models. *bioRxiv*, pages 2022–11, 2022.

[67] D. Wissel, D. Rowson, and V. Boeva. Systematic comparison of multi-omics survival models reveals a widespread lack of noise resistance. *Cell Reports Methods*, 3(4), 2023.

[68] Z. Xi, W. Chen, X. Guo, W. He, Y. Ding, B. Hong, M. Zhang, J. Wang, S. Jin, E. Zhou, et al. The rise and potential of large language model based agents: A survey. *arXiv preprint arXiv:2309.07864*, 2023.

[69] L. Xia, B. Kao, and C. Huang. Opengraph: Towards open graph foundation models. *arXiv preprint arXiv:2403.01121*, 2024.

[70] R. Yang, L. Song, Y. Li, S. Zhao, Y. Ge, X. Li, and Y. Shan. Gpt4tools: Teaching large language model to use tools via self-instruction. *Advances in Neural Information Processing Systems*, 36, 2024.

[71] M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 68(1):49–67, 2006.

[72] T. Zhang, T. Zhu, P. Xiong, H. Huo, Z. Tari, and W. Zhou. Correlated differential privacy: Feature selection in machine learning. *IEEE Transactions on Industrial Informatics*, 16(3):2115–2124, 2019.

APPENDIX

A. DETAILED INSTRUCTION

/* Main System Prompt */
For the given feature, your task is to provide a feature importance score (between 0 and 1; larger value indicates greater importance).

/* Specific Sample Vlaues */
Here are some data points in the format of (feature value, target value), please refer to this to determine how informative the feature is in predicting the target value:
(<0, no)
(no checking, no)
(<0, no)
(<0, no)
(0<=X<200, no)
(<0, no)
(>=200, no)
(<0, no)
(no checking, yes)
(no checking, yes)
(0<=X<200, yes)
(0<=X<200, yes)
(no checking, yes)
(0<=X<200, yes)
(0<=X<200, yes)
(<0, yes)

/* Output Format Instruction */
Here is an example:
‘‘‘
Question: What is the importance score for the given feature
Answer: The importance score is 0.9
‘‘‘

/* Main User Prompt*/
Question: What is the importance score for the given feature
Answer: The importance score is

Table 4: Detailed instruction for data-driven method in Credit-g dataset.

/* Dataset-specific Context */
Context: Using data collected at a German bank, we wish to build a machine learning model that can accurately predict whether a client carries high or low credit risk (target variable). The dataset contains a total of 20 features (e.g., credit history, savings account status). Prior to training the model, we first want to identify a subset of the 20 features that are most important for reliable prediction of the target variable.

/* Main System Prompt */
For each feature input by the user, your task is to provide a feature importance score (between 0 and 1; larger value indicates greater importance) for predicting whether an individual carries high credit risk and a reasoning behind how the importance score was assigned.

/* Output Format Instructions */
The output should be formatted as a JSON instance that conforms to the JSON schema below.

As an example, for the schema "properties": "foo": "title": "Foo", "description": "a list of strings", "type": "array", "items": "type": "string", "required": ["foo"] the object "foo": ["bar", "baz"] is a well formatted instance of the schema. The object "properties": "foo": ["bar", "baz"] is not well-formatted.

Here is the output schema:
‘‘‘
{"description": "Langchain Pydantic output parsing structure.", "properties": {"reasoning": {"title": "Reasoning", "description": "Logical reasoning behind feature importance score", "type": "string"}, "score": {"title": "Score", "description": "Feature importance score", "type": "number"}}, "required": ["score"]}
‘‘‘
/* Demonstration */
Here is an example output:
-Variable: Installment rate in percentage of disposable income
{"reasoning": "The installment rate as apercentage of disposable incomeprovides insight intoa person's financial responsibility and capability.This percentage can be seen as a measure of how much of a person's available income is committed to repaying their debts. If this rate is high, it might indicate that the person is taking more debt than they can comfortably repay and may hint a talack off inancial responsibility, implyinghigher credit risk. If this rate is low, it likely indicates that the person can manage their current financial obligations comfortably, implying lowercredit risk. Thus, the score is 0.9.", "score": 0.9}

/*Main User Prompt*/
Provide a score and reasoning for "Status of existing checking account, in Deutsche Mark." formatted according to the output schema above:

Table 5: Detailed instruction for text-based method in Credit-g dataset.